Le paquetage minitoc*

Jean-Pierre F. Drucbert
jean-pierre.drucbert@onera.fr

16 septembre 2005

*Ce document correspond à minitoc v43, daté 2005/09/16.

Sommaire

Table des matières

Liste des Figures

Liste des Tableaux

FIX INDEX ? FIX INDEX ?

Première partie

Manuel de l’utilisateur

Table des Matières

Chapitre 1

Le paquetage `minitoc`

Sommaire

Figures

Tableaux

1.1 Introduction

Le paquetage `minitoc`, écrit initialement par Nigel Ward et Dan Jurafsky, a été presque complètement reconstruit par Jean-Pierre F. Drucbert (ONERA/Centre de Toulouse). Ce paquetage crée une mini-table des matières (une « minitoc » [1]) au début de chaque chapitre d'un document. Il est aussi possible d'avoir une mini-liste des figures (une « minilot ») et une mini-liste des tableaux (une « minilot »). La classe de document devrait, bien sûr, définir des chapitres (classes telles que `book` ou `report`) ou des sections (classes telles que `article` [2]). Donc, ce paquetage ne devrait pas être utilisé avec des classes de document ne possédant pas les commandes de sectionnement standard (classes telles que `letter`). Lorsque la classe de document définit un niveau de sectionnement « partie » (c'est-à-dire des classes telles que `book`, `report` et `article`), vous pouvez créer une table des matières « partielle » (une « parttoc ») au début de chaque partie d'un document. Il est aussi possible d'avoir une liste des figures partielle (une « partlof ») et une liste des

[1] Le paquetage `minitoc` introduit son propre jargon, expliqué dans ce document. Il ne devrait, cependant, pas être trop difficile de l'apprendre et de l'utiliser ; il sera évidemment utilisé ici.

[2] Puisque la classe standard `proc` charge la classe standard `article`, cette classe sera simplement considérée comme une variante de la classe `article`.

tableaux partielle (une « partlot ») au début de chaque partie d'un document. Lorsque la classe de document ne possède pas de commande `\chapter` mais a une commande `\section`, vous pouvez utiliser des tables des matières au niveau section (« secttocs ») au début de chaque section ; et vous pouvez aussi avoir des listes des figures (« sectlofs ») ou des tableaux (« sectlots ») au niveau section.

Toutes ces tables (« minitocs », « partlofs », « sectlots », etc.) sont collectées référencées sous le terme de « mini-tables » (ou parfois « mini-listes »).

Note : vous ne pouvez pas utiliser des mini-tables au niveau chapitre et au niveau section dans le même document. Cette restriction est destinée à éviter des documents pleins de mini-tables des matières, listes de figures et listes de tableaux locales à chaque niveau de sectionnement.

Note : les commandes relatives au niveau partie ne sont définies que si la classe de document définit `\part`. Les commandes relatives au niveau section ne sont définies que si la classe de document définit `\section` mais ne définit pas `\chapter`.

La version courante de ce paquetage est #43.

1.1.1 Licence

Ce paquetage doit être distribué et/ou peut être modifié sous les conditions de la **LaTeX Project Public License**, soit la version 1.3 de cette licence soit (au choix) toute version ultérieure. La plus récente version de cette licence est dans

`http://www.latex-project.org/lppl.txt`

et la version 1.3 ou ultérieure fait partie de toutes les distributions de LaTeX version 2003/12/01 ou ultérieure.

Mais, s'il vous plaît, ne me parlez pas de versions bricolées ; elles ne seront pas maintenues.

1.2 Comment utiliser le paquetage minitoc

1.2.1 Charger le paquetage et créer les mini-tables

`\usepackage`
`\minitoc`

Pour utiliser le paquetage `minitoc`, vous devez introduire une commande :

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

dans le préambule du document[3]. La mini-table des matières sera dans le chapitre, après la commande `\chapter`, à l'emplacement de la commande `\minitoc`. La commande

[3]Cette commande doit être placée *après* toute modification faite sur les commandes de sectionnement ; si vous modifiez des commandes de sectionnement après le chargement du paquetage `minitoc`, celui-ci pourrait ne pas fonctionner correctement.

TAB. 1.1 – Commandes pour une `minitoc`

```
\documentclass[...]{book}
\usepackage[...options...]{minitoc}
...
\setlength{\mtcindent}{24pt}                 défaut
\setlength{\mtcskipamount}{\bigskipamount}   défaut
...
\setcounter{minitocdepth}{2}                 défaut
\renewcommand{\mtcfont}{\small\rmfamily\upshape\mdseries}          défaut
\renewcommand{\mtcSfont}{\small\rmfamily\upshape\bfseries}         défaut
ou :
\mtcsetdepth{minitoc}{2}                                           défaut
\mtcsetfont{minitoc}{*}{\small\rmfamily\upshape\mdseries}          défaut
\mtcsetfont{minitoc}{section}{\small\rmfamily\upshape\bfseries}    défaut
...
\begin{document}
...
\dominitoc
\dominilof
\dominilot
\tableofcontents                             ou \faketableofcontents
\listoffigures                               ou \fakelistoffigures
\listoftables                                ou \fakelistoftables
...
\chapter{...}
\minitoc                                     si vous en voulez une
\mtcskip
\minilof                                     si vous en voulez une
\mtcskip
\minilot                                     si vous en voulez une
...
```

`\minitoc` peut se placer *presque n'importe où*[4] à l'intérieur d'un chapitre. Bien sûr, il est préférable de la placer au début du chapitre, éventuellement après un texte introductif. Mais vous pouvez aussi décider de la placer à la fin du chapitre. Vous devriez utiliser les mêmes conventions dans tous les chapitres. Si vous souhaitez ajouter la mini-table des matières pour un chapitre, vous devez utiliser la séquence donnée dans le tableau 1.1. Pour chaque mini-table des matières, un fichier auxiliaire sera créé avec un nom de la forme ⟨*document*⟩`.mtc`⟨*N*⟩, où ⟨*N*⟩ est le numéro absolu du chapitre. « Absolu » signifie que ce numéro est unique, et croit toujours depuis le premier chapitre[5]. Le suffixe est `.mlf`⟨*N*⟩ pour les mini-listes de figures et `.mlt`⟨*N*⟩ pour les mini-listes de tableaux. (Si vous travaillez sous MS-DOS ou tout autre système d'exploitation utilisant des suffixes

[4]« Presque n'importe où » signifie « en un endroit normal », comme entre deux alinéas de texte normal, ou dans une minipage (suffisamment large), mais pas dans une position trop bizarre (comme une note marginale ou infrapaginale). Même un environnement multi-colonnes ou flottant peut être utilisé, mais avec soin. Mais notez qu'une minitoc peut être assez longue, si le chapitre est complexe et si vous demandez des détails avec une valeur élevée de `minitocdepth`.

[5]Le concept d'un compteur « absolu » pour les mini-tables a résolu certains problèmes ténébreux, et aussi frappé d'obsolescence certaines commandes, telles que `\firstpartis`, `\firstchapteris` et `\firstsectionis`.

courts pour les noms de fichiers, voyez la section 1.6 page 39 et la section 2.5 page 43). Il y a des commandes similaires aux niveaux partie ou section, selon la classe du document.

1.2.2 Préparation des mini-tables

`\dominitoc` `\dominilof` `\dominilot`

Les commandes[6] `\dominitoc`, `\dominilof` et `\dominilot` (pour les mini-tables au niveau chapitre, prennent les fichiers ⟨*document*⟩`.toc`, ⟨*document*⟩`.lof` et ⟨*document*⟩`.lot`, respectivement, et en prélèvent des tranches pour créer les fichiers ⟨*document*⟩`.mtc`⟨*N*⟩, ⟨*document*⟩`.mlf`⟨*N*⟩ et ⟨*document*⟩`.mlt`⟨*N*⟩.

`\dosecttoc` `\dosectlof` `\dosectlot` `\doparttoc` `\dopartlof` `\dopartlot` `\tableofcontents` `\listoffigures` `\listoftables`

Les commandes `\dosecttoc`, `\dosectlof` et `\dosectlot` (pour les mini-tables au niveau section), et `\doparttoc`, `\dopartlof` et `\dopartlot` (pour les mini-tables au niveau partie) sont analogues.

Pour obtenir un résultat satisfaisant (c'est-à-dire, non vide), veuillez noter que toutes ces commandes doivent *impérativement être placées avant* toute commande analogue aux commandes `\tableofcontents`, `\listoffigures` et `\listoftables`, ou leurs cousines `\fake...` !

Il est aussi *fortement* recommandé de placer ces commandes *avant* toute commande de sectionnement produisant une entrée dans la tables des matières (pour les commandes `\do...toc`), et *avant* toute commande analogue à `\caption` produisant une entrée dans la liste des figures (pour les commandes `\do...lof`) ou dans la liste des tableaux (pour les commandes `\do...lot`); sinon les mini-tables pourraient être dans le désordre.

1.2.3 Placement des mini-tables

`\mtcskip` `\mtcskipamount` `\bigskipamount`

La commande `\mtcskip` peut être utilisée pour ajouter un saut vertical entre deux mini-tables. Sa hauteur est `\mtcskipamount` (égale à `\bigskipamount` par défaut). `\mtcskip` élimine tout saut vertical qui précède immédiatement, afin de ne pas accumuler des espaces verticaux lorsqu'une mini-table est vide et donc ignorée par l'option `checkfiles`.

`\secttoc` `\section`

La table des matières au niveau section sera dans la section, après la commande `\section`, à l'emplacement de la commande `\secttoc`. La commande `\secttoc` peut être placée *presque n'importe où* à l'intérieur d'une section. Bien sûr, il est préférable de la placer au début de la section, ou éventuellement après un texte introductif. Vous devriez utiliser les mêmes conventions dans toutes les sections. Si vous souhaitez ajouter la table des matières au niveau section pour une section, vous devez utiliser la séquence donnée dans le tableau 1.2 page suivante. Pour chaque table des matières au niveau section, un fichier auxiliaire sera créé avec un nom de la forme ⟨*document*⟩`.stc`⟨*N*⟩, où ⟨*N*⟩ est le numéro absolu de section. Le suffixe est `.slf`⟨*N*⟩ pour les listes de figures au

[6]Le code de ces commandes `\do...` est directement dérivé de celui du paquetage `xr` [12], de David Carlisle, avec sa permission.

TAB. 1.2 – Commandes pour une `secttoc`

```
\documentclass[...]{article}
\usepackage[...options...]{minitoc}
...
\setlength{\stcindent}{24pt}   défaut
...
\setcounter{secttocdepth}{2}   défaut
\renewcommand{\stcfont}{\small\rmfamily\upshape\mdseries}          défaut
\renewcommand{\stcSSfont}{\small\rmfamily\upshape\bfseries}        défaut
ou :
\mtcsetdepth{secttoc}{2}                                           défaut
\mtcsetfont{secttoc}{*}{\small\rmfamily\upshape\mdseries}          défaut
\mtcsetfont{secttoc}{subsection}{\small\rmfamily\upshape\bfseries}   défaut
...
\begin{document}
...
\dosecttoc
\dosectlof
\dosectlot
\tableofcontents               ou \faketableofcontents
\listoffigures                 ou \fakelistoffigures
\listoftables                  ou \fakelistoftables
...
\section{...}
\secttoc                       si vous en voulez une
\sectlof                       si vous en voulez une
\sectlot                       si vous en voulez une
...
```

niveau section et `.slt`⟨*N*⟩ pour les listes de tableaux au niveau section. (Si vous travaillez sous MS-DOS ou tout autre système d'exploitation utilisant des suffixes courts pour les noms de fichiers, voyez la section 1.6 page 39 et la section 2.5 page 43).

`\usepackage`
`\FloatBarrier`

Comme les éléments flottants (figures et tableaux) pourraient dériver quelque part en dehors de la zone d'impression du texte de la section, les sectlofs et sectlots peuvent être assez étranges. Afin d'avoir un meilleur comportement de ces mini-tables, il peut être utile d'ajouter l'option `insection` dans la commande `\usepackage` :

```
\usepackage[insection]{minitoc}
```

si vous désirez des sectlofs et sectlots plus cohérentes. Il peut même être nécessaire d'utiliser la commande `\FloatBarrier` de ce paquetage pour placer correctement la figure ou la table et avoir une mini-table correcte. L'option `insection` charge le paquetage `placeins` [3] avec ses options `verbose`, `section` et `bottom`. Le paquetage `placeins`, de Donald Arsenau, est disponible sur les archives CTAN ; `placeins.sty` contient sa propre documentation, dont une copie est dans `placeins.txt`. Il vous faut une version plus récente que 2005/04/18.

Si vous voulez ajouter une table des matières partielle pour une partie, vous devez utiliser la séquence donnée dans le tableau 1.3 page suivante. Pour chaque table des matières au niveau partie, un fichier auxiliaire sera créé avec un nom de la forme ⟨*document*⟩`.ptc`⟨*N*⟩, où ⟨*N*⟩ est le numéro absolu de partie. Le suffixe est `.plf`⟨*N*⟩ pour les listes de figures au

TAB. 1.3 – Commandes pour une `parttoc`

```
\documentclass[...]{book}
\usepackage[...options...]{minitoc}
...
\setlength{\ptcindent}{0pt}    défaut
...
\setcounter{parttocdepth}{2}   défaut
\renewcommand{\ptcfont}{\normalsize\rmfamily\upshape\mdseries}        défaut
\renewcommand{\ptcCfont}{\normalsize\rmfamily\upshape\bfseries}       défaut
\renewcommand{\ptcSfont}{\normalsize\rmfamily\upshape\mdseries}       défaut
ou :
\mtcsetdepth{parttoc}{2}                                              défaut
\mtcsetfont{parttoc}{*}{\normalsize\rmfamily\upshape\mdseries}        défaut
\mtcsetfont{parttoc}{chapter}{\normalsize\rmfamily\upshape\bfseries}  défaut
\mtcsetfont{parttoc}{section}{\normalsize\rmfamily\upshape\mdseries}  défaut
...
\begin{document}
...
\doparttoc
\dopartlof
\dopartlot
\tableofcontents               ou \faketableofcontents
\listoffigures                 ou \fakelistoffigures
\listoftables                  ou \fakelistoftables
...
\part{...}
\parttoc                       si vous en voulez une
\partlof                       si vous en voulez une
\partlot                       si vous en voulez une
...
```

niveau partie et `.plt`⟨*N*⟩ pour les listes de tableaux au niveau partie. (Si vous travaillez sous MS-DOS ou tout autre système d'exploitation utilisant des suffixes courts pour les noms de fichiers, voyez la section 1.6 page 39 et la section 2.5 page 43).

Note : l'utilisateur est responsable de demander ou non une mini-table (mini-lof, -lof ou -lof) pour un certain chapitre. Demander une minilof pour un chapitre sans aucune figure aura pour résultat une mini-liste des figures vide et laide (c'est-à-dire le titre et deux filets horizontaux). Il est aussi responsable de demander ou non une table des matières partielle (ou une liste partielle des figures ou des tableaux) pour une certaine partie. Demander une partlot pour une partie sans aucune figure aura pour résultat une liste partielle des figures vide et laide (c'est-à-dire le titre seul sur une page). Des remarques analogues s'appliquent aux mini-tables au niveau section (secttoc, sectlof et sectlot) et aux mini-tables au niveau partie (parttoc, partlof et partlot).

Mais depuis la version #35, les mini-tables vides sont simplement ignorées et ce problème devrait disparaître dans des circonstances normales. Néanmoins, il est recommandé de ne pas mettre de commande `\minitoc` dans un chapitre sans sections ni de commande `\minilof` ou `\minilot` dans un chapitre sans figures ou sans tableaux. L'option de paquetage `checkfiles` (see section 1.2.3 page 20), active par défaut, saute les mini-tables vides (avec un note dans le fichier `.log`) ; l'option de paquetage `nocheckfiles` rétablit l'ancien comportement (les mini-tables vides sont imprimées).

Par défaut, les mini-tables et tables des matières partielles contiennent seulement les références jusqu'aux sections et sous-sections. Les compteurs `parttocdepth`, `minitocdepth` et `secttocdepth`, similaires à `tocdepth`, permettent à l'utilisateur de modifier ce comportement. Les mini-listes des figures ou des tableaux ne sont pas affectées par ces compteurs, mais s'il existe des compteurs de profondeurs pour ces listes (`lofdepth` et `lotdepth`), comme cela est fait par le paquetage `subfig` [13] de Steven Douglas Cochran, de nouveaux compteurs sont créés si nécessaires, portant des noms évidents tels que `partlofdepth`, `partlotdepth`, `minilofdepth`, `minilotdepth`, `sectlofdepth` et `sectlotdepth`.

1.2.4 Chapitres, parties et sections étoilés

`\addstarredpart`
`\addstarredchapter`
`\addstarredsection`

NOTE : si vous utilisez `\chapter*` et une commande

```
\addcontentsline{toc}{chapter}{...}
```

pour ajouter quelque chose dans la table des matières, la numérotation des fichiers auxiliaires de `minitoc` serait altérée. Pour éviter ce problème, dites :

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

`\addcontentsline`

Ces commandes ne s'appliquent que pour le niveau d'une part-, mini- ou sect-toc ; pour des niveaux inférieurs, utilisez, comme d'habitude :

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

par exemple, pour ajouter une entrée de niveau section dans la table des matières globale et dans la minitoc d'un chapitre étoilé :

```
\chapter*{Titre du chapitre}
\addstarredchapter{Titre du chapitre}
\minitoc
\section*{Première section}
\addcontentsline{toc}{section}{Première section}
\section*{Seconde section}
\addcontentsline{toc}{section}{Seconde section}
```

`\adjustptc`
`\adjustmtc`
`\adjuststc`

Il y a parfois un problème avec des mini-tables lorsque vous utilisez `\chapter*` (ou `\section*`) : les minitocs apparaissent dans le mauvais chapitre. Vous pouvez ajouter une commande `\adjustmtc` (ou `\adjuststc`, ou `\adjustptc`) à la fin du chapitre étoilé (ou de la section ou partie étoilée) pour incrémenter le compteur correspondant. N'utilisez pas de commandes telles que `\stepcounter{mtc}` ou `\addtocounter{mtc}{...}` (ceci devrait marcher, mais c'est de la triche), parce que le paquetage `mtcoff` (voir section 1.8 page 40) sait quoi faire avec `\adjustmtc` (et autres), mais ne sait pas quoi faire de `\stepcounter` ou `\addtocounter`, car ce sont des commandes LaTeX basiques standard, pas des commandes spécifiques de `minitoc`. Syntaxe :

TAB. 1.4 – Commandes pour ajouter une entrée dans la table des matières pour une partie, un chapitre ou une section étoilés.

Niveau	Avec titre
partie	\mtcaddpart[⟨*titre*⟩]
chapitre	\mtcaddchapter[⟨*titre*⟩]
section	\mtcaddsection[⟨*titre*⟩]

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

où *n* est l'incrément (défaut : 1).

\decrementptc \decrementmtc \decrementstc \incrementptc \incrementmtc \incrementstc

Il y a des commandes similaires pour *décrémenter* ou *incrémenter* de 1 ces compteurs : \decrementptc, \decrementmtc, \decrementstc, \incrementptc, \incrementmtc et \incrementstc; les mêmes remarques que ci-dessus s'appliquent.

\mtcaddpart \mtcaddchapter \mtcaddsection

Une manière plus élégante de résoudre ce problème est d'utiliser des commandes similaires à :

\mtcaddchapter[⟨*titre*⟩]

Cette commande ajoute une entrée dans la table des matières (et ajuste le compteur, car elle appelle \adjustmtc). Le tableau 1.4 liste ces commandes, que vous placez *après* \chapter*, etc. Si l'argument optionnel est omis, ou est vide ou blanc, aucune entrée ne sera visible dans la table des matières ni dans les minitocs. Si l'argument optionnel est quelque chose d'invisible (comme ~, \space ou \quad), le résultat sera étrange mais cependant logiquement correct.

1.3 Composition des mini-tables

Les mini-tables sont composées dans un environnement analogue à verse, et donc peuvent s'étendre sur plusieurs pages.

1.3.1 Mini-tables au niveau chapitre

\mtcfont \mtcSfont \mtcSSfont \mtcSSSfont \mtcPfont \mtcSPfont \mlffont \mltfont

La mini-table des matières est composée dans la fonte \mtcfont, qui est \small\rmfamily par défaut. Les entrées de type section sont composées dans la fonte \mtcSfont, qui est \small\bfseries par défaut. Pour les sous-sections, sous-sous-sections, paragraphes et sous-paragraphes, les commandes \mtcSSfont, \mtcSSSfont, \mtcPfont et \mtcSPfont sont disponibles (par défaut, \small\rmfamily) pour permettre l'utilisation de fontes diverses. Les mini-listes de figures et de tableaux sont

TAB. 1.5 – Fontes et titres pour les commandes de mini-tables.

Commande	Fonte choix par défaut	Chaîne titre choix par défaut	Fonte du titre choix par défaut
Pour les commandes `\part...` :			
`\parttoc`	`\ptcfont`	`\ptctitle`	`\ptifont`
	`\normalsize\rmfamily`[*]	Table of Contents[†]	`\LARGE\bfseries`[*]
	`\small\rmfamily`[**]		`\Large\bfseries`[**]
`\partlof`	`\plffont`	`\plftitle`	`\ptifont`
	`\normalsize\rmfamily`[*]	List of Figures[†]	`\LARGE\bfseries`[*]
	`\small\rmfamily`[**]		`\Large\bfseries`[**]
`\partlot`	`\pltfont`	`\plttitle`	`\ptifont`
	`\normalsize\rmfamily`[*]	List of Tables[†]	`\LARGE\bfseries`[*]
	`\small\rmfamily`[**]		`\Large\bfseries`[**]
Pour les commandes `\mini...` :[*]			
`\minitoc`	`\mtcfont`	`\mtctitle`	`\mtifont`
	`\small\rmfamily`	Contents[†]	`\large\bfseries`
`\minilof`	`\mlffont`	`\mlftitle`	`\mtifont`
	`\small\rmfamily`	Figures[†]	`\large\bfseries`
`\minilot`	`\mltfont`	`\plttitle`	`\mtifont`
	`\small\rmfamily`	Tables[†]	`\large\bfseries`
Pour les commandes `\sect...` :[**]			
`\secttoc`	`\stcfont`	`\stctitle`	`\stifont`
	`\small\rmfamily`	Contents[†]	`\Large\bfseries`
`\sectlof`	`\slffont`	`\mlftitle`	`\stifont`
	`\small\rmfamily`	Figures[†]	`\Large\bfseries`
`\sectlot`	`\sltfont`	`\plttitle`	`\stifont`
	`\small\rmfamily`	Tables[†]	`\Large\bfseries`

[*] pour les classes de document avec le niveau `\chapter` (par ex. `book`, `report`).
[**] pour les classes de document sans le niveau `\chapter` (par ex. `article`).
[†] par défaut pour l'anglais ; changé par les fichiers de définition de langage ou `\renewcommand` .
Toutes ces fontes utilisent `\rmfamily`, `\upshape`et `\mdseries` par défaut.

composées dans les fontes `\mlffont` et `\mltfont`, qui sont `\small\rmfamily` par défaut. Les tableaux 1.5 et 1.6 page suivante listent ces nombreuses commandes[7].

Notez que le choix de fontes par défaut n'est certainement pas parfait et donc n'est pas définitif. Un symptôme de cette imperfection est la présence d'alignements imprécis dans les mini-tables, si des fontes grasses et non-grasses sont mélangées[8] (la longueur effective de `1em` n'est pas la même pour ces fontes). Ceci peut éventuellement être corrigé en changeant quelques fontes.

1.3.2 Titres pour les mini-tables au niveau chapitre

`\mtifont` `\mtctitle` `\mlftitle` `\mlttitle` `\mtcsettitle` `\mtcselectlanguage`

Les titres sont composés dans la fonte `\mtifont` (`\large\bfseries` par défaut) et les textes des titres sont définis par `\mtctitle`, `\mlftitle` et `\mlttitle`, qui sont

[7] Merci à Stefan ULRICH, qui a fourni ces tableaux initialement.

[8] Ceci apparaît notamment si vous utilisez les fontes Computer Modern Roman (CMR). Le symptôme disparaît si vous n'utilisez plus de fontes CMR grasses ou si vous utilisez les fontes TX (paquetage `txfonts` [40]), par exemple, comme dans ce document. Voir aussi la section 2.29 page 51.

TAB. 1.6 – Fontes pour les entrées des mini-tables.

Niveau	Fonte	Choix par défaut
Pour les entrées d'une \parttoc :		
Chapitre*	\ptcCfont*	\normalsize\bfseries*
Section	\ptcSfont	\normalsize\rmfamily*
		\small\bfseries**
Sous-section	\ptcSSfont	*(comme \ptcfont)*
Sous-sous-section	\ptcSSfont	*(comme \ptcfont)*
Paragraphe	\ptcPfont	*(comme \ptcfont)*
Sous-paragraphe	\ptcSPfont	*(comme \ptcfont)*
Pour les entrées d'une \minitoc :*		
Section	\mtcSfont	\small\bfseries
Sous-section	\mtcSSfont	*(comme \mtcfont)*
Sous-sous-section	\mtcSSfont	*(comme \mtcfont)*
Paragraphe	\mtcPfont	*(comme \mtcfont)*
Sous-paragraphe	\mtcSPfont	*(comme \mtcfont)*
Pour les entrées d'une \secttoc :**		
Sous-section	\stcSSfont	\normalsize\bfseries
Sous-sous-section	\stcSSfont	*(comme \stcfont)*
Paragraphe	\stcPfont	*(comme \stcfont)*
Sous-paragraphe	\stcSPfont	*(comme \stcfont)*

*pour les classes de document avec le niveau \chapter (par ex. book, report).
**pour les classes de document sans le niveau \chapter (par ex. article).

les chaînes « Contents », « Figures » et « Tables » par défaut. Ces titres devraient être redéfinis par \renewcommand or \mtcsettitle pour les langues autres que l'anglais. Les fichiers d'option de langue tels que french.mld et english.mld (le suffixe .mld signifie « *minitoc language definition (file)* ») (et d'autres, voir le tableau 1.7 page suivante et la section 1.3.12 page 31) sont disponibles. Vous pouvez facilement préparer un fichier similaire pour votre langue préférée (voir la section 2.26 page 50). Vous pouvez changer la langue de ces titres en utilisant la macro \mtcselectlanguage{*langue*}.

1.3.3 Mini-tables au niveau partie

\ptcfont
\ptcCfont
\ptcSfont
\ptcSSfont
\ptcSSSfont
\ptcPfont
\ptcSPfont
\plffont
\pltfont

La table des matières au niveau partie est composée dans la fonte \ptcfont, qui est définie comme étant \normalsize\rmfamily par défaut. Les entrées de type chapitre sont composées dans la fonte \ptcCfont, qui est \normalsize\bfseries par défaut. Les entrées de type section sont composées dans la fonte \ptcSfont, qui est \normalsize\rmfamily par défaut. Pour les sous-sections, sous-sous-sections, paragraphes et sous paragraphes, les commandes \ptcSSfont, \ptcSSSfont, \ptcPfont et \ptcSPfont sont disponibles (par défaut, \normalsize\rmfamily) si vous souhaitez utiliser diverses fontes. Les listes de figures et de tableaux au

TAB. 1.7 – Langues disponibles

1. afrikaan (afrikaans)
2. arab (arabic) [a,c]
3. armenian [c]
4. bahasa [c]
5. bangla [c]
6. basque
7. bicig [c]
8. brazil (brazilian)
9. breton
10. bulgarian [c]
11. bulgarianb [c]
12. buryat [c]
13. catalan
14. chinese1 [c,g]
15. chinese2 [c,g]
16. croatian
17. czech
18. danish
19. dutch
20. english (american, british, canadian, UKenglish, USenglish)
21. esperant (esperanto)
22. estonian
23. ethiopia (ethiopian) [c]
24. farsi1 [c,f,g]
25. farsi2 [c,f,g]
26. finnish
27. finnish2
28. french (frenchb, frenchle, frenchpro, francais, acadien, canadien)
29. galician
30. german (austrian)
31. germanb
32. greek [c]
33. greek-mono [c,e]
34. greek-polydemo [c,e]
35. greek-polykatha [c,e]
36. guarani [h]
37. hangul1 [c,d,g]
38. hangul2 [c,d,g]
39. hangul3 [c,d,g]
40. hangul4 [c,d,g]
41. hanja1 [c,d,g]
42. hanja2 [c,d,g]
43. hebrew [c,d]
44. icelandic [f]
45. interlingua
46. irish
47. italian
48. japanese [c,d,g]
49. japanese2 [c,d,g]
50. japanese3 [c,d,g]
51. japanese4 [c,d,g]
52. japanese5 [c,d,g]
53. latin
54. latin2
55. latvian (letton) [e]
56. lithuanian
57. lsorbian
58. magyar (hungarian)
59. magyar2
60. malayalam-keli [c]
61. malayalam-rachana [c]
62. malayalam-rachana2 [c]
63. mongol [c]
64. ngermanb (ngerman, naustrian)
65. norsk
66. nynorsk
67. polish
68. polish2 [c,e]
69. portuguese (portuges)
70. romanian
71. russian [b,c]
72. russianb [b,c]
73. russianc [b,c]
74. russian2m [c,e]
75. russian2o [c,e]
76. samin
77. scottish
78. serbian
79. serbianc [c]
80. slovak
81. slovene
82. spanish (castillan, castillian)
83. spanish2
84. spanish3 [e,f]
85. swedish
86. thai [c,d,f,g]
87. turkish
88. ukraineb [b,c]
89. usorbian
90. vietnam (vietnamese) [c,d]
91. welsh

[a] La langue « `arab(ic)` » requiert l'utilisation du paquetage ArabTEX [25, 26].
[b] La langue « `russian` » n'est pas encore supportée par `babel` [10], mais « `russianb` » est supportée si vous utilisez babel-3.6 ou une version supérieure ; « `russianc` » est un supplément.
[c] Certaines langues peut requérir des fontes spécifiques.
[d] Requiert le paquetage CJK [29].
[e] Requiert Lambda, la version de LATEX pour Omega.
[f] Requiert un codage en entrée sur 8 bits.
[g] Utilise aussi un fichier `.mlo`.
[h] Requiert un codage en entrée spécifique.

niveau partie sont composées dans les fontes `\plffont` et `\pltfont`, qui sont `\normalsize\rmfamily` par défaut.

1.3.4 Titres pour les mini-tables au niveau partie

\ptifont \ptctitle \plftitle \plttitle \mtcsettitle \mtcselectlanguage

Les titres sont composés dans la fonte \ptifont (\LARGE\bfseries par défaut) et les textes des titres sont définis par \ptctitle, \plftitle et \plttitle, qui sont les chaînes « Table of Contents », « List of Figures » et « List of Tables » par défaut. Ces commandes de titres devraient être redéfinies par \renewcommand or \mtcsettitle pour les langues autres que l'anglais. Des fichiers de définition de langue tels que french.mld et english.mld (et de nombreux autres, voir le tableau 1.7 page précédente et la section 1.3.12 page 31) sont disponibles. Vous pouvez facilement préparer un fichier similaire pour votre langue préférée (voir la section 2.26 page 50). Vous pouvez changer la langue de ces titres en utilisant la macro \mtcselectlanguage{*langue*}.

1.3.5 Mini-tables au niveau section

\stcfont \stcSSfont \stcSSSfont \stcPfont \stcPSfont \slffont \sltfont

La table des matières au niveau section est composée dans la fonte \stcfont, qui est définie comme étant \normalsize\rmfamily par défaut. Les entrées de sous-section sont composées dans la fonte \stcSSfont, qui est \normalsize\bfseries par défaut. Les entrées de sous-sous-section sont composées dans la fonte \stcSSSfont, qui est \normalsize\rmfamily par défaut. Pour les paragraphes et sous-paragraphes, les commandes \stcPfont et \stcSPfont sont disponibles (par défaut, \normalsize\rmfamily) si vous désirez utiliser diverses fontes. Les listes de figures et de tableaux au niveau section sont composées dans les fontes \slffont et \sltfont, qui sont définies comme étant \normalsize\rmfamily par défaut.

1.3.6 Titres pour les mini-tables au niveau section

\stifont \stctitle \slftitle \slttitle \mtcsettitle \mtcselectlanguage

Les titres sont composés dans la fonte \stifont (\normalsize\bfseries par défaut) et les textes des titres sont définis par \stctitle, \slftitle et \slttitle, qui sont les chaînes « Contents », « Figures » et « Tables » par défaut. Ces commandes de titres devraient être redéfinies par \renewcommand ou \mtcsettitle pour les langues autres que l'anglais. Des fichiers de définition de langue tels que french.mld et english.mld (et de nombreux autres, voir le tableau 1.7 page précédente et la section 1.3.12 page 31) sont disponibles. Vous pouvez facilement préparer un fichier similaire pour votre langue préférée (voir la section 2.26 page 50). Vous pouvez changer la langue de ces titres en utilisant la macro \mtcselectlanguage{*langue*}.

1.3.7 Position des titres

1.3.7.1 Pour les mini-tables au niveau partie

\doparttoc \dopartlof \dopartlot \parttoc \partlof \partlot

Par défaut, les titres sont cadrés à gauche. Les commandes de préparation \doparttoc, \dopartlof et \dopartlot acceptent un argument optionnel pour changer la position par défaut du titre correspondant : [l] pour gauche (défaut), [c] pour centré, [r] pour droite, ou [e] (ou [n]) pour vide (pas de titre). La modification est globale pour le document. Si vous souhaitez changer la position du titre pour une seule parttoc (ou partlof ou partlot), il vous suffit d'utiliser un tel argument optionnel avec la commande \parttoc (ou \partlof ou \partlot).

1.3.7.2 Pour les mini-tables au niveau chapitre

\dominitoc \dominilof \dominilot \minitoc \minilof \minilot

Par défaut, les titres sont cadrés sur la gauche. Les commandes de préparation \dominitoc, \dominilof et \dominilot acceptent un argument optionnel pour changer la position par défaut du titre correspondant : [l] pour gauche (défaut), [c] pour centré, [r] pour droite, ou [e] (ou [n]) pour vide (pas de titre). La modification est globale pour le document. Si vous souhaitez changer la position du titre pour une seule minitoc (ou minilof ou minilot), il vous suffit d'utiliser un tel argument optionnel avec la commande \minitoc (ou \minilof ou \minilot).

1.3.7.3 Pour les mini-tables au niveau section

\dosecttoc \dosectlof \dosectlot \secttoc \sectlof \sectlot

Par défaut, les titres sont cadrés sur la gauche. Les commandes de préparation \dosecttoc, \dosectlof et \dosectlot acceptent un argument optionnel pour changer la position par défaut du titre correspondant : [l] pour gauche (défaut), [c] pour centré, [r] pour droite, ou [e] (ou [n]) pour vide (pas de titre). La modification est globale pour le document. Si vous souhaitez changer la position du titre pour une seule secttoc (ou sectlof ou sectlot), il vous suffit d'utiliser un tel argument optionnel avec la commande \secttoc (ou \sectlof ou \sectlot).

1.3.7.4 Résumé du positionnement des titres

\doparttoc \dopartlof \dopartlot \dominitoc \dominilof \dominilot \dosecttoc \dosectlof \dosectlot \parttoc \partlof \partlot \minitoc \minilof \minilot \secttoc \sectlof \sectlot

Pour résumer, par défaut, tous les titres sont sur la gauche. Cependant, chacune des commandes de préparation suivantes :

\doparttoc, \dopartlof, \dopartlot,
\dominitoc, \dominilof, \dominilot,
\dosecttoc, \dosectlof, \dosectlot

accepte un argument optionnel pour modifier le positionnement du titre : [l] pour gauche (défaut), [c] pour centré, [r] pour droite, [e] ou [n] pour vide (pas de titre), pour toutes les mini-tables correspondantes. Les commandes d'insertion suivantes :

\parttoc, \partlof, \partlot,
\minitoc, \minilof, \minilot,
\secttoc, \sectlof, \sectlot

acceptent les mêmes arguments optionnels, mais ces options changent le positionnement seulement pour le titre de la mini-table courante.

1.3.8 Espacement des lignes dans les mini-tables

\iftightmtc
\tightmtctrue
\tightmtcfalse

Avec les commandes `\tightmtctrue` (ou l'option de paquetage `tight`) et `\tightmtcfalse` (ou l'option de paquetage `loose`, qui est le choix par défaut), les mini-tables auront moins (*tight*) ou plus (*loose*) d'espacement entre les lignes de leurs entrées.

Mais avec les classes Koma-script (`scrartcls`, `scrbook` et `scrreprt`), il peut être nécessaire d'utiliser les options ou commandes suivantes, car il nous faut forcer à zéro `\parskip` au lieu de `\parsep` pour resserrer la mini-table. L'efficacité des options ci-dessous dépend des options données à ces classes Koma-script (options `parindent`, `parskip` et variantes).

\ifktightmtc
\ktightmtctrue
\ktightmtcfalse

Pour les classes Koma-script, avec les commandes `\ktightmtctrue` (ou l'option de paquetage `k-tight`) et `\ktightmtcfalse` (ou l'option de paquetage `k-loose`, qui est le choix par défaut), les mini-tables auront moins (*tight*) ou plus (*loose*) d'espacement entre les lignes de leurs entrées.

1.3.9 Commandes simplifiées pour les fontes

\mtcsetfont
\mtcsettitlefont

Pour simplifier la redéfinition des fontes des mini-tables, il y a deux commandes utiles :

`\mtcsetfont{`*mini-table*`}{`*niveau-sectionnement*`}{`*commandes*`}`
`\mtcsettitlefont{`*mini-table*`}{`*commandes*`}`

Par exemple :

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

redéfinira `\mtcSSfont` avec les commandes de fontes données.

Notez que `\mtcsetfont{parttoc}{*}{...}` permet aussi de redéfinir `\ptcfont`, etc.

De plus,

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

redéfinira `\ptifont` (pour les titres dans les parttocs, partlofs et partlofs) avec les commandes de fontes données.

1.3.10 Commande simplifiée pour les titres des mini-tables

\mtcsettitle

Pour simplifier la redéfinition des titres des mini-tables, la commande `\mtcsettitle` est aussi disponible :

`\mtcsettitle{`*mini-table*`}{`*chaîne titre*`}`

Par exemple,

```
\mtcsettitle{minitoc}{Description du contenu}
```

redéfinira `\mtctitle` avec la chaîne donnée. Cette commande vérifie que vous redéfinissez un titre pour un type de mini-table disponible dans votre classe de document.

1.3.11 Commande simplifiée pour les profondeurs des mini-tables

`\mtcsetdepth`

Pour simplifier la redéfinition des profondeurs des mini-tables, vous disposez de la commande `\mtcsetdepth` :

`\mtcsetdepth`{*mini-table*}{*profondeur*}

Par exemple :

```
\mtcsetdepth{minitoc}{4}
```

forcera le compteur `minitocdepth` la valeur donnée. Cette commande vérifie que vous forcez une profondeur pour un type de mini-table disponible dans votre classe de document (et qu'il est possible de changer sa profondeur).

1.3.12 Langues pour les titres

La plupart des chaînes définies dans les fichiers d'option de langue (`.mld`) ont été prises dans le superbe paquetage `babel` [10] de Johannes Braams, certaines ont été adaptées, d'autres ont été mises à disposition par d'aimables utilisateurs ou prises dans des paquetages spécifiques, tels que ArabTEX [25, 26], ArmTEX (arménien) [16], BangTEX (bangla) [36], `ethiop` [6], `guarani` [7], Malayalam [1], Montex (mongol) [14, 15], CJK (chinois, coréen-hangul/hanja, japonais, thaï) [29], FarsiTEX (farsi ou iranien) `vietnam.sty` — le latvien (letton), des variantes de grec (greek-mono, greek-polydemo, greek-polykatha), de polonais (polish2), de russe (russian2m, russian2o) et d'espagnol (spanish3) ont besoin de Lambda, c'est-à-dire de la version de LaTeX pour Omega, (voir [24]), — ou ont même été trouvées en cherchant sur le Web (`bulgarianb.mld` pour le haut bulgare cyrilique, `japanese.mld` pour le japonais, `serbianc.mld` pour le serbe cyrillique). D'autres langues sont les bienvenues. Voir le tableau 1.7 page 27.

Mais pour certaines langues orientales [9], les sources des titres utilisent quelques codages exotiques qu'il est difficile de manipuler dans un fichier `.dtx`, donc le fichier `.mld` est alors un simple emballage qui charge un fichier spécial, surnommé fichier `.mlo` [10], qui n'est pas engendré par le fichier `.dtx` dans la version actuelle du paquetage `minitoc`, mais via des environnements `filecontents` dans le fichier `minitoc.ins`, et en jouant sur le « catcode » du caractère « delete ».

1.3.13 Modifier la mise en page des mini-tables *(dispositif expérimental)*

La mise en page est décrite dans la figure 1.1 page suivante (cette figure a été adaptée de [44]), qui définit certaines commandes internes (ce ne sont pas des *dimensions*, mais des commandes LaTeX, crées par `\newcommand`, modifiables via `\renewcommand`).

[9] Principalement pour le chinois, le farsi, le coréen (hangul et hanja), le japonais et le thaï, avec leurs variantes.

[10] Le suffixe `.mlo` signifie *minitoc language object* (objet de langage pour minitoc).

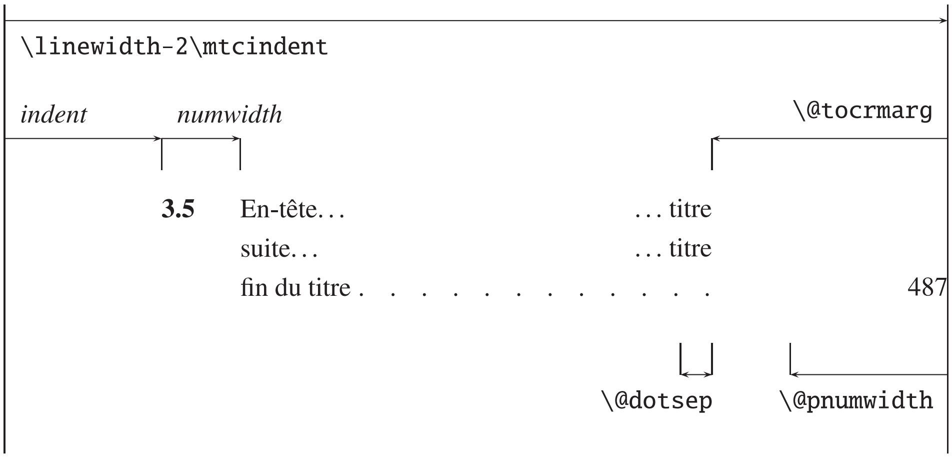


FIG. 1.1 – Mise en page d'une entrée dans une table des matières (liste des figures, liste des tableaux)

- `\@dotsep`, qui est la séparation entre les points de conduite dans une ligne de points. C'est un nombre pur exprimant des *unités mathématiques* ; 18 unités mathématiques font `1em` (un cadratin), qui est à peu près la largeur d'un « m » dans la fonte courante.
- `\@pnumwidth`, qui est la largeur de l'espace réservé pour le numéro de page. C'est une commande LaTeX contenant la représentation d'une longueur (par ex., `1.55em`).
- `\@tocrmarg`, est la distance (marge) entre le bord droit de la table et la fin de la ligne de points de conduite. Elle devrait être plus grande que `\@pnumwidth`, et peut être une longueur élastique (c'est-à-dire contenir de la glu, comme like `2.55em plus 1fil`) ; si vous spécifiez la portion « `... plus 1fil` », le texte de l'entrée sera en déchiqueté sur la droite ; ceci est utile si vous avez des entrées longues, et peut éviter la plupart des coupures de mots.

`\mtcsetformat` Puisque ces commandes sont internes (leurs noms contiennent le caractère « `@` ») et doivent avoir un effet local seulement sur des mini-tables spécifiques, vous devriez les modifier indirectement via la commande `\mtcsetformat` :

`\mtcsetformat{`*mini-table*`}{`*paramètre*`}{`*valeur*`}`

où *mini-table* est l'un des mots-clés `parttoc`, `partlof`, `partlot`, `minitoc`, `minilof`, `minilot`, `secttoc`, `sectlof` ou `sectlot` ; *paramètre* est l'un des mots-clés `dotinterval` (pour `\@dotsep`), `pagenumwidth` (pour `\@pnumwidth`) ou `tocrightmargin` (pour `\@tocrmarg`) ; ainsi :

```
\mtcsetformat{partlof}{tocrightmargin}{2.55em plus 1fil}
```

établira la marge de droite à `2.55em plus 1fil` dans la liste des tableaux au niveau partie. L'élasticité (`plus 1fil`) est utile si les captions des tableaux sont longs.

Notez que le paramètre `tocrightmargin` (pour `\@tocrmarg`) devrait manifestement être plus grand que le paramètre `pagenumwidth` (ceci est visible dans la figure 1.1).

Si le paramètre `dotinterval` (pour `\@dotsep`) est très grand (essayez 450, puis augmentez ou diminuez), les points de conduite seront tellement espacés qu'ils disparaîtront.

TAB. 1.8 – Filets horizontaux

	filets dans		pas de filets dans	défaut pour book	défaut pour report	défaut pour article
\ptcrule	parttocs	\noptcrule	parttocs	**N**	**N**	**O**
\plfrule	parttocs	\noplfrule	parttocs	**N**	**N**	**O**
\pltrule	parttocs	\nopltrule	parttocs	**N**	**N**	**O**
\mtcrule	minitocs	\nomtcrule	minitocs	**O**	**O**	**N-D**
\mlfrule	minitocs	\nomlfrule	minitocs	**O**	**O**	**N-D**
\mltrule	minitocs	\nomltrule	minitocs	**O**	**O**	**N-D**
\stcrule	secttocs	\nostcrule	secttocs	**N-D**	**N-D**	**O**
\slfrule	secttocs	\noslfrule	secttocs	**N-D**	**N-D**	**O**
\sltrule	secttocs	\nosltrule	secttocs	**N-D**	**N-D**	**O**

TAB. 1.9 – Numéros de pages

Type	**Numéros de pages (défaut)**	**Pas de numéros de pages**
parttoc	\ptcpagenumbers	\noptcpagenumbers
minitoc	\mtcpagenumbers	\nomtcpagenumbers
secttoc	\stcpagenumbers	\nostcpagenumbers
partlof	\plfpagenumbers	\noplfpagenumbers
minilof	\mlfpagenumbers	\nomlfpagenumbers
sectlof	\slfpagenumbers	\noslfpagenumbers
partlot	\pltpagenumbers	\nopltpagenumbers
minilot	\mltpagenumbers	\nomltpagenumbers
sectlot	\sltpagenumbers	\nosltpagenumbers

1.4 Dispositifs spéciaux

1.4.1 Filets horizontaux

\mtcsetrules

Par défaut, la plupart des mini-tables ont des filets horizontaux après leurs titres et à leurs fins, L'exception est la « parttoc » dans un document de type analogue à book ou report (c'est-à-dire lorsque la commande \chapter est définie). Pour activer ou désactiver ces filets, les commandes du tableau 1.8 sont disponibles. Mais vous pouvez aussi utiliser la commande suivante, plus simple :

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

où *mini-table* est l'un des mots-clés parttoc, partlof, partlot, minitoc, minilof, minilot, secttoc, sectlof ou sectlot ; les mots-clés on et off ont les synonymes suivants [11] :
– on, ON, yes, YES, y, Y, true, TRUE, t, T, vrai, VRAI, v, V, oui, OUI, o, O et 1 ;
– off, OFF, no, NO, n, N, false, FALSE, faux, FAUX, f, F, non, NON et 0.

[11] O et o sont la lettre O, 0 est le chiffre zéro.

1.4.2 Numéros de pages, points de conduites

`\mtcsetpagenumbers`

Par défaut, les numéros de page sont cités dans chaque minitoc, minilof, etc. Certains auteurs désirent avoir seulement les titres des sections (avec les numéros des sections), mais pas les numéros des pages. Donc les déclarations évidentes du tableau 1.9 page précédente sont disponibles. Mais vous pouvez aussi utiliser la commande suivante :

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

où *mini-table* est l'un des mots-clés `parttoc`, `partlof`, `partlot`, `minitoc`, `minilof`, `minilot`, `secttoc`, `sectlof` ou `sectlot` ; les mots-clés `on` et `off` ont les synonymes suivants [12] :
– `on`, `ON`, `yes`, `YES`, `y`, `Y`, `true`, `TRUE`, `t`, `T`, `vrai`, `VRAI`, `v`, `V`, `oui`, `OUI`, `o`, `O` et `1` ;
– `off`, `OFF`, `no`, `NO`, `n`, `N`, `false`, `FALSE`, `faux`, `FAUX`, `f`, `F`, `non`, `NON` et `0`.
Dans les mini-tables, il y a des points de conduite entre les titres de sections et les numéros de pages. L'option de paquetage `undotted` supprime ces points de conduite. L'option de paquetage `dotted` est le choix par défaut. Voir aussi la section 1.3.13 page 31.

1.4.3 Dispositifs pour les parttocs et autres mini-tables

Par défaut, une parttoc (ou une partlof ou une partlot) est précédée et suivie par une `\cleardoublepage`, et a un style de page `empty`. Depuis la version #32, vous pouvez modifier ce comportement en redéfinissant les commandes du tableau 1.10 page suivante, dont la signification est évidente.

`\mtcsetfeature`

La commande :

`\mtcsetfeature{`*mini-table*`}{`*mot-clé*`}{`*commandes*`}`

vous permet de redéfinir chacune de ces commandes. *mini-table* est l'un des noms de type de mini-table : `parttoc`... `sectlot`. *mot-clé* est l'un des suivants : `before`, `after` ou `pagestyle`. *commandes* est soit une séquence de commandes telles que `\clearpage`, `\cleardoublepage`, `\thispagestyle{...}`, etc., soit `\empty` (ne fait rien).

1.4.3.1 Remarque sur les styles de page [13]

Les commandes par défaut pour les styles de page des mini-tables au niveau partie sont définies comme étant `\thispagestyle{empty}`, car dans les classes de document définissant la commande `\chapter`, les mini-tables au niveau parties sont sur leurs propres pages. Si le document est imprimé recto-verso, la première page est recto. Habituellement, ces pages ne sont pas numérotées et n'ont ni en-tête ni bas de page. Ce comportement provient des définitions par défaut des commandes du tableau 1.10 page suivante. Si vous désirez un comportement différent, vous pouvez changer ces définitions. Notez que, par défaut, seule la *première* page de ces mini-tables est composée dans le style de page

[12] `O` et `o` sont la lettre `O`, `0` est le chiffre zéro.

[13] Cette remarque est tirée et adaptée d'une version préliminaire de la seconde édition du JMPL [4], par Benjamin Bayart, où il commente le paquetage `minitoc`.

TAB. 1.10 – Dispositifs pour parttocs et autres mini-tables

Type	**Commande**	**Défaut**
parttoc	`\beforeparttoc`	`\cleardoublepage`
parttoc	`\afterparttoc`	`\cleardoublepage`
parttoc	`\thispageparttocstyle`	`\thispagestyle{empty}`
partlof	`\beforepartlof`	`\cleardoublepage`
partlof	`\afterpartlof`	`\cleardoublepage`
partlof	`\thispagepartlofstyle`	`\thispagestyle{empty}`
partlot	`\beforepartlot`	`\cleardoublepage`
partlot	`\afterpartlot`	`\cleardoublepage`
partlot	`\thispagepartlotstyle`	`\thispagestyle{empty}`
minitoc	`\beforeminitoc`	`\empty`
minitoc	`\afterminitoc`	`\empty`
minitoc	`\thispageminitocstyle`	`\empty`
minilof	`\beforeminilof`	`\empty`
minilof	`\afterminilof`	`\empty`
minilof	`\thispageminilofstyle`	`\empty`
minilot	`\beforeminilot`	`\empty`
minilot	`\afterminilot`	`\empty`
minilot	`\thispageminilotstyle`	`\empty`
secttoc	`\beforesecttoc`	`\empty`
secttoc	`\aftersecttoc`	`\empty`
secttoc	`\thispagesecttocstyle`	`\empty`
sectlof	`\beforesectlof`	`\empty`
sectlof	`\aftersectlof`	`\empty`
sectlof	`\thispagesectlofstyle`	`\empty`
sectlot	`\beforesectlot`	`\empty`
sectlot	`\aftersectlot`	`\empty`
sectlot	`\thispagesectlotstyle`	`\empty`

`\mtcsetfeature{`*mini-table*`}{before|after|pagestyle}{`*commande*`}`
Modifie les dispositifs pour une mini-table.

`empty`. Vous pouvez établir le style de cette première page en utilisant `\thispagestyle` et établir le style des pages suivantes en utilisant `\pagestyle`, mais vous ne devez pas oublier de rétablir le style normal après la mini-table. Regardez ce court exemple [14] :

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

où nous ajoutons une commande `\cleardoublepage` avant chaque parttoc, puis nous établissons le style de page `empty` pour la première page des parttocs, le style de page `myheadings` pour les pages suivantes des parttocs, et établissons le style de page `headings` pour les pages après la mini-table, après une commande `\cleardoublepage`.

1.4.4 Le problème du « Chapitre 0 » (résolu)

Certains documents ne commencent pas par le chapitre numéro un, mais par le chapitre numéro zéro (ou même portant un numéro plus bizarre).

1.4.4.1 Avant la version #23 (1994/11/08)

`\firstpartis`
`\firstchapteris`
`\firstsectionis`

Pour que le paquetage `minitoc` fonctionne avec de tels documents, vous devez insérer la commande :

`\firstchapteris{`⟨*N*⟩`}`

avant la commande `\dominitoc` et les commandes analogues. ⟨*N*⟩ est le numéro du premier chapitre. Cette commande *ne modifie pas* la numérotation des chapitres, vous devez utiliser une commande `\addtocounter{chapter}{-1}` pour obtenir un premier chapitre numéroté 0. Les commandes `\firstpartis` et `\firstsectionis` sont similaires pour des parties et sections avec une numérotation non standard.

1.4.4.2 Depuis la version #23 (1994/11/08)

Ces commandes sont désormais obsolètes, car ce problème a été résolu (via la numérotation « absolue » des fichiers auxiliaires pour les mini-tables). Elles produisent seulement des avertissements bénins.

[14] Cet exemple montre que le troisième argument peut être une *séquence* de commandes : nous établissons le style de la page courante et le style des pages suivantes.

1.4.5 Entrées spéciales pour la table des matières, la liste des figures, la liste des tableaux, la bibliographie et l'index

Si vous souhaitez ajouter des entrées dans la table des matières pour des objets tels que la table de matières elle-même, la liste des figures, la liste de tableaux, la bibliographie ou l'index, vous devriez utiliser le paquetage `tocbibind` [47] de Peter R. Wilson (ce paquetage est disponible sur les archives CTAN).

`\dominitoc`

Mais ces entrées sont considérées comme des chapitres (ou des sections dans un document de classe article) lorsque le fichier est analysé pour préparer les minitocs (la phase `\dominitoc`).

`\mtcaddchapter` `\tableofcontents` `\listoffigures` `\listoftables`

Donc vous devez ajouter une commande `\mtcaddchapter`, *sans argument*, après chacune des commandes `\tableofcontents`, `\listoffigures` et `\listoftables` mises en jeu.

`\adjustmtc` `\bibliography`

Pour la bibliographie, vous devriez ajouter une commande `\adjustmtc` après la commande `\bibliography`.

`\printindex` `\addcontentsline` `\mtcaddchapter` `\mtcfixindex`

Pour l'index, c'est un peu plus compliqué ; vous ajoutez les commandes suivantes juste après la commande `\printindex` :

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

Mais ceci peut être remplacé par :

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

où l'argument optionnel est le niveau de l'entrée de l'index dans la table des matières. Par défaut, si `\chapter` est définie, le niveau `chapter` est utilisé, sinon le niveau `section`. Si ni `\chapter` ni `\section` ne sont définies, le niveau `part` sera utilisé si `\part` est définie ; sinon une erreur est signalée. Il est *recommandé* que vous vérifiez le résultat et, si nécessaire, que vous ajustiez l'argument optionnel.

`\printglossary` `\addcontentsline` `\mtcaddchapter` `\mtcfixglossary`

Pour le glossaire, c'est comme pour l'index : vous ajoutez les commandes suivantes juste après la commande `\printglossary` :

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

Mais ceci peut être remplacé par :

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

où l'argument optionnel est le niveau de l'entrée du glossaire dans la table des matières. Par défaut, si `\chapter` est définie, le niveau `chapter` est utilisé, sinon le niveau `section`. Si ni `\chapter` ni `\section` ne sont définies, le niveau `part` sera utilisé si `\part` est définie ; sinon une erreur est signalée. Il est *recommandé* que vous vérifiez le résultat et, si nécessaire, que vous ajustiez l'argument optionnel.

Bien sûr, les documents dans lesquels la table des matières, la liste des figures, la liste des tableaux, la bibliographie et/ou l'index (ou le glossaire) sont traités comme des sections étoilées, vous devez modifier ces additions pour utiliser des commandes au niveau section.

Et procédez avec soin, en suivant dans le fichier `.log` l'insertion des fichiers `.mtc`⟨*N*⟩ (et analogues). Il y a quelques exemples dans le fichier `add.tex` distribué avec `minitoc`.

1.4.6 L'option notoccite

\cite

Cette option charge le paquetage `notoccite` [2] (de Donald ARSENEAU). Elle évite des problèmes avec des commandes `\cite` situées dans des commandes de sectionnement ou des captions : si vous exécutez alors BIBTEX en utilisant le style `unsrt` (non trié), ou un style similaire, ces citations seront numérotées en commençant à la page de la table des matières où se trouve la citation parasite, et non avec le numéro qu'elles auraient dans le texte principal. Le paquetage `notoccite` évite ceci. Comme `minitoc` imprime des tables des matières, il est sujet au même problème.

1.5 L'option hints

Cette option du paquetage détecte certaines actions et le chargement de certains paquetages et classes connus pour interagir avec `minitoc`, ainsi que certaines mauvaises utilisations et erreurs fréquentes. La liste des paquetages et classes qui interagissent n'est évidemment pas close. Si un tel paquetage (ou classe) est chargé, cette option écrit quelques indications (*hints*) dans le fichier `.log` et émet un avertissement. Les indications écrites dans le fichier `.log` peuvent vous suggérer de consulter le présent document ou le fichier `minitoc.bug`. *Votre avis sur cette option sera le bienvenu.* Cette option est activée par défaut. Vous pouvez l'inhiber via l'option `nohints`. Les problèmes et problèmes potentiels suivants sont actuellement détectés :

\part \@part \spart \@spart \chapter \@chapter \schapter \@schapter \section \@sect \@ssect

– L'altération de certaines des commandes suivantes [15] : `\part`, `\@part`, `\@spart`, `\chapter`, `\@chapter`, `\@schapter`, `\section`, `\@sect` et `\@ssect`. Notez que le paquetage `hyperref` (see section 2.17 page 47) altère ces commandes lors de `\begin{document}`, donc de problème pourrait être signalé si vous utilisez ce paquetage, mais ces altérations semblent bénignes.
– Présence des paquetages et classes suivants, qui demandent certaines précautions : `amsbook` (classe), `appendix`, `placeins` (attention à ses options et à sa date (au moins 2005/04/18)), `tocbibind` et `tocloft`.
– Présence des paquetages ou classes suivants, qui sont malheureusement incompatibles avec le paquetage `minitoc` : `amsart` (classe), `amsproc` (classe), `titlesec` et `titletoc` [16].

[15] Les commandes contenant le caractère « `@` » dans leurs noms sont des commandes internes de LATEX, d'un paquetage ou d'une classe ; elles sont parfois altérées par d'autres paquetages ; reconsidérez l'ordre de chargement des paquetages.

[16] Le paquetage `titlesec` redéfinit les commandes de sectionnement d'une manière complètement étrangère à la manière standard de LATEX ; donc `minitoc` et `titlesec-titletoc` sont fondamentalement incompatibles, et ceci est fort regrettable.

\parttoc \doparttoc \sectlot \dosectlot \tableofcontents \listoftables \faketableofcontents \fakelistoftables \sectlof \sectlot

– Utilisation de \parttoc sans appeler \doparttoc, ... , utilisation de \sectlot sans appeler \dosectlot. Ou l'inverse.
– Utilisation de \parttoc sans appeler \[fake]tableofcontents, ... , utilisation de \sectlot sans appeler \[fake]listoftables.
– Utilisation de \sectlof et/ou \sectlot sans utiliser l'option de paquetage insection de minitoc (ou le paquetage placeins sans ses options section et below).
– Si vous utilisez des suffixes courts (à cause de votre système d'exploitation ou de l'option de paquetage shortext, voyez à la section 1.6) et que vous allez au-delà de la limite de 99 parties, chapitres ou sections, l'option de paquetage signale un avertissement.

abstract \mtcaddchapter \mtcaddsection

– Si le paquetage abstract [45] (de Peter R. Wilson), est utilisé avec son option addtotoc, une entrée « Abstract » (ou équivalente) est ajoutée à la table des matières, comme un chapitre étoilé si la classe de document définit \chapter, sinon comme une section étoilée. Ceci est détecté et vous devriez ajouter une commande \mtcaddchapter[] ou \mtcaddsection[] après votre environnement abstract.
– Si le paquetage sectsty [32] (de Rowland McDonnell) est utilisé, il doit être chargé *avant* le paquetage minitoc.
– Si vous essayez d'insérer des mini-tables vides, l'option hints vous donne un avertissement global (sauf si vous avez aussi utilisé l'option nocheckfiles, voyez la section 1.2.3 page 20).

\firstpartis \firstchapteris \firstsectionis

– Si vous utilisez l'une des commandes obsolètes (\firstpartis, \firstchapteris ou \firstsectionis), un avertissement est émis pour chaque utilisation, bien sûr, mais aussi une indication (*hint*) globale comme rappel.

1.6 Utilisation avec MS-DOS

Sous MS-DOS (et autres vieux systèmes d'exploitations pour PC), les suffixes des noms de fichiers sont limités à 3 caractères. Le paquetage minitoc détermine dynamiquement le type de suffixes disponible et l'utilisera. Toutes les autres modifications seront faites automatiquement. Les suffixes .mtc$\langle N\rangle$ deviendront .M$\langle N\rangle$, où $\langle N\rangle$ est le numéro absolu de chapitre. Les suffixes .mlf$\langle N\rangle$ et .mlt$\langle N\rangle$ deviennent .F$\langle N\rangle$ et .T$\langle N\rangle$. Les suffixes .ptc$\langle N\rangle$ deviennent .P$\langle N\rangle$, où $\langle N\rangle$ est le numéro absolu de partie. Les suffixes .plf$\langle N\rangle$ et .plt$\langle N\rangle$ deviennent .G$\langle N\rangle$ et .U$\langle N\rangle$. Les suffixes .stc$\langle N\rangle$ deviennent .S$\langle N\rangle$, où $\langle N\rangle$ est le numéro absolu de section. Les suffixes .slf$\langle N\rangle$ et .slt$\langle N\rangle$ deviennent .H$\langle N\rangle$ et .V$\langle N\rangle$. Tous ces suffixes sont listés dans le tableau 1.11 page suivante. Bien sûr, ceci implique une limite de 99 chapitres dans un document, mais avez-vous vraiment besoin d'autant de chapitre (ou de sections dans un article) ? La limite de 99 parties ne semble pas trop grave pour la plupart des documents, mais pour les sections, ceci pourrait être tragique. L'option hints (section 1.5 page précédente) signalera de telles situations. Voir aussi la section 2.5 page 43.

TAB. 1.11 – Suffixes des fichiers auxiliaires

mini-table	**suffixes longs** (UNIX, etc.)	**suffixes courts** (MS-DOS, etc,)
parttoc	`.ptc`⟨*N*⟩	`.P`⟨*N*⟩
partlof	`.plf`⟨*N*⟩	`.G`⟨*N*⟩
partlot	`.plf`⟨*N*⟩	`.U`⟨*N*⟩
minitoc	`.mtc`⟨*N*⟩	`.M`⟨*N*⟩
minilof	`.mlf`⟨*N*⟩	`.F`⟨*N*⟩
minilot	`.mlf`⟨*N*⟩	`.T`⟨*N*⟩
secttoc	`.stc`⟨*N*⟩	`.S`⟨*N*⟩
sectlof	`.slf`⟨*N*⟩	`.H`⟨*N*⟩
sectlot	`.slf`⟨*N*⟩	`.V`⟨*N*⟩

1.7 Pourquoi plusieurs exécutions de LaTeX sont-elles nécessaires ?

Les mini-tables, aux niveaux partie, chapitre et section, occupent un certain espace sur les premières pages de chaque partie, chapitre ou section, donc la numérotation des pages est altérée. Après la première exécution de LaTeX, les mini-tables (aux niveaux partie, chapitre et section) seront vides (et en fait sautées depuis la version #35) ; après la deuxième exécution, elles apparaissent (lorsqu'elles ne sont pas vides), mais puisqu'elles modifient la numérotation des pages, les numéros des pages y sont faux ; après la troisième exécution, les mini-tables devraient être correctes (voir la figure 2.1 page 43).

1.8 Le paquetage `mtcoff`

Si un document a été préparé avec le paquetage `minitoc` package, il contient de nombreuses commandes spécifiques de `minitoc`, dont la plupart sont des commandes `\dominitoc`, `\faketableofcontents` et `\minitoc` (et leurs équivalentes pour les listes de figures et de tableaux). Si vous souhaitez composer ce document sans aucune mini-table, il vous suffit de remplacer le paquetage `minitoc` par le paquetage `mtcoff` (sans option), et toutes ces commandes seront ignorées, en écrivant éventuellement des messages d'avertissement dans le fichier `.log` file. Au moins deux exécutions de LaTeX seront nécessaires pour avoir une numérotation des pages et des références croisées correctes. Ceci nettoie aussi les fichiers `.aux`, `.toc`, `.lof` et `.lot` en y éliminant les commandes spécifiques de `minitoc` superflues.

Chapitre 2

Questions fréquemment posées

Sommaire

Figures

Voici une liste de problèmes et de questions fréquemment posées à propos du paquetage `minitoc`. Si votre version a un numéro plus petit que 43, vous êtes priés de faire une mise à jour à la version #43. Cette liste est aussi donnée dans le fichier `minitoc.bug`, sous forme de texte pur.

Si un problème survient, il est souvent avisé : a) d'utiliser l'option `hints` (voir la section 1.5 page 38), qui est activée par défaut, et b) de lire le fichier ⟨*document*⟩`.log`, qui peut contenir des messages pertinents.

2.1 Comment éviter une coupure de page près des filets avant et après la mini-table ?

`\enlargethispage`

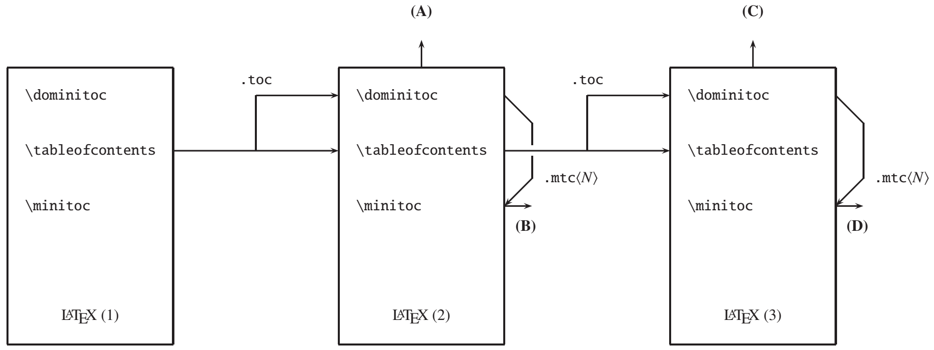
Ce problème semblait résolu depuis la version #8, mais la version #12 ajoute de meilleures corrections. Vous pouvez devoir faire quelques ajustements ultimes avec `\enlargethispage`. Voir le manuel LaTeX [27]. L'emploi du paquetage `needspace` [43] peut aussi être utile.

2.2 Comment implanter d'autres mises en page pour une mini-table ?

Les suggestions sont les bienvenues, mais voyez la section 1.3.13 page 31.

2.3 Deux contre-obliques consecutives dans une entrée de contenu provoquent une erreur

Utilisez `\protect\linebreak`.



(A) `\tableofcontents` produit une table matières, qui est probablement inexacte.
(B) `\minitoc` produit des minitocs, qui sont probablement inexactes.
(C) `\tableofcontents` produit une table matières, qui est probablement exacte.
(D) `\minitoc` produit des minitocs, qui sont probablement exactes.

FIG. 2.1 – Trois compilations pour `minitoc`

2.4 Changer l'ordre des chapitres crée le chaos

Si vous changez l'ordre des chapitres, le chaos en découle,... avec les mini-tables allant dans les mauvais chapitres.

La meilleure méthode semble d'effectuer une exécution avec le paquetage `mtcoff` remplaçant le paquetage `minitoc`, puis de rétablir le paquetage `minitoc` et de ré-exécuter LaTeX trois fois (oui, cela prend du temps...). Voir la figure 2.1. Exécuter avec le paquetage `mtcoff` garantit que les fichiers auxiliaires standard seront purgés des commandes « superflues » introduites par `minitoc`. Une solution plus radicale est de détruire les fichiers `.aux`, `.toc`, `.lof` et `.lot` relatifs au document, puis de ré-exécuter LaTeX trois fois.

2.5 Suffixes pour les noms des fichiers auxiliaires

Ce paquetage crée des fichiers auxiliaires ayant des suffixes tels que `.mtc`⟨*N*⟩. *Certains systèmes d'exploitation ne permettent que des suffixes limités à 3 caractères. Que faire ?*

Aucune modification n'est nécessaire : tout est devenu automatique depuis la version #28 ! Si vous voulez vraiment utiliser des suffixes de 3 caractères, utilisez simplement l'option de paquetage `shortext`. Alors vous recevrez d'abord les messages d'autoconfiguration, puis un message disant que vous allez utiliser des suffixes courts. Mais faites attention à ne pas avoir plus 99 mini-tables du même type (même vides) !

2.6 Jouer avec le numéro de chapitre

Ne trichez pas avec le compteur « `chapter` », c'est-à-dire, n'écrivez pas des horreurs telles que

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

Ceci casserait la mécanique du paquetage. Il est préférable d'ajouter des commandes `\chapter`, afin de créer des chapitres vides (mais numérotés d'une manière légale). Depuis la version #10, le paquetage `minitoc` fonctionne avec les appendices. La version #19 vous permet de commencer par un chapitre numéroté autrement que un. Et regardez « Entrées spéciales pour la table des matières, la liste des figures, la liste des tableaux, la bibliographie et l'index », section 1.4.5 page 37.

La même remarque s'applique aux compteurs `part` et `section`.

2.7 Classes de document supportées

Le paquetage `minitoc` est restreint aux classes de document qui définissent les chapitres de la manière standard, comme « `book` » et « `report` », ou les sections de la manière standard, comme « `article` ». Il y a des « parttocs » si la classe de document définit la commande `\part`. Notez que des classes telles que « `letter` », qui n'ont pas la structure de sectionnement classique, ne peuvent pas être supportées. Les classes utilisant des commandes de sectionnement portant d'autres noms ne sont pas supportées [1]. Voir aussi la section 2.24 page 50.

2.8 Compatibilité avec les versions de LaTeX

Certains utilisateurs ne sont pas parvenus à faire fonctionner `minitoc`. Ils ont reçu un message tel que :

```
Undefined command ... \@inputcheck ...
Your version of latex.tex is obsolete. Trying to continue...
```

ou :

```
Undefined command ... \reset@font ...
Your version of latex.tex is very obsolete.
Trying to continue... crossing fingers.
```

La commande `\reset@font` a été ajoutée à `latex.tex` le 29 septembre 1991, et la commande `\@inputcheck` le 18 mars 1992, et cette version de `latex.tex` a été distribuée le 25 mars 1992. Si vous obtenez un tel message, vous avez une vieille version de `latex.tex`. Obtenez une version récente depuis les archives et régénérez un format `latex.fmt` via `initex` (ou votre outil de configuration).

[1] Ce serait très difficile : tout utilisateur peut créer de nouvelles commandes de sectionnement (souvent à l'aide de certains paquetages) portant des noms standard ou nouveaux ; ceci n'est limité que par l'imagination. Le paquetage `minitoc` s'appuie sur les noms des commandes de sectionnement standard et sur la syntaxe de ces commandes.

2.9 Autres mini-tables

Certains utilisateurs exigeants désirent avoir des minilofs, minilots et minibbls. Tout d'abord, les « minibbls » (mini-bibliographies) sont un autre problème, fortement lié au traitement par BibTeX des fichiers `.aux`. Regardez les paquetages `chapterbib`, `bibunits`, `multibib` et `bibtopic`. La version #13 a implanté les minilofs et minilots basiques. Les minibbls ne sont pas l'objet de ce paquetage (voir `http://www.tex.ac.uk/cgi-bin/texfaq2html?label=multbib`).

2.10 Pourquoi autant de fichiers auxiliaires ?

Ce paquetage crée des tas de fichiers auxiliaires et certains utilisateurs ont argué qu'il y en avait trop. Une réorganisation profonde serait nécessaire pour éviter ce grand nombre. Utiliser un seul grand fichier auxiliaire (ou un pour toutes les minitocs, un pour toutes les minilofs , etc.) rendrait la lecture de ce fichier très lente, car il devrait être lu pour chaque commande `\miniXXX` ! De plus, ceci rendrait l'implantation de l'option `checkfiles` (section 1.2.3 page 20) pratiquement impossible. Notez que les nombreux fichiers `*.mtc*`, etc., peuvent être détruits après l'exécution de LaTeX. Ils sont reconstruits par la commande `\dominitoc` (et ses analogues). Mais, depuis la version #35, `minitoc` est capable de détecter et sauter les fichiers `*.mtc*` (et analogues) vides pour éviter des titres isolés avec seulement deux filets fins. Tout ceci ne serait pas facile à faire avec un seul gros fichier auxiliaire.

Ces fichiers contiennent les mini-tables extraites des fichiers `.toc`, `.lof` et `.lot`. Ils ne sont plus utiles après l'exécution de LaTeX. Si vous exécutez LaTeX au moyen d'un script ou d'un « makefile », il peut être utile de lui ajouter un dispositif de nettoyage (qui devrait être optionnel, pour permettre la mise au point). La table 1.11 page 40 donne la liste des suffixes pour ces fichiers (notez qu'un fichier auxiliaire ⟨*document*⟩`.mtc` est aussi créé comme fichier de travail).

Comme exemple, vous pouvez regarder le script `rubber` [5] (écrit en Python) par Emmanuel Beffara :

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

2.11 Comment faire des mini-tables à des niveaux autres que le chapitre ?

Ici aussi, une reconstruction a été nécessaire. Depuis la version #15, il y a des parttocs, partlofs et partlots pour le niveau partie dans les documents dont la classe est similaire à `book`, `report` ou `article`, des secttocs, sectlofs et sectlots pour le niveau section dans les documents dont la classe est similaire à `article`. Notez que vous ne pouvez pas avoir les dispositifs de `minitoc` aux niveaux chapitre et section dans le même document, car cela produirait un monstre illisible. L'utilisateur doit choisir la classe de base du document selon la taille de celui-ci (par exemple, ne pas écrire un article de plus de 100 sections : c'est un rapport, ou même un livre !).

	partie	**chapitre**	**section**
`book`	*	*	
`report`	*	*	
`article`	*		*

2.12 Incompatibilité avec LaTeX2.09

`\protect` `\contentsline` La version la plus récente de LaTeX 2ε ajoute `\protect` avant `\contentsline` dans les fichiers `.toc`, `.lof` et `.lof`. La version #17 de `minitoc` tente d'être compatible avec LaTeX 2ε et LaTeX2.09. Ce sera la *dernière* version utilisable avec LaTeX2.09. Les versions #18 et ultérieures sont spécifiques de LaTeX 2ε, et ne sont plus compatibles avec LaTeX2.09, qui est totalement obsolète.

2.13 Documents ré-initialisant le numéro de chapitre pour chaque partie

Depuis la version #23, `minitoc` fonctionne avec les classes de document qui ré-initialisent le numéro de chapitre (ou de section) pour chaque partie (ou chapitre). Ceci est possible car les fichiers auxiliaires pour les mini-tables ont désormais un numéro *absolu*.

2.14 Les mini-tables ont leurs lignes trop espacées

Depuis la version #29, vous pouvez avoir des mini-tables plus resserrées avec l'option `tight`, avec l'option `k-tight` pour les classes Koma-script (depuis la version #43).

2.15 Les secttocs sont fausses

Les secttocs ne fonctionnent pas : corrigé (version #38).

2.16 Comment retirer les points de conduite ?

Les lignes de points (points de conduite) entre les titres de sections et les numéros de pages sont retirés par l'option `undotted` (#29). Voir aussi la section 1.3.13 page 31.

2.17 Comment utiliser le paquetage `hyperref` avec `minitoc` ?

Depuis la version #31, `minitoc` fonctionne correctement avec le puissant paquetage `hyperref` [39], grâce à Heiko OBERDIEK, qui a utilisé les travaux de Bernd JAEHNE et Didier VERNA. Si vous ajoutez le chargement du paquetage `hyperref` à un document utilisant déjà `minitoc`, vous recevrez un message d'erreur concernant des accolades fermantes en trop. Il suffit de laisser se terminer l'exécution de LaTeX, puis de ré-exécuter LaTeX sur le document. Il n'y aura aucun problème si vous enlevez le chargement de `hyperref` puis l'ajoutez de nouveau ; ce problème ne se produit que lorsque vous passez de `minitoc` version #30 à `minitoc` version #31 (ou supérieure) avec un document déjà traité et en ajoutant `hyperref` en même temps ! Il semble préférable de traiter le document avec `minitoc` version #31 (ou supérieure) sans `hyperref`, puis avec `hyperref`, parce que certaines commandes internes écrites dans les fichiers auxiliaires ont été modifiées. S'il est utilisé, le paquetage `hyperref` doit être chargé *avant* `minitoc`. Notez que `(f)minitoc.dtx` montrent un exemple (qui n'est pas vraiment) basique de l'utilisation du paquetage `hyperref` avec `minitoc`.

2.18 Problème lors de la mise à jour de `minitoc`

Lors de la mise à jour de la version #30 ou inférieure vers la version #31 ou supérieure, vous devriez détruire les fichiers `.toc`, `.lof`, `.lot` du document, sinon la première exécution de LaTeX avec la version #31 ou supérieure produira beaucoup d'erreurs (l'exécution suivante devrait être correcte). Voir aussi la section 2.17.

2.19 Une table des matières locale pour l'ensemble des appendices

\doparttoc
\tableofcontents
\appendix
\part
\parttoc
\addtocontents
\protect
\setcounter
\chapter
\partbegin

Certains utilisateurs ont besoin d'une table des matières pour les appendices, mais sans mettre les entrées de cette table dans la table des matières principale. Une solution est de mettre les appendices dans une subdivision `\part` du document et de demander une table des matières au niveau de cette partie :

```
\doparttoc          % après \begin{document}
. . .
\tableofcontents . . .
\appendix
```

```
\part{Appendices}    % crée une subdivision de niveau partie
\parttoc             % crée une table des matières locale
% Pour supprimer la partie appendices dans la TdM principale
\addtocontents{toc}{\protect\setcounter{tocdepth}{-1}}
\chapter{Premier appendice}
. . .
% Ajoutez ceci à la fin des appendices s'il y a quelque chose
% après les appendices (comme un index ou une bibliographie)
% pour placer une borne au contenu de \parttoc
\addtocontents{toc}{\protect\partbegin}
```

Voir aussi la section 2.25 page 50.

2.20 Utilisation avec le paquetage `appendix`

appendices
\addcontentsline
\adjustmtc
\adjuststc

Si vous utilisez le paquetage `appendix` [46] (de Peter R. WILSON), vous observerez un sérieux problème avec les minitocs dans l'environnement `appendices` (et après lui) : elles ne correspondent pas avec leur appendice. En fait, l'ouverture de l'environnement par `\begin{appendices}` dissimule une commande `\addcontentsline` pour un chapitre ou une section, semant le trouble dans la numérotation des minitocs ou secttocs. Deux solutions sont disponibles. La première consiste à ajouter une commande `\adjustmtc` ou `\adjuststc` (selon que les appendices sont au niveau chapitre ou section) après *chaque* commande `\begin{appendices}`. L'autre solution est d'ajouter les commandes suivantes dans le préambule *après* le chargement du paquetage `appendix` :

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

si les appendices sont au niveau chapitre, OU :

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

si les appendices sont au niveau section.

2.21 Utilisation avec le paquetage `tocloft`

\mtcsetfont

(Cette réponse est donnée dans la documentation du paquetage `tocloft` [44].) Les paquetages `tocloft` (de Peter R. WILSON) et `minitoc` ont une fâcheuse interaction[2], qui, heureusement, peut être corrigée. Dans le cours normal des choses, lorsque `minitoc` est utilisé dans un document avec chapitres, il composera les entrées de sections dans une fonte grasse. Si `tocloft` est utilisé en conjonction avec `minitoc`, alors les entrées de sections dans les minitocs sont composées dans la fonte normale, sauf pour les numéros de pages qui sont en fonte grasse, tandis que les entrées de sections dans la table des matières principale sont entièrement en fonte normale.

[2]Découverte par Lyndon DUDDING.

Un remède, si vous voulez que toutes des entrées de sections dans les minitocs soient en fonte normale, est de placer :

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

ou :

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

dans le préambule

Sinon, le remède est l'incantation suivante :

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

Pour que les entrées de sections dans la table des matières principale et dans les minitocs soient toutes en gras, placez l'incantation dans le préambule. Pour que ces entrées soient en gras dans les minitocs mais en fonte normale dans la table des matières principale, placer l'incantation entre la commande `\tableofcontents` et la première commandes `\chapter`.

Puisque `tocloft` est un paquetage très puissant et très utile, cela vaut la peine d'ajouter ces remèdes si vous voulez profiter des avantages de ce paquetage.

2.22 Utilisation avec la classe `memoir`

La classe `memoir` [48] offre essentiellement les fonctionnalités des paquetages `appendix`, `tocbibind` et `tocloft` (cette classe et ces paquetages ont le même auteur, Peter R. Wilson), et donc elle a les mêmes problèmes ; voir ci-dessus les solutions disponibles. Si votre version de la classe `memoir` est récente, la syntaxe de la commande `\chapter` est différente et la classe `memoir` *n'est plus compatible* avec le paquetage `minitoc`, mais une modification est insérée pour corriger le problème.

2.23 Il y a trop de commandes pour les fontes, les titres et les profondeurs

`\mtcsetfont` `\mtcsettitlefont` Depuis la version #41, les commandes `\mtcsetfont` et `\mtcsettitlefont` sont disponibles. Vous n'avez plus besoin de connaître `\mtcSSSfont`, `\ptifont`, etc.

`\mtcsettitle` Depuis la version #42, la commande `\mtcsettitle` est disponible. Vous n'avez plus besoin de connaître `\mtctitle`, `\slttitle`, etc.

`\mtcsetdepth` Depuis la version #43, la commande `\mtcsetdepth` est disponible. n'avez plus besoin de connaître les compteurs `minitocdepth`, `seclotdepth`, etc.

2.24 Compatibilité avec les classes de document de l'$\mathcal{AMS}$

\mtcaddchapter

Les classes de document `amsart.cls` et `amsproc.cls` sont incompatibles avec `minitoc`. La classe de document `amsbook.cls` requiert l'insertion de commandes si vous désirez une liste des figures et/ou une liste des tableaux :

```
\listoffigures
\mtcaddchapter % ajout
\listoftables
\mtcaddchapter % ajout
```

2.25 Comment dissimuler certaines entrées dans la table des matières principale

mtchideinmaintoc
mtchideinmainlof
mtchideinmainlot

Ceci est un problème similaire à celui de la section 2.19 page 47. Un exemple est d'avoir une table des matières locale pour un chapitre (`\minitoc`) dont les entrées ne devraient pas apparaître dans la table des matières principale. Il suffit d'utiliser l'environnement `mtchideinmaintoc` :

```
\chapter{Titre}
\begin{mtchideinmaintoc}[niveau]
\minitoc
\section{sous-titre}
...
\end{mtchideinmaintoc}
```

Cet environnement accepte un argument optionnel, qui est la profondeur de dissimulation dans la table des matières principale (par défaut, `-1`, dissimulation complète).

Bien sûr, les environnements `mtchideinmainlof` et `mtchideinmainlot` sont également disponibles, pour dissimuler certaines entrées dans les listes des figures ou des tableaux principales.

2.26 Comment définir votre propre fichier `.mld` ?

\mtcsettitle
\mtcselectlanguage

Tout d'abord, vous ne devriez pas modifier directement l'un des fichiers `.mld` de la distribution. La manière la plus simple de modifier un certain titre est de redéfinir la commande correspondante via `\renewcommand` ou mieux, via `\mtcsettitle`. Si vous souhaitez vraiment avoir votre propre fichier `.mld`, vous copiez un fichier `.mld` existant vers un fichier portant un nouveau nom (pas le nom d'un fichier `.mld` de la distribution). Puis vous modifiez ce nouveau fichier `.mld` et vous pourrez l'utiliser via `\mtcselectlanguage`. Vous pouvez toujours me contacter pour ajouter ce nouveau fichier `.mld` à la distribution. Ces remarques s'appliquent aussi aux paires ⟨*langage*⟩`[.mld-.mlo]` de fichiers de définition de langues.

2.27 Utilisation avec le paquetage abstract

abstract
\mtcaddchapter
\mtcaddsection
\chapter

Lorsque le paquetage abstract [45] (de Peter R. WILSON), est utilisé avec son option addtotoc, une entrée « Abstract » (ou similaire) est ajoutée dans la table des matières, comme un chapitre étoilé si la classe du document définit \chapter, sinon comme une section étoilée. Ce problème est détecté par l'option hints et vous devriez ajouter une commande \mtcaddchapter[] ou \mtcaddsection[] après votre environnement abstract.

2.28 Utilisation avec le paquetage sectsty

Si le paquetage sectsty [32] (de Rowland McDONNELL) est utilisé, il doit être chargé *avant* le paquetage minitoc, car il altère (redéfinit) les commandes de sectionnement. Bien sûr, l'option hints détecte ce problème.

2.29 Alignement bizarre dans les minitocs

Dans les minitocs, les titres des sous-sections ne sont pas alignés avec les sections, alors qu'ils le sont dans la table des matières principale.

Les entrées d'une table des matières sont mises en forme par des commandes internes telles que \l@part, \l@chapter, \l@section, etc.

Les niveaux « partie » et « chapitre » (et « section » pour un article) utilisent des commandes spécifiques qui sont assez complexes pour une mise en forme plus élaborée. Pour les niveaux « section » (dans les classes report et book) et inférieurs, ces commandes sont (class book, book.cls) par défaut :

```
\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}}
```

qui seront appliquées dans la table des matières principale et dans les minitocs. Les arguments de \@dottedtocline sont

1) la profondeur logique (qui sera comparée à tocdepth ou minitocdepth) ;

2) l'indentation ;

3) la largeur réservée pour le numéro de page.

Dans les classes standard book, report et article, les dimensions (deuxième et troisième arguments) sont données en unités « em », et cette unité dépend de la fonte courante. Dans la table des matières principale, les entrées de sections et de sous-sections sont écrites dans la *même* fonte, donc normalement l'alignement est correct. Mais dans les minitocs, les entrées pour les sections sont écrites dans une fonte grasse tandis que celles des sous-sections sont écrites dans une fonte non grasse (voir la table 1.6 page 26), donc un « em » a des tailles différentes dans ces deux fontes et l'alignement est altéré.

Il y a plusieurs solutions :

– Redéfinir les commandes `\l@section` ... `\l@subparagraph` pour utiliser des unités indépendantes de la fonte (pt, mm, pc, etc.). Cette redéfinition doit être effectuée entre `\makeatletter` et `\makeatother`, puisque ces commandes contiennent un `@` dans leurs noms ; vous devez utiliser `\renewcommand*` pour redéfinir ces commandes.
– Utiliser le paquetage `tocloft` [44] pour changer l'indentation, avec des unités indépendantes de la fonte. Mais alors regardez la section 2.21 page 48.
– Utiliser la même fonte pour les entrées de sections et de sous-sections dans les minitocs, en utilisant la commande `\mtcsetfont` (voir la section 1.3.9 page 30) ou en redéfinissant les commandes `\mtcSfont`, `\mtcSSfont`, `\mtcSSSfont`, `\mtcPfont` et `\mtcSPfont` (voir la table 1.6 page 26), ou analogues.

Chapitre 3

Installation

Tableaux

Installation du paquetage `minitoc` (version #43).

Ce paquetage contient de nombreux fichiers. La liste de tous les fichiers est donnée dans `minitoc.l`. Voir le tableau 3.1 page 55.

Les fichiers sont répartis en « classes » ci-dessous (un même fichier peut apparaître dans plusieurs classes). Chaque classe spécifie la fonction et le placement de ses fichiers.

(0) Les fichiers `minitoc.ins` et `minitoc.dtx` sont les fichiers sources basiques de ce paquetage. Le fichier `fminitoc.dtx` charge `minitoc.dtx` mais sélectionne la documentation en français. La sélection de la langue est faite en utilisant des constructions `\ifcase` ... `\or` ... `\fi`.

(1) Les fichiers `minitoc.sty`, `mtcoff.sty`, et *tous* les fichiers `*.mld` et `*.mlo` constituent le paquetage lui-même[1] `mtcpatchmem.sty` est une correction temporaire pour compatibilité avec la classe `memoir`. Ils doivent être *tous* installés dans un répertoire où LaTeX 2ε trouve les fichiers `.sty`.

(2) Fichiers informatifs textuels :
- `INSTALL` est un fichier décrivant l'installation du paquetage. À part la langue, vous êtes (presque) en train de le lire (mais il est plus court).
- `minitoc.l` contient la liste de tous les fichiers de la distribution de `minitoc`. Voir le tableau 3.1 page 55.
- `README` est un fichier décrivant sommairement le paquetage `minitoc`, plus quelques informations utiles.

[1]Le grand nombre de fichiers `*.mld` est (en partie) une conséquence du fait que certaines langues ont des alias (ou dialectes) et donc un fichier `*.mld` pour chaque nom (un fichier `*.mld` peut en charger un autre) ; les langues *english* et *french* sont des exemples évidents. Pour certaines langues, la multiplicité des fichiers `*.mld` correspond à une multiplicité de fontes et/ou de codages (chinois, grec, japonais, coréen, malayalam, polonais, russe, serbe), ou même à des réformes de l'orthographe (allemand, grec, norvégien).

- `catalog` contient des informations basiques sur le paquetage `minitoc` (résumé, date, auteur, version, licence).
- `TODO` liste quelques propositions de développement du paquetage, pas encore réalisées. Commentaires et suggestions sont les bienvenus.

(3) `minitoc-ex.tex`, `mini-art.tex`, `add.tex`, `add.bib` sont des exemples de fichiers, avec lesquels vous pouvez jouer.

(4) `minitoc.bug`, `minitoc.sum` sont de la documentation en texte pur : liste des problèmes (questions fréquemment posées, voir le chapitre 2 page 41) et sommaire des commandes (voir le chapitre 4 page 56).

(5) `minitoc.ins`, `minitoc.ist`, `mtcglo.ist`, `minitoc.lan`, `minitoc.dtx` et `minitoc.bib` forment le code source de la documentation en anglais (ou presque).

(6) `minitoc.dtx`, `fminitoc.dtx`, `fminitoc.bib`, `fminitoc.ist`, `fminitoc.lan`, `franc.sty`, `frbib.sty`, `frnew.sty`, `frplain1.bst` sont les codes sources (et les outils) pour la documentation en français [2].

(7) `minitoc.pdf`, `minitoc.ps` forment la documentation en anglais (ou presque), dans les formats PDF et PostScript.

(8) `fminitoc.pdf`, `fminitoc.ps` forment la documentation en français, dans les formats PDF et PostScript. La documentation en français et ses fichiers sources ne doivent pas être omis.

(9) `pmk` est un script *shell* [3] pour préparer le paquetage et sa documentation ; vous devriez l'adapter à vos besoins. Il y a aussi quatre scripts partiels (à adapter) :
- `imk`, qui prépare le paquetage en partant de `minitoc.ins` et `minitoc.dtx` ;
- `emk`, qui prépare la documentation anglaise en partant de `minitoc.dtx` ;
- `fmk`, qui prépare la documentation française en partant de `fminitoc.dtx` et `minitoc.dtx` ;
- `rmk`, qui trie les fichiers en classes (un répertoire pour chaque classe).

Ces scripts sont actuellement écrits en C-*shell*, mais ils sont très simples et devraient être faciles à convertir dans tout autre *shell* classique.

- Les fichiers de (0) doivent être installés dans un répertoire où LaTeX 2_ε trouve les fichiers `.dtx` et `.ins`.
- Les fichiers de (1) doivent être installés dans un répertoire où LaTeX 2_ε trouve les fichiers `.sty`.
- Les fichiers de (2), (3), (4), (5), (6) et (9) doivent être installés dans un répertoire distinct, mais ne doivent pas être omis.
- Les fichiers de (7) et (8) doivent être installés comme documentation en ligne. Notez que les fichiers PostScript de la distribution peuvent être imprimés recto-verso sur certaines imprimantes.

Notez que `minitoc.dtx` et `fminitoc.dtx` sont des exemples (non triviaux) de l'utilisation de `minitoc` (avec `hyperref`).

[2] Ceci peut sembler assez étrange. En fait, les documentations en anglais et en français sont toutes deux dans le fichier `minitoc.dtx`. `fminitoc.dtx` établit un indicateur puis charge `minitoc.dtx` ; en conséquence, le fichier `fminitoc.dtx` est bien plus petit que `minitoc.dtx`. Donc, `minitoc.ins` contient aussi certains fichiers utilitaires qui sont créés automatiquement (quelques fichiers `.sty`, `minitoc.ist`, `mtcglo.ist`, `fminitoc.ist`, `minitoc.lan`, `fminitoc.lan`). Les versions anglaise et française ne sont pas des traductions mot-à-mot, mais elles sont en parallèle dans le fichier `minitoc.dtx`, et ceci aide pour la maintenance. Notez que le fichier `fminitoc.ins` n'existe pas.

[3] Vous avez le temps de déguster un cappuccino pendant que ce script s'exécute !

TAB. 3.1 – Liste des fichiers (minitoc.l)

classe (0) :

–minitoc.ins
–minitoc.dtx
–fminitoc.dtx

classe (1) :

–minitoc.sty
–mtcoff.sty
–mtcpatchmem.sty
–acadian.mld
–acadien.mld
–afrikaan.mld
–afrikaans.mld
–american.mld
–arab.mld
–arabic.mld
–armenian.mld
–austrian.mld
–bahasa.mld
–bangla.mld
–basque.mld
–bicig.mld
–brazil.mld
–brazilian.mld
–breton.mld
–british.mld
–bulgarian.mld
–bulgarianb.mld
–buryat.mld
–canadian.mld
–canadien.mld
–castillan.mld
–castillian.mld
–catalan.mld
–chinese1.mld et chinese1.mlo
–chinese2.mld et chinese2.mlo
–croatian.mld
–czech.mld
–danish.mld
–dutch.mld
–english.mld
–esperant.mld
–esperanto.mld
–estonian.mld
–ethiopia.mld
–ethiopian.mld
–farsi1.mld et farsi1.mlo
–farsi2.mld et farsi2.mlo
–finnish.mld
–finnish2.mld
–francais.mld
–french.mld
–frenchb.mld
–frenchle.mld
–frenchpro.mld
–galician.mld
–german.mld
–germanb.mld
–greek.mld
–greek-mono.mld
–greek-polydemo.mld
–greek-polykatha.mld
–guarani.mld
–hangul1.mld et hangul1.mlo
–hangul2.mld et hangul2.mlo
–hangul3.mld et hangul3.mlo
–hangul4.mld et hangul4.mlo
–hanja1.mld et hanja1.mlo
–hanja2.mld et hanja2.mlo
–hebrew.mld
–hungarian.mld
–icelandic.mld
–interlingua.mld
–irish.mld
–italian.mld
–japanese.mld et japanese.mlo
–japanese2.mld et japanese2.mlo
–japanese3.mld et japanese3.mlo
–japanese4.mld et japanese4.mlo
–japanese5.mld et japanese5.mlo
–latin.mld
–latin2.mld
–latvian.mld
–letton.mld
–lithuanian.mld
–lsorbian.mld
–magyar.mld
–magyar2.mld
–malayalam-keli.mld
–malayalam-rachana.mld
–malayalam-rachana2.mld
–mongol.mld
–naustrian.mld
–ngerman.mld
–ngermanb.mld
–norsk.mld
–nynorsk.mld
–polish.mld
–polish2.mld
–portuges.mld
–portuguese.mld
–romanian.mld
–russian.mld
–russianb.mld
–russianc.mld
–russian2m.mld
–russian2o.mld
–samin.mld
–scottish.mld
–serbian.mld
–serbianc.mld
–slovak.mld
–slovene.mld
–spanish.mld
–spanish2.mld
–spanish3.mld
–swedish.mld
–thai.mld et thai.mlo
–turkish.mld
–UKenglish.mld
–ukraineb.mld
–USenglish.mld
–usorbian.mld
–vietnam.mld
–vietnamese.mld
–welsh.mld

classe (2) :

–INSTALL
–minitoc.l
–README
–catalog
–TODO

classe (3) :

–minitoc-ex.tex
–mini-art.tex
–add.tex
–add.bib

classe (4) :

–minitoc.bug
–minitoc.sum

classe (5) :

–minitoc.ins
–minitoc.dtx
–minitoc.bib
–minitoc.ist
–mtcglo.ist
–minitoc.lan

classe (6) :

–minitoc.dtx
–fminitoc.dtx
–fminitoc.bib
–fminitoc.ist
–fminitoc.lan
–franc.sty
–frbib.sty
–frnew.sty
–frplain1.bst

classe (7) :

–minitoc.pdf
–minitoc.ps

classe (8) :

–fminitoc.pdf
–fminitoc.ps

classe (9) :

–pmk
–imk
–emk
–fmk
–rmk

Chapitre 4

Mémento

Tableaux

TAB. 4.1 – Options du paquetage

Options	**Défaut**	**Signification**
`shortext`	`*NON*`	Pour utiliser des suffixes courts pour les fichiers auxiliaires.
`loose`, `tight`	`loose`	Espacement des lignes dans les mini-tables.
`k-loose`, `k-tight`	`k-loose`	Espacement des lignes dans les mini-tables (classes Koma-script).
`dotted`, `undotted`	`dotted`	Ponts de conduites entre les entrées et les numéros de pages.
`insection`	`*NON*`	Empêche les éléments flottants (figures et tableaux) de dériver en dehors de leur section. Utile si vous utilisez des sectlofs/sectlots.
`notoccite`	`*NON*`	Utile si vous avez des commandes `\cite` dans les titres de sectionnements et utilisez un style de bibliographie sans tri.
`hints`, `nohints`	`hints`	Ajoute des indications (*hints*) dans le fichier `.log`. Utile pour détecter certains problèmes.

Les options de langue sont listées dans le tableau 1.7 page 27. Défaut : `english`.

TAB. 4.2 – Commandes générales

Commande	**Signification**
`\faketableofcontents`	Remplace `\tableofcontents` si vous souhaitez des mini-tables des matières mais pas de table des matières principale.
`\fakelistoffigures`	Remplace `\listoffigures` si vous souhaitez des mini-listes des figures mais pas de liste des figures principale.
`\fakelistoftables`	Remplace `\listoftables` si vous souhaitez des mini-listes des tableaux mais pas de liste des tableaux principale.
`\mtcselectlanguage{`*langue*`}`	Charge *langue*`.mld` pour sélectionner une langue pour les titres des mini-tables.
`\mtcsetdepth{`*mini-table*`}{`*profondeur*`}`	Change la profondeur pour certaines mini-tables.
`\mtcsetfeature{`*mini-table*`}{before|after|pagestyle}{`*commandes*`}`	Modifie les dispositifs pour une mini-table.
`\mtcsetfont{`*mini-table*`}{`*niveau-sectionnement*`}{`*commandes de fonte*`}`	Redéfinit une commande de fonte minitoc.
`\mtcsetformat{`*mini-table*`}{`*paramètre*`}{`*valeur*`}`	Change la mise en page de certaines mini-tables.
`\mtcsetpagenumbers{`*mini-table*`}{on|off}`	Active/inhibe les numéros de pages dans certaines mini-tables.
`\mtcsetrules{`*mini-table*`}{on|off}`	Active/inhibe les filets horizontaux dans certaines mini-tables.
`\mtcsettitle{`*mini-table*`}{`*chaîne titre*`}`	Change le titre pour certaines mini-tables.
`\mtcsettitlefont{`*mini-table*`}{`*commandes de fonte*`}`	Change la fonte du titre pour certaines mini-tables.
`\mtcskip`	Pour ajouter un saut vertical entre les mini-tables.
`\mtcskipamount`	Longueur de `\mtcskip`. Défaut : `\bigskipamount`.
`\tightmtcfalse`	Mini-tables avec lignes peu serrées. Défaut.
`\tightmtctrue`	Mini-tables avec lignes serrées.
`\ktightmtcfalse`	Mini-tables avec lignes peu serrées. Défaut. (Classes Koma-script).
`\ktightmtctrue`	Mini-tables avec lignes serrées. (Classes Koma-script).
`\undottedmtcfalse`	Points de conduite dans les mini-tables (de l'entrée jusqu'au numéro de page). Défaut.
`\undottedmtctrue`	Pas de points conduite dans les mini-tables (de l'entrée jusqu'au numéro de page).

TAB. 4.3 – Commandes, niveau partie

Commande	**Signification**
\doparttoc[*x*]	Avant \[fake]tableofcontents si vous utilisez \parttoc[*].
\dopartlof[*x*]	Avant \[fake]listoffigures si vous utilisez \partlof[*].
\dopartlot[*x*]	Avant \[fake]listoftables si vous utilisez \partlot[*].
\parttoc[*x*]	Après chaque commande \part pour laquelle une parttoc est nécessaire[*].
\partlof[*x*]	Après chaque commande \part pour laquelle une partlof est nécessaire[*].
\partlot[*x*]	Après chaque commande \part pour laquelle une partlot est nécessaire[*].
\setcounter{parttocdepth}{*profondeur*}	
	Profondeur des parttocs ultérieures. Analogue à tocdepth. Défaut : 2. N'a aucune action sur les partlofs et partlots.
ou :	
\mtcsetdepth{parttoc|partlof|partlot}{*profondeur*}	
	Idem, mais peut aussi agir sur les partlofs et partlots.
\ptcindent	Indentation gauche et droite d'une table partielle. Défaut : 24pt.
\ptcfont	Commande de fonte pour les parttocs. Défaut : \small\rmfamily\upshape\mdseries (article) ou : \normalsize\rmfamily\upshape\mdseries (book, report).
\ptcCfont	Commande de fonte pour les parttocs, entrées chapitres. Défaut : \normalsize\rmfamily\upshape\bfseries.
\ptcSfont	Commande de fonte pour les parttocs, entrées sections. Défaut : \small\rmfamily\upshape\bfseries (article) ou : \small\rmfamily\upshape\bfseries (book, report).
\ptcSSfont	Commande de fonte pour les parttocs, entrées sous-sections[**].
\ptcSSSfont	Commande de fonte pour les parttocs, entrées, entrées sous-sous-sections[**].
\ptcPfont	Commande de fonte pour les parttocs, entrées paragraphes[**].
\ptcSPfont	Commande de fonte pour les parttocs, entrées sous-paragraphes[**].
\plffont	Fonte pour les partlofs. Défaut : \small\rmfamily\upshape\mdseries.
\pltfont	Fonte pour les partlots. Défaut : \small\rmfamily\upshape\mdseries.
\ptctitle	Titre des parttocs. Défaut : Table of Contents.
\plftitle	Titre des partlofs. Défaut : List of Figures.
\plttitle	Titre des partlots. Défaut : List of Tables.
\ptifont	Fonte pour les titres des partXXX. Défaut : \Large\rmfamily\upshape\bfseries (article) ou : \LARGE\rmfamily\upshape\bfseries (book, report).

∗ : [*x*] est un argument optionnel pour établir la position du titre ; le choix est local pour les commandes \partXXX, global pour les commandes \dopartXXX. Les valeurs de *x* sont : l pour gauche (défaut), c pour centré, r pour droite, n ou e pour pas de titre.
∗∗ : la valeur par défaut est comme \ptcfont.

TAB. 4.4 – Commandes, niveau chapitre

Commande	Signification
\dominitoc[*x*]	Avant \[fake]tableofcontents si vous utilisez \minitoc*.
\dominilof[*x*]	Avant \[fake]listoffigures si vous utilisez \minilof*.
\dominilot[*x*]	Avant \[fake]listoftables si vous utilisez \minilot*.
\minitoc[*x*]	Après chaque commande \chapter pour laquelle une minitoc est nécessaire*.
\minilof[*x*]	Après chaque commande \chapter pour laquelle une minilof est nécessaire*.
\minilot[*x*]	Après chaque commande \chapter pour laquelle une minilot est nécessaire*.
\setcounter{minitocdepth}{*profondeur*}	
	Profondeur des minitocs ultérieures. Analogue à tocdepth. Défaut : 2. N'a aucune action sur les minilofs et minilots.
ou :	
\mtcsetdepth{minitoc|minilof|minilot}{*profondeur*}	
	Idem, mais peut aussi agir sur les minilofs et minilots.
\mtcindent	Indentation gauche et droite d'une mini-table. Défaut : 24pt.
\mtcfont	Commande de fonte pour les minitocs. Défaut : \small\rmfamily\upshape\mdseries.
\mtcSfont	Commande de fonte pour les minitocs, entrées sections. Défaut : \small\rmfamily\upshape\bfseries.
\mtcSSfont	Commande de fonte pour les minitocs, entrées sous-sections**.
\mtcSSSfont	Commande de fonte pour les minitocs, entrées sous-sous-sections**.
\mtcPfont	Commande de fonte pour les minitocs, entrées paragraphes**.
\mtcSPfont	Commande de fonte pour les minitocs, entrées sous-paragraphes**.
\mlffont	Commande de fonte pour les minilofs. Défaut : \small\rmfamily\upshape\mdseries.
\mltfont	Commande de fonte pour les minilots. Défaut : \small\rmfamily\upshape\mdseries.
\mtctitle	Titre des minitocs. Défaut : Contents.
\mlftitle	Titre des minilofs. Défaut : Figures.
\mlttitle	Titre des minilots. Défaut : Tables.
\mtifont	Fonte pour les titres des miniXXX. Défaut : \large\rmfamily\upshape\bfseries.

$*$: [*x*] est un argument optionnel pour établir la position du titre ; le choix est local pour les commandes \miniXXX, global pour les commandes \dominiXXX. Les valeurs de *x* sont : l pour gauche (défaut), c pour centré, r pour droite, n ou e pour pas de titre.

$**$: la valeur par défaut est comme \mtcfont.

TAB. 4.5 – Commandes, niveau section

Commande	**Signification**
`\dosecttoc[x]`	Avant `\[fake]tableofcontents` si vous utilisez `\secttoc`*.
`\dosectlof[x]`	Avant `\[fake]listoffigures` si vous utilisez `\sectlof`*.
`\dosectlot[x]`	Avant `\[fake]listoftables` si vous utilisez `\sectlot`*.
`\secttoc[x]`	Après chaque commande `\section` pour laquelle une secttoc est nécessaire*.
`\sectlof[x]`	Après chaque commande `\section` pour laquelle une sectlof est nécessaire*.
`\sectlot[x]`	Après chaque commande `\section` pour laquelle une sectlot est nécessaire*.
`\setcounter{secttocdepth}{profondeur}`	
	Profondeur des secttocs ultérieures. Analogue à `tocdepth`. Défaut : 2. N'a aucune action sur les sectlofs et sectlots.
ou :	
`\mtcsetdepth{secttoc|sectlof|sectlot}{profondeur}`	
	Idem, mais peut aussi agir sur les sectlofs et sectlots.
`\stcindent`	Indentation gauche et droite d'une mini-table. Défaut : `24pt`.
`\stcfont`	Commande de fonte pour les secttocs. Défaut : `\small\rmfamily\upshape\mdseries`.
`\stcSSfont`	Commande de fonte pour les secttocs, entrées sous-sections**.
`\stcSSSfont`	Commande de fonte pour les secttocs, entrées sous-sous-sections**.
`\stcPfont`	Commande de fonte pour les secttocs, entrées paragraphes**.
`\mtcSPfont`	Commande de fonte pour les secttocs, entrées sous-paragraphes**.
`\slffont`	Commande de fonte pour les sectlofs. Défaut : `\small\rmfamily\upshape\mdseries`.
`\sltfont`	Commande de fonte pour les sectlots. Défaut : `\small\rmfamily\upshape\mdseries`.
`\stctitle`	Titre des secttocs. Défaut : `Contents`.
`\slftitle`	Title des sectlofs. Défaut : `Figures`.
`\slttitle`	Titre des sectlots. Défaut : `Tables`.
`\stifont`	Fonte pour les titres des sectXXX. Défaut : `\large\rmfamily\upshape\bfseries`.

* : `[x]` est un argument optionnel pour établir la position du titre ; le choix est local pour les commandes `\sectXXX`, global pour les commandes `\dosectXXX`. Les valeurs de x sont : `l` pour gauche (défaut), `c` pour centré, `r` pour droite, `n` ou `e` pour pas de titre.
** : la valeur par défaut est comme `\stcfont`.

TAB. 4.6 – Commandes pour les filets horizontaux

Commande	**Signification**
`\[no]ptcrule`	Active ou inhibe les filets dans les parttocs.
`\[no]mtcrule`	Active ou inhibe les filets dans les minitocs.
`\[no]stcrule`	Active ou inhibe les filets dans les secttocs.
`\[no]plfrule`	Active ou inhibe les filets dans les partlofs.
`\[no]mlfrule`	Active ou inhibe les filets dans les minilofs.
`\[no]slfrule`	Active ou inhibe les filets dans les sectlofs.
`\[no]pltrule`	Active ou inhibe les filets dans les partlots.
`\[no]mltrule`	Active ou inhibe les filets dans les minilots.
`\[no]sltrule`	Active ou inhibe les filets dans les sectlots.
`\mtcsetrules{mini-table}{on|off}`	
	Active/inhibe les filets horizontaux dans certaines mini-tables.

Par défaut, les parttocs n'ont pas de filets ; les minitocs et secttocs ont des filets. Dans les articles, les parttocs ont des filets.

TAB. 4.7 – Commandes pour les numéros de pages

Commande	Signification
`\[no]ptcpagenumbers`	Active ou inhibe les numéros de pages dans les parttocs.
`\[no]plfpagenumbers`	Active ou inhibe les numéros de pages dans les partlofs.
`\[no]pltpagenumbers`	Active ou inhibe les numéros de pages dans les partlots.
`\[no]mtcpagenumbers`	Active ou inhibe les numéros de pages dans les minitocs.
`\[no]mlfpagenumbers`	Active ou inhibe les numéros de pages dans les minilofs.
`\[no]mltpagenumbers`	Active ou inhibe les numéros de pages dans les minilots.
`\[no]stcpagenumbers`	Active ou inhibe les numéros de pages dans les secttocs.
`\[no]slfpagenumbers`	Active ou inhibe les numéros de pages dans les sectlofs.
`\[no]sltpagenumbers`	Active ou inhibe les numéros de pages dans les sectlots.
`\mtcsetpagenumbers{`*mini-table*`}{on|off}`	Active/inhibe les numéros de pages dans certaines mini-tables.

Par défaut, les numéros de pages sont présents.

TAB. 4.8 – Commandes spécifiques pour les dispositifs des parttocs et autres mini-tables

Commande	Défaut	Signification
`\beforeparttoc`	`\cleardoublepage`	Action avant une parttoc.
`\beforepartlof`	`\cleardoublepage`	Action avant une partlof.
`\beforepartlot`	`\cleardoublepage`	Action avant une partlot.
`\afterparttoc`	`\cleardoublepage`	Action après une parttoc.
`\afterpartlof`	`\cleardoublepage`	Action après une partlof.
`\afterpartlot`	`\cleardoublepage`	Action après une partlot.
`\thispageparttocstyle`	`\thispagestyle{empty}`	Style de page pour une parttoc.
`\thispagepartlofstyle`	`\thispagestyle{empty}`	Style de page pour une partlof.
`\thispagepartlotstyle`	`\thispagestyle{empty}`	Style de page pour une partlot.
`\beforeminitoc`	`\empty`	Action avant une minitoc.
`\beforeminilof`	`\empty`	Action avant une minilof.
`\beforeminilot`	`\empty`	Action avant une minilot.
`\afterminitoc`	`\empty`	Action après une minitoc.
`\afterminilof`	`\empty`	Action après une minilof.
`\afterminilot`	`\empty`	Action après une minilot.
`\thispageminitocstyle`	`\empty`	Style de page pour une minitoc.
`\thispageminilofstyle`	`\empty`	Style de page pour une minilof.
`\thispageminilotstyle`	`\empty`	Style de page pour une minilot.
`\beforesecttoc`	`\empty`	Action avant une secttoc.
`\beforesectlof`	`\empty`	Action avant une sectlof.
`\beforesectlot`	`\empty`	Action avant une sectlot.
`\aftersecttoc`	`\empty`	Action après une secttoc.
`\aftersectlof`	`\empty`	Action après une sectlof.
`\aftersectlot`	`\empty`	Action après une sectlot.
`\thispagesecttocstyle`	`\empty`	Style de page pour une secttoc.
`\thispagesectlofstyle`	`\empty`	Style de page pour une sectlof.
`\thispagesectlotstyle`	`\empty`	Style de page pour une sectlot.
`\mtcsetfeature{`*mini-table*`}{before|after|pagestyle}{`*commandes*`}`		Modifie les dispositifs pour une mini-table.

TAB. 4.9 – Commandes de préparation et d'insertion

Type	Phase	Niveau		
		partie	**chapitre**	**section**
table des matières	préparation	\doparttoc[*p*]	\dominitoc[*p*]	\dosecttoc[*p*]
	insertion	\parttoc[*p*]	\minitoc[*p*]	\secttoc[*p*]
liste des figures	préparation	\dopartlof[*p*]	\dominilof[*p*]	\dosectlof[*p*]
	insertion	\partlof[*p*]	\minilof[*p*]	\sectlof[*p*]
liste des tableaux	préparation	\dopartlot[*p*]	\dominilot[*p*]	\dosectlot[*p*]
	insertion	\partlot[*p*]	\minilot[*p*]	\sectlot[*p*]

Chacune de ces commandes accepte un argument optionnel *p*, qui spécifie la position du titre de la mini-table. Cet argument *p* a un effet global dans les commandes de préparation, mais local dans les commandes d'insertion. C'est une lettre : [l] pour aligné à gauche (défaut), [c] pour centré, [r] pour aligné à droite, [e] ou [n] pour vide (pas de titre).

TAB. 4.10 – Commandes d'ajustement

Commande	**Signification**
\adjustptc[*n*]	Ajuste (incrémente) le compteur de parttocs ptc de *n*.
\adjustmtc[*n*]	Ajuste (incrémente) le compteur de minitocs mtc de *n*.
\adjuststc[*n*]	Ajuste (incrémente) le compteur de secttocs stc de *n*.
\decrementptc	Ajuste (décrémente de 1) le compteur de parttocs ptc.
\decrementmtc	Ajuste (décrémente de 1) le compteur de minitocs mtc.
\decrementstc	Ajuste (décrémente de 1) le compteur de secttocs stc.
\incrementptc	Ajuste (incrémente de 1) le compteur de parttocs ptc.
\incrementmtc	Ajuste (incrémente de 1) le compteur de minitocs mtc.
\incrementstc	Ajuste (incrémente de 1) le compteur de secttocs stc.
\mtcaddpart[*titre*]	Ajoute le titre d'une \part* dans la TdM.
\mtcaddchapter[*titre*]	Ajoute le titre d'une \chapter* dans la TdM.
\mtcaddsection[*titre*]	Ajoute le titre d'une \section* dans la TdM.
\mtcfixindex[chapter|section|part]	Ajoute l'entrée pour l'index dans la TdM.
\mtcfixglossary[chapter|section|part]	Ajuste l'entrée pour le glossaire dans la TdM.
\begin{mtchideinmaintoc}[*profondeur*] ... \end{mtchideinmaintoc}	Environnement pour dissimuler des entrées dans la table des matières principale.
\begin{mtchideinmainlof}[*profondeur*] ... \end{mtchideinmainlof}	Environnement pour dissimuler des entrées dans la liste des figures principale.
\begin{mtchideinmainlot}[*profondeur*] ... \end{mtchideinmainlot}	Environnement pour dissimuler des entrées dans la liste des tableaux principale.

TAB. 4.11 – Commandes obsolètes

Commande	**Signification**
\firstpartis{*N*}	*N* est le numéro de la première partie.
\firstchapteris{*N*}	*N* est le numéro du premier chapter.
\firstsectionis{*N*}	*N* est le numéro de la première section.

Remerciements

Il me faut remercier les personnes suivantes [1], pour leur aide, leurs questions, leurs interventions dans les groupes de discussion [2] et/ou les paquetages et classes qu'elles ont écrits :
Hassan Abolhassani, Alex AJ, Achod André Aradian, Donald Arseneau, David Aspinall, Philipp Bachmann, Marin Balgarensky, Benjamin Bayart, Stephan P. von Bechtolsheim, Emmanuel Beffara, Javier Bezos, Laurent Bloch, Johannes Braams, Daniel Bourbonnais, David Carlisle, Steven Douglas Cochran, David B. Cook, Oliver Corff, Prakash Countcham, Serguei Dachian, Adrian Daerr, Ben De Rydt, Lyndon Dudding, Marko Èehaja, Victor Eijkhout, Thomas Esser, Karl F. Everitt, Robin Fairbairns, Ulrike Fischer, Mohammad Ghodsi, Michel Goossens, Boumediene Hamzi, Danny Heap, Thorsten Hein, Florence Henry, Morten Høgholm, Don Hosek, Yufan Hu, Dmitry Ivanov, Bernd Jaehne, Radwan Jalam, Dan Jurafsky, Bil Kleb, Thankmar Kronzucker, Alexej M. Kryukov, Toshiki Kumazawa, Frank Küster, Fabio Lanari, Robert Lange, Dag Langmyhr, Jean-Marc Lasgouttes, Claire Lauvernet, Werner Lemberg, Thomas Leonhardt, Stéphane Lepolozec, Adam Lewenberg, Knut Lickert, Dan Luecking, Anders Lyhne, Marcus Marr, Rowland McDonnell, Yanick Michou, Frank Mittelbach, Lapo Filippo Mori, Michael Morrison, Cuong Nguyen, Julien Nicolas, Heiko Oberdiek, Piet van Oostrum, Palash Baran Pal, Roozbeh Pournader, Sebastian Rahtz, Tony Roberts, Denis Roegel, Jan Michel Rynning, David Samsoen, Uwe Schneider, Dung Ta Quang, Laurent Tordella, Stefan Ulrich, Jari Vaario, Vincent Vaquin, Didier Verna, Sylvain Vesco, Nigel Ward, Staszek Wawrykiewicz, Stephan Webanck, Peter R. Wilson et Tim Wright.

[1] Et que celles que j'ai oubliées me pardonnent.

[2] Essentiellement, `fr.comp.text.tex` et `comp.text.tex`.

Deuxième partie

Implantation

Table des Matières

Chapitre 5

Code commenté de `minitoc.sty`

Sommaire

Figures

5.1 Introduction

Ce long chapitre présente le code du paquetage `minitoc` et tente de l'expliquer. Certains commentaires du fichier source original[1] sont sautés, comme l'historique, car ils n'ont pas besoin d'être examinés davantage (ils seront mis dans l'historique des modifications).

Le code est éclaté en sections pour faciliter la lecture, et ces sections sont parfois réordonnées, dans ce même but.

La plupart des commandes externes de `minitoc` ont `mtc`, `ptc`, `stc` ou le nom de l'un des types de mini-tables (`parttoc`,..., `sectlot`) dans leurs noms. La plupart des commandes internes de `minitoc` ont `@mtc`, `@ptc`, `@stc` ou `parttoc@`,..., `sectlot@` dans leurs noms, ou suivent une convention similaire. Les quelques exceptions devraient être assez explicites pour ne pas entrer en conflit avec d'autres paquetages.

5.2 Code d'identification

Le code de `minitoc.sty` commence ici :

```
1 ⟨*minitoc⟩
```

[1]C'est-à-dire, celui de la version #42. La version #43 inclut la conversion du paquetage au format `.dtx/.ins`. La version #42 n'a pas été distribuée pour cette raison.

\NeedsTeXFormat \ProvidesPackage \PackageInfo

Cette section de code identifie le paquetage par son nom, son numéro de version et sa date. Une trace est écrite dans le fichier .log. Ce paquetage ne fonctionnera pas avec LaTeX2.09.

```
\NeedsTeXFormat{LaTeX2e}[1996/06/01]%
\ProvidesPackage{minitoc}%
   [2005/09/16 v43 Package minitoc]
\PackageInfo{minitoc}%
   {*** minitoc package, version 43 ***\@gobble}
```

5.3 Un descripteur de fichier pour écrire

\tf@mtc \newwrite

Un descripteur de fichier est nécessaire pour écrire les fichiers contenant les mini-tables, c'est \tf@mtc.

```
\newwrite\tf@mtc
```

5.4 Indentation et saut

\mtcindent \mtcskip \mtcskipamount \parskip

Nous définissons l'indentation \mtcindent (des deux côtés) des mini-tables et la commande \mtcskip pour faire un saut vertical avant une a mini-table, sa valeur est \mtcskipamount (défaut : \bigskipamount).

```
\newlength\mtcindent
\newskip\mtcskipamount
\setlength{\mtcskipamount}{\bigskipamount}
\def\mtcskip{\unskip\removelastskip{\parskip=\z@\addvspace{\mtcskipamount}}}
```

Notez que \mtcskip utilise un groupe local pour éviter l'influence de \parskip.

5.5 Tests et indicateurs

Nous avons besoin de déclarer quelques indicateurs [2] (via \newif) pour détecter le chargement de certains paquetages ou classes et la disponibilité de certaines commandes (ceci sera utilisé par l'option hints (voir la section 5.76 page 191) ou pour autoriser la définition de certaines commandes de minitoc).

[2] Il n'y a pas tant d'années, certains auteurs préféraient utiliser des compteurs plutôt que des indicateurs, car un indicateur coûte 3 séquences de contrôle (\iftoto, \totofalse et \tototrue), qui consomment de la mémoire. Mais le nombre de registres compteurs est limité à 256 dans le moteur TeX natif (bien plus avec ε-TeX, mais encore en nombre limité), tandis que le coût de la mémoire a beaucoup baissé aujourd'hui. Et un code programmé avec des indicateurs (\iftoto ... \else ... \fi) est plus facile à structurer qu'un code programmé à l'aide de compteurs, à mon humble avis.

5.5.1 Utilisation de mini-listes d'éléments flottants, au niveau section.

\if@mtc@sect@floats@
\dosectlof
\dosectlot

Nous testerons si les commandes `\dosectlof` et `\dosectlot` ont été utilisées :

```
\newif\if@mtc@sect@floats@ \@mtc@sect@floats@false
```

5.5.2 Présence de certains paquetages et classes.

\if@mtc@placeinsLoaded@
\if@mtc@memoirLoaded@
\if@mtc@memoirnew@
\if@mtc@sectstyLoaded@
\if@mtc@sectstyLoaded@a@

Nous testerons si le paquetage `placeins` est chargé, puis si la classe `memoir` est chargée (et si c'est une version récente), puis si le paquetage `sectsty` est chargé (avant ou après `minitoc`).

```
\newif\if@mtc@placeinsLoaded@ \@mtc@placeinsLoaded@false
\newif\if@mtc@memoirLoaded@ \@mtc@memoirLoaded@false
\newif\if@mtc@memoirnew@ \@mtc@memoirnew@false
\newif\if@mtc@sectstyLoaded@ \@mtc@sectstyLoaded@false
\newif\if@mtc@sectstyLoaded@a@ \@mtc@sectstyLoaded@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@sectlot@
\if@mtc@empty@sectlof@
\if@mtc@empty@sectlot@

Nous testerons si vous avez tenté d'insérer des mini-tables vides :

```
\newif\if@mtc@empty@parttoc@ \@mtc@empty@parttoc@false
\newif\if@mtc@empty@partlof@ \@mtc@empty@partlof@false
\newif\if@mtc@empty@partlot@ \@mtc@empty@partlot@false
\newif\if@mtc@empty@minitoc@ \@mtc@empty@minitoc@false
\newif\if@mtc@empty@minilof@ \@mtc@empty@minilof@false
\newif\if@mtc@empty@minilot@ \@mtc@empty@minilot@false
\newif\if@mtc@empty@secttoc@ \@mtc@empty@secttoc@false
\newif\if@mtc@empty@sectlof@ \@mtc@empty@sectlof@false
\newif\if@mtc@empty@sectlot@ \@mtc@empty@sectlot@false
```

5.5.3 Présence ou absence de certaines commandes de sectionnement

Nous définissons et positionnons des indicateurs concernant la présence des commandes de sectionnement (en fait, celle des compteurs associés à ces commandes).

\if@mtc@part@def@

Le compteur `part` :

```
\newif\if@mtc@part@def@ \@mtc@part@def@false
  \@ifundefined{part}{\@mtc@part@def@false}{\@mtc@part@def@true}
```

\if@mtc@chapter@def@ Le compteur chapter :

```
\newif\if@mtc@chapter@def@ \@mtc@chapter@def@false
  \@ifundefined{chapter}{\@mtc@chapter@def@false}{\@mtc@chapter@def@true}
```

\if@mtc@section@def@ Le compteur section :

```
\newif\if@mtc@section@def@ \@mtc@section@def@false
  \@ifundefined{section}{\@mtc@section@def@false}{\@mtc@section@def@true}
```

Nous définissons et positionnons des indicateurs concernant l'absence des commandes de sectionnement :

\if@mtc@part@undef@ Le compteur part :

```
\newif\if@mtc@part@undef@ \@mtc@part@undef@true
   \@ifundefined{part}{\@mtc@part@undef@true}{\@mtc@part@undef@false}
```

\if@mtc@chapter@undef@ Le compteur chapter :

```
\newif\if@mtc@chapter@undef@ \@mtc@chapter@undef@true
   \@ifundefined{chapter}{\@mtc@chapter@undef@true}{\@mtc@chapter@undef@false}
```

\if@mtc@section@undef@ Le compteur section :

```
\newif\if@mtc@section@undef@ \@mtc@section@undef@true
   \@ifundefined{section}{\@mtc@section@undef@true}{\@mtc@section@undef@false}
```

5.5.4 Indicateurs à tester pour savoir si certaines commandes ont été utilisées

Nous définissons une paire d'indicateurs pour chaque type de mini-table : un pour la commande elle-même et un pour la commande de préparation (\do...). Ces indicateurs seront utilisés par l'option de paquetage hints (section 5.76 page 191).

\if@parttoc@used@
\if@partlof@used@
\if@partlot@used@
\if@doparttoc@used@
\if@dopartlof@used@
\if@dopartlot@used@

Pour le niveau partie :

```
\newif\if@parttoc@used@ \global\@parttoc@used@false
\newif\if@partlof@used@ \global\@partlof@used@false
\newif\if@partlot@used@ \global\@partlot@used@false
\newif\if@doparttoc@used@ \global\@doparttoc@used@false
\newif\if@dopartlof@used@ \global\@dopartlof@used@false
\newif\if@dopartlot@used@ \global\@dopartlot@used@false
```

\if@minitoc@used@
\if@minilof@used@
\if@minilot@used@
\if@dominitoc@used@
\if@dominilof@used@
\if@dominilot@used@

Pour le niveau chapitre :

```
\newif\if@minitoc@used@ \global\@minitoc@used@false
\newif\if@minilof@used@ \global\@minilof@used@false
\newif\if@minilot@used@ \global\@minilot@used@false
\newif\if@dominitoc@used@ \global\@dominitoc@used@false
\newif\if@dominilof@used@ \global\@dominilof@used@false
\newif\if@dominilot@used@ \global\@dominilot@used@false
```

\if@secttoc@used@
\if@sectlof@used@
\if@sectlot@used@
\if@dosecttoc@used@
\if@dosectlof@used@
\if@dosectlot@used@

Pour le niveau section :

```
\newif\if@secttoc@used@ \global\@secttoc@used@false
\newif\if@sectlof@used@ \global\@sectlof@used@false
\newif\if@sectlot@used@ \global\@sectlot@used@false
\newif\if@dosecttoc@used@ \global\@dosecttoc@used@false
\newif\if@dosectlof@used@ \global\@dosectlof@used@false
\newif\if@dosectlot@used@ \global\@dosectlot@used@false
```

\if@firstpartis@used@
\if@firstchapteris@used@
\if@firstsectionis@used@

Nous détectons aussi l'utilisation de certaines commandes obsolètes :

```
\newif\if@firstpartis@used@ \global\@firstpartis@used@false
\newif\if@firstchapteris@used@ \global\@firstchapteris@used@false
\newif\if@firstsectionis@used@ \global\@firstsectionis@used@false
```

5.6 Préparation pour l'option `notoccite`

\mtc@hook@beforeinputfile
\if@mtc@notoccite@

Nous déclarons un indicateur pour la présence de cette option et la nouvelle commande interne `\mtc@hook@beforeinputfile`, qui est une commande « *hook* » (point d'attache, commande redéfinissable) utilisée par cette option (ceci a été demandé par Donald Arseneau pour son paquetage `notoccite` [2]). Voir la section 1.4.6 page 38.

```
\newif\if@mtc@notoccite@ \@mtc@notoccite@false
\@ifundefined{mtc@hook@beforeinputfile}%
   {\let\mtc@hook@beforeinputfile\relax}{}
```

5.7 Préparation pour les options `tight` and `k-tight`

\iftightmtc
\ifktightmtc

Nous déclarons simplement un indicateur pour chacune de ces options ; ils sont faux par défaut (options `loose` et `k-loose`) :

```
\newif\iftightmtc \tightmtcfalse
\newif\ifktightmtc \ktightmtcfalse
```

5.8 Préparation pour travailler avec hyperref

\AtBeginDocument
\if@mtc@hyper@used@

Ce code prépare l'interface avec le paquetage `hyperref` [39]. Un indicateur est défini, puis cette préparation est effectuée dans un bloc `\AtBeginDocument` si ce paquetage est chargé. Cette action définit quelques commandes pour le paquetage `hyperref`.

```
\PackageInfo{minitoc}{*** compatible with hyperref ***\@gobble}
\newif\if@mtc@hyper@used@ \global\@mtc@hyper@used@false
\AtBeginDocument{%
  \@ifpackageloaded{hyperref}{%
    \global\@mtc@hyper@used@true
    \def\toclevel@xpart{1000}%
    \def\toclevel@xchapter{1000}%
    \def\toclevel@xsect{1000}%
    \let\toclevel@starpart\toclevel@part
    \let\toclevel@starchapter\toclevel@chapter
    \let\toclevel@starsection\toclevel@section
    \let\toclevel@starsubsection\toclevel@subsection
    \let\toclevel@starsubsubsection\toclevel@subsubsection
    \let\toclevel@starparagraph\toclevel@paragraph
    \let\toclevel@starsubparagraph\toclevel@subparagraph
  }{}%
}
```

5.9 Tester si le paquetage sectsty est chargé, et quand

\AtBeginDocument
\if@mtc@sectstyLoaded@
\if@mtc@sectstyLoaded@a@

Nous devons tester si le paquetage `sectsty` [32] est chargé avant ou après `minitoc`, donc nous testons lors du chargement de `minitoc` et aussi dans un bloc `\AtBeginDocument`, une fois que tous les paquetages ont été chargés. Voir la section 5.76.2.6 page 205.

```
 \@ifpackageloaded{sectsty}{\@mtc@sectstyLoaded@true}{}
 \AtBeginDocument{\@ifpackageloaded{sectsty}{\@mtc@sectstyLoaded@a@true}{}}
```

5.10 La classe memoir est-elle chargée ?

\if@mtc@memoirLoaded@
\if@mtc@memoirnew@
\if@mtcpatchmemoir@

Nous testons si la classe memoir [48] est chargée. Cette classe nécessite certains ajustements pour la compatibilité ou être incompatible si elle est trop récente. Dans ce dernier cas, une correction est insérée (voir le chapitre 7 page 224).

```
\newif\if@mtcpatchmemoir@ \@mtcpatchmemoir@false
\@ifclassloaded{memoir}%
  {\@mtc@memoirLoaded@true\relax%
  \PackageInfo{minitoc}%
     {*** the memoir class is loaded: compatibility attempted ***\@gobble}}%
  {\@mtc@memoirLoaded@false}
\if@mtc@memoirLoaded@
 \@ifundefined{@m@mchapter}%
  {\@mtc@memoirnew@false\PackageInfo{minitoc}%
     {*** old version of the memoir class ***\@gobble}}
  {\@mtc@memoirnew@true\PackageInfo{minitoc}%
     {*** recent version of the memoir class ***\@gobble}
     \PackageInfo{minitoc}{*** This version of the memoir class uses \MessageBreak
                               a version of \string\chapter\space which is \MessageBreak
                               incompatible with the minitoc package. \MessageBreak
 We try to patch ***\@gobble}%
  \@mtcpatchmemoir@true}
\fi
```

\if@mtcpatchmemoir@

Et voici cette correction :

```
\if@mtcpatchmemoir@
\InputIfFileExists{mtcpatchmem.sty}{}{%
  \Package{Error}{minitoc}
    {*** Unable to patch the memoir class ***}%
    {So it remains incompatible. Sorry.}}
\fi
```

5.11 Tester la vacuité d'un fichier

\mtc@ifmtarg
\mtc@xifmtarg
\mtc@EndWhile
\mtc@WhilePreCondition
\mtc@WhileCondition
\mtc@WhileBody
\mtc@While
\mtc@WhileNext

Quelques macros pour tester si un argument d'une macro est vide (elles sont tirées du paquetage ifmtarg [49], de Peter R. Wilson et Donald Arseneau, et de while.tip, de Stephan P. von Bechtolsheim [42]). Le groupe est nécessaire pour garder local le changement de « *catcode* » de « Q », donc il est nécessaire d'utiliser un \gdef pour \mtc@ifmtarg.

```
\begingroup
\catcode`\Q=3
\long\gdef\mtc@ifmtarg#1{%
\mtc@xifmtarg#1QQ\@secondoftwo\@firstoftwo\@nil}
```

```
\long\gdef\mtc@xifmtarg#1#2Q#3#4#5\@nil{#4}
\endgroup
\let\mtc@EndWhile = \fi
\def\mtc@While #1#2#3\mtc@EndWhile{%
   \def\mtc@WhilePreCondition{#1}%
   \def\mtc@WhileCondition{#2}%
   \def\mtc@WhileBody{#3}%
   \mtc@@While
}
\def\mtc@@While{%
   \mtc@WhilePreCondition
   \mtc@WhileCondition
      \def\mtc@WhileNext{%
          \mtc@WhileBody
          \mtc@@While
      }%
   \else
      \def\mtc@WhileNext{}%
   \fi
   \mtc@WhileNext
}
```

\if@mtc@checkfiles
\if@mtc@FE
\if@mtc@LI
\mtc@While
\mtc@Body
\mtc@EndWhile
\mtc@CkFile
\mtc@CkStr
\mtc@Rline
\tf@mtc
\@inputcheck

Voici quelques macros pour tester si un fichier est vide ou non : `\mtc@CkFile{file}` rend `\@mtc@FEtrue` si le fichier est vide, `\@mtc@FEfalse` si le fichier n'est pas vide. Un fichier inexistant est vide. Un fichier rempli d'espace blanc (espace, tabulation, saut de ligne) est vide. Les commentaires sont vides.

Note : sur un gros fichier vide, la boucle `\mtc@While` peut prendre du temps, mais pas une éternité (33 s pour 10^6 lignes avec mon ordinateur), et la première ligne non vide arrête la boucle. `\jobname.mtc` est utilisé comme fichier de travail. Il est effacé après usage.

```
\newif\if@mtc@LI\@mtc@LItrue
\newif\if@mtc@FE\@mtc@FEtrue
\newif\if@mtc@checkfiles\@mtc@checkfilestrue
\def\mtc@Body{\immediate\read\@inputcheck to
                                     \mtc@Rline\relax
  \ifeof\@inputcheck\relax\@mtc@LIfalse\fi
  \expandafter\ifx\mtc@Rline\par\relax
     \def\mtc@Rline{}
  \else
     \ifeof\@inputcheck\relax\global\@mtc@LIfalse\fi
     \mtc@ifmtarg{\mtc@Rline}{\relax}%
              {\@mtc@FEfalse\@mtc@LIfalse}
  \fi
}
\def\mtc@CkFile#1{%
\@mtc@LItrue\@mtc@FEtrue
\if@mtc@checkfiles
\IfFileExists{#1}{%
\immediate\openin\@inputcheck #1\relax
\mtc@While{}{\if@mtc@LI\relax}%
```

```
  {\mtc@Body}%
\mtc@EndWhile}%
{\@mtc@FEtrue}%
\else
\@mtc@FEfalse%
\fi}
\closein\@inputcheck\relax
\def\mtc@CkStr#1{%
\immediate\openout\tf@mtc \jobname.mtc
\immediate\write\tf@mtc{#1}%
\immediate\closeout\tf@mtc
\mtc@CkFile{\jobname.mtc}%
\immediate\openout\tf@mtc \jobname.mtc
\immediate\closeout\tf@mtc}
```

5.12 Macros internes pour décrémenter les compteurs de minitoc

\mtc@onebackpart
\mtc@onebackchapter
\mtc@onebacksection
\addtocounter

Il est parfois nécessaire de décrémenter un compteur de minitoc (ptc, mtc ou stc) de un. Ces macros sont :

```
\def\mtc@onebackpart{\addtocounter{ptc}{-1}}
\def\mtc@onebackchap{\addtocounter{mtc}{-1}}
\def\mtc@onebacksect{\addtocounter{stc}{-1}}
```

5.13 Modifier la commande \part

\part
\mtc@svspart
\mtc@svpart
\@spart
\@part
\stepcounter

Si la commande \part n'est pas définie (par la classe de document, habituellement), nous ne pouvouns pas la modifier et un avertissement est émis [3]. Sinon, nous modifions ses deux branches, \@part (pour la version non étoilée) ou \@spart (pour la version étoilée, \part*) : nous ajoutons \stepcounter{ptc} pour incrémenter le compteur de parttocs ptc. Voir aussi la section 5.51 page 132.

Le code de la section suivante (section 5.14 page suivante) est aussi sauté si \part n'est pas définie.

```
\@ifundefined{part}{%
  \PackageWarningNoLine{minitoc}%
    {*** part level macros NOT available ***}
}{%% else undefined part (\part defined)
  \PackageInfo{minitoc}%
```

[3] Une classe de document ayant des commandes de sectionnement mais pas de commande \part est probablement non standard, d'où l'avertissement affiché sur le terminal.

```
    {*** part level macros available ***\@gobble}
\let\mtc@svspart\@spart
\def\@spart{\stepcounter{ptc}\mtc@svspart}
\let\mtc@svpart\@part
\def\@part{\stepcounter{ptc}\mtc@svpart}
```

5.14 Ajout d'une entrée dans la table des matières pour une partie étoilée

\mtcaddpart \mtc@ifmtarg \contentsline \addcontentsline \adjustptc \l@xpart \l@part

Pour ajouter une entrée dans la table des matières pour une partie étoilée, nous avons besoin de la macro `\mtcaddpart`, qui a un argument optionnel, le titre de la partie tel qu'il devrait apparaître dans la table des matières.

Par défaut, cet argument est vide. S'il est vide (le test est fait par `\mtc@ifmtarg`) ou omis, nous ajoutons une ligne `\contentsline{xpart}{}...` dans le fichier `.toc`. S'il n'est pas vide, nous ajoutons une ligne `\contentsline{part}{titre...}...` dans le fichier `.toc`. Nous ajoutons toujours une ligne `\contentsline{xpart}{}...` dans les fichiers `.lof` et `.lot`. Puis nous incrémentons le compteur `ptc`, via `\adjustptc` (définie dans la section 5.45 page 118). Utiliser `xpart` comme premier argument de `\contentsline` signifie que `\l@xpart` sera invoquée au lieu de `\l@part` pour imprimer l'entrée dans la table des matières, mais `\l@xpart` utilise une très grande profondeur (10000) pour cette entrée, donc celle-ci ne sera jamais vraiment imprimée (sauf si vous trichez).

```
\newcommand{\mtcaddpart}[1][]{%
    \mtc@ifmtarg{#1}{\addcontentsline{toc}{xpart}{}}%
                    {\addcontentsline{toc}{part}{#1}}%
    \addcontentsline{lof}{xpart}{}%
    \addcontentsline{lot}{xpart}{}%
    \adjustptc}
```

Ce code termine (temporairement) les commandes au niveau partie.

```
}%
```

5.15 Macros au niveau section

\chapter \section

Les macros au niveau section sont définies si `\chapter` n'est pas définie mais `\section` est définie, c'est-à-dire dans les classes de document telles que `article`, mais pas dans les classes de document telles que `book` ou `report`. Donc nous testons si `\chapter` est définie et si `\section` est définie, avec les avertissements adéquats. Si aucune de ces deux commandes n'est définie, vous avez un sérieux problème pour utiliser le paquetage `minitoc` avec la classe de votre document.

```
\@ifundefined{chapter}{\PackageInfo{minitoc}%
          {*** chapter level macros NOT available ***\@gobble}%
  \@ifundefined{section}{\PackageInfo{minitoc}%
          {*** section level macros NOT available ***\@gobble}%
    \PackageWarningNoLine{minitoc}%
       {*** no section or chapter level macros available ***
        \MessageBreak
        *** PLEASE VERIFY YOUR MAIN DOCUMENT CLASS ***}}%
  {\PackageInfo{minitoc}%
       {*** section level macros available ***\@gobble}%
```

5.16 Corrections pour la numérotation

\mtc@onebacksect \tableofcontents \listoffigures \listoftables \mtcsv@tableofcontents \mtcsv@listoffigures \mtcsv@listoftables

Comme la table des matières, la liste des figures et la liste des tableaux sont considérées comme des sections (étoilées), il nous faut décrémenter le compteur de secttocs (`stc`) via `\mtc@onebacksect` lorsque les commandes correspondantes sont exécutées. Donc nous modifions ces commandes :

```
    \let\mtcsv@tableofcontents\tableofcontents
    \let\mtcsv@listoffigures\listoffigures
    \let\mtcsv@listoftables\listoftables
    \def\tableofcontents{\mtcsv@tableofcontents\mtc@onebacksect}
    \def\listoffigures{\mtcsv@listoffigures\mtc@onebacksect}
    \def\listoftables{\mtcsv@listoftables\mtc@onebacksect}
```

5.17 Modification de la commande `\section`

\mtc@svsection \mtc@svss \@ssect \@sect \section \stepcounter

Si la commande `\section` n'est pas définie (par la classe de document, habituellement), nous ne pouvons pas la modifier et un avertissement est affiché. Sinon, nous modifions ses deux branches, `\@sect` (pour la version non étoilée) ou `\@ssect` (pour la version étoilée, `\section*`) : nous ajoutons `\stepcounter{stc}` pour incrémenter le compteur de secttocs `stc`, seulement dans le cas non étoilé (la version #25 a éliminé une décrémententation superflue de ce compteur).

```
     \let\mtc@svsection\section
     \def\section{\stepcounter{stc}\mtc@svsection}
     \let\mtc@svss\@ssect
```

5.18 Ajout d'une entrée dans la table des matières pour une section étoilée

\mtcaddsection \mtc@ifmtarg \contentsline \adjuststc \l@xsection \l@section

Pour ajouter une entrée dans la table des matières pour une section étoilée, nous avons besoin de la macro `\mtcaddsection`, qui a un argument optionnel, le titre de la section tel qu'il doit apparaître dans la table des matières. Par défaut, cet argument est vide. S'il est vide (ce qui est testé par `\mtc@ifmtarg`) ou omis, nous ajoutons une ligne `\contentsline{xsection}{}...` dans le fichier `.toc`. S'il n'est pas vide, nous ajoutons une ligne `\contentsline{section}{title...}...` dans le fichier `.toc`. Nous ajoutons toujours une ligne `\contentsline{xsection}{}...` dans les fichiers `.lof` et `.lot`. Puis nous incrémentons le compteur `stc`, via `\adjuststc` (définie dans la section 5.54 page 142). Le fait d'utiliser `xsection` comme premier argument de `\contentsline` signifie que `\l@xsection` sera invoquée au lieu de `\l@section` pour imprimer l'entrée dans la table des matières, mais `\l@xsection` utilise une très grande profondeur (10000) pour cette entrée, donc elle ne sera jamais réellement imprimée (sauf si vous trichez).

```
\newcommand{\mtcaddsection}[1][]{%
    \mtc@ifmtarg{#1}{\addcontentsline{toc}{xsection}{}}%
                    {\addcontentsline{toc}{section}{#1}}%
    \addcontentsline{lof}{xsection}{}%
    \addcontentsline{lot}{xsection}{}%
    \adjuststc}
```

Ce code termine (temporairement) les commandes au niveau section et nous continuons avec les commandes au niveau chapitre.

```
}}%
 {%
```

5.19 Macros au niveau chapitre

\chapter

Les macros au niveau chapitre sont définies si `\chapter` est définie, c'est-à-dire, dans les classes de document telles que `book` ou `report`. Donc nous testons si `\chapter` est définie, avec les avertissements adéquats. Ce test a déjà été fait plus haut, nous sommes dans la branche « sinon » de `\@ifundefined{chapter}`.

```
\PackageInfo{minitoc}{*** chapter level macros available ***\@gobble}
```

5.20 Modifier la commande `\chapter`

`\chapter` `\@chapter` `\mtc@svchapter` `\stepcounter`

La commande `\chapter` est définie (par la classe de document, habituellement). Nous modifions ses deux branches, `\@chapter` (pour la version non étoilée) ou `\@schapter` (pour la version étoilée, `\chapter*`) : nous ajoutons un appel à `\stepcounter{mtc}` pour incrémenter le compteur de minitocs `mtc`. Seule la branche non étoilée est modifiée ici. L'autre branche sera modifiée plus loin (voir la section 5.36 page 103).

```
\let\mtc@svchapter\@chapter
\def\@chapter{\stepcounter{mtc}\mtc@svchapter}
```

5.21 Ajout d'une entrée dans la table des matières pour un chapitre étoilé

`\mtcaddchapter` `\mtc@ifmtarg` `\contentsline` `\adjustmtc` `\l@xchapter` `\l@chapter`

Pour ajouter une entrée dans la table des matières pour un chapitre étoilé, nous avons besoin de la macro `\mtcaddchapter`, qui a un argument optionnel, le titre du chapitre tel qu'il devrait apparaître dans la table des matières. Par défaut, cet argument est vide. S'il est vide (ce qui est testé via `\mtc@ifmtarg`) ou omis, nous ajoutons une ligne `\contentsline{xchapter}{}...` dans le fichier `.toc`. S'il n'est pas vide, nous ajoutons une ligne `\contentsline{chapter}{titre...}...` dans le fichier `.toc`. Nous ajoutons toujours une ligne `\contentsline{xchapter}{}...` dans les fichiers `.lof` et `.lot`. Puis nous incrémentons le compteur `mtc`, via `\adjustmtc` (définie dans la section 5.31 page 90). Le fait d'utiliser `xchapter` comme premier argument de `\contentsline` signifie que `\l@xchapter` sera invoquée au lieu de `\l@chapter` pour imprimer l'entrée dans la table des matières, mais `\l@xchapter` utilise une très grande profondeur (10000) pour cette entrée, et donc elle ne sera jamais vraiment imprimée (sauf si vous trichez).

```
\newcommand{\mtcaddchapter}[1][]{%
    \mtc@ifmtarg{#1}{\addcontentsline{toc}{xchapter}{}}%
                    {\addcontentsline{toc}{chapter}{#1}}%
    \addcontentsline{lof}{xchapter}{}%
    \addcontentsline{lot}{xchapter}{}%
    \adjustmtc}
```

Ce code termine (temporairement) les commandes au niveau chapitre, c'est-à-dire termine la construction `\@ifundefined{chapter}` du début de la section 5.15 page 78.

```
}%
```

5.22 Déclarations diverses

\newread \mtc@toks \mtc@string \mtc@strut \mtc@strutbox \mtc@hstrut \mtc@hstrutbox \mtc@v \mtc@zrule \mtc@BBR

La commande \newread doit être redéclarée comme étant \outer (comme me l'a conseillé Donald ARSENEAU). Nous avons besoin d'un registre *token* (\mtc@toks), d'une chaîne temporaire (\mtc@string), d'étais ou *struts* (de deux sortes, dont chacune utilise une boîte contenant un filet vertical invisible) un filet dont toutes les dimensions sont à zéro (\mtc@zrule) et d'une commande pour décourager les coupures de page (\mtc@BBR, pour « *bad break* »). Pour les étais, qui sont des boîtes contenant un filet vertical invisible, nous utilisons des unités « ex », pour suivre la fonte courante.

```
\def\newread{\alloc@6\read\chardef\sixt@@n}
\newtoks\mtc@toks
\def\mtc@string{\relax}
\newbox\mtc@strutbox
\setbox\mtc@strutbox=\hbox{\rule[1.8ex]{\z@}{2.5ex}}
\def\mtc@strut{\relax\ifmmode\copy\mtc@strutbox
                     \else\unhcopy\mtc@strutbox\fi}
\newbox\mtc@hstrutbox
\setbox\mtc@hstrutbox=\hbox{\rule[1.ex]{\z@}{1.ex}}
\def\mtc@hstrut{\relax\ifmmode\copy\mtc@hstrutbox
                      \else\unhcopy\mtc@hstrutbox\fi}
\def\mtc@v{\leavevmode\mtc@strut}
\def\mtc@zrule{\rule[\z@]{\z@}{\z@}}
\def\mtc@BBR{\unpenalty\nopagebreak[4]}
```

5.23 Autoconfiguration des suffixes

\tf@mtc

Ce code est une astuce pour déterminer si le système d'exploitation est capable ou non d'utiliser des suffixes longs (>3 caractères) dans les noms de fichiers. Nous définissons un descripteur de fichier (\tf@mtc) pour écrire des fichiers [4]. Ce code est bavard si les suffixes longs ne peuvent pas être utilisés, sinon les messages sont écrits seulement dans le fichier .log. Le séquencement de ces opérations est vital. La figure 5.1 page suivante montre ce séquencement. Un astérisque ($*$) indique quel fichier est lu lors de la phase 3.

[4] Notez que c'est le *seul* nouveau descripteur de fichier créé par le paquetage minitoc. Tous les fichiers écrits par minitoc utilisent ce descripteur, ou l'un des descripteurs standard, comme pour le fichier .log. En fait, minitoc écrit aussi dans les fichiers .toc, .lof et .lot, mais via des descripteurs de fichiers déjà utilisés par des commandes standard telle que \tableofcontents, \listoffigures et \listoftables. Nous pouvons en conclure que minitoc lui-même utilise un seul descripteur de fichier (ou flux d'écriture). Certaines autres tentatives pour faire des tables des matières par chapitre ont échoué en conduisant rapidement à un épuisement des descripteurs de fichiers (TEX offre seulement 16 descripteurs de fichiers en écriture), parce qu'elles appelaient la macro interne standard \@starttoc, qui invoque \newwrite, pour chaque mini-table. Puisque minitoc écrit dans seulement un seul fichier à la fois (et dans le fichier .log, et dans les fichiers standard de contenus, bien sûr), nous pouvons réutiliser le même descripteur de fichier et éviter ce sérieux problème. Le paquetage minitoc écrit dans les fichiers de contenus lorsqu'il rencontre une commande de sectionnement majeure (\part, \chapter, ou \section), si nécessaire. Il écrit dans les fichiers auxiliaires pour les mini-tables seulement via les commandes de préparation des mini-tables (\doparttoc, …, \dosectlot), un à la fois. Vous n'avez pas besoin d'un nouveau marteau pour chaque clou.

Phase (temps de gauche à droite) :		1	2	3
S.E. avec suffixes longs	\jobname.mtc1	VRAI	VRAI	*
	\jobname.mtc		FAUX	
S.E. avec suffixes courts	\jobname.mtc(1)	VRAI	FAUX	*

FIG. 5.1 – Astuce pour détecter la limitation à des suffixes courts.

\if@longextensions@
\tf@mtc
\openout
\write
\closeout
\input
\jobname

(0) D'abord, un message et un nouvel indicateur :

```
\PackageInfo{minitoc}%
   {*** Autoconfiguration of extensions ***\@gobble}
\newif\if@longextensions@\@longextensions@false
```

(1) Nous écrivons « `\@longextensions@true` » dans `\jobname.mtc1`. Mais si le système d'exploitation a des suffixes courts, le vrai nom du fichier sera tronqué à `\jobname.mtc`.

```
\immediate\openout\tf@mtc \jobname.mtc1
\immediate\write\tf@mtc{\string\@longextensions@true}
\immediate\closeout\tf@mtc
```

(2) Nous écrivons « `\@longextensions@false` » dans `\jobname.mtc`.

```
\immediate\openout\tf@mtc \jobname.mtc
\immediate\write\tf@mtc{\string\@longextensions@false}
\immediate\closeout\tf@mtc
```

(3) Nous lisons `\jobname.mtc1`. Mais si le système d'exploitation a des suffixes courts, le vrai nom du fichier sera tronqué à `\jobname.mtc`.

```
\input{\jobname.mtc1}
```

(4) En conséquence, l'indicateur est vrai si nous lisons effectivement depuis le fichier `\jobname.mtc1`, mais faux si nous lisons depuis `\jobname.mtc`. Le texte et la sévérité des messages sont différents.

```
\if@longextensions@
 \PackageInfo{minitoc}%
   {*** Long extensions (Unix-like) will be used ***\@gobble}
 \PackageInfo{minitoc}%
 {==> this version is configured for UNIX-like \MessageBreak
   \space\space\space\space(long extensions) file names\@gobble}%
\else
 \PackageWarningNoLine{minitoc}%
   {*** Short extensions (MSDOS-like) will be used ***sight***}
 \PackageWarningNoLine{minitoc}%
   {==> this version is configured for MSDOS-like
   \MessageBreak \space\space\space\space(8+3) file names}
\fi
```

(5) Nous effaçons le contenu des deux fichiers (car `\jobname.mtc` est aussi utilisé comme fichier de travail plus tard, voir la section 5.11 page 75).

```
\immediate\openout\tf@mtc \jobname.mtc
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc \jobname.mtc1
\immediate\closeout\tf@mtc
```

5.24 Détection des versions obsolètes de LaTeX

\@inputcheck
\reset@font

Ce code détecte les vieilles versions du noyau LaTeX qui ne sont plus maintenues et avec lesquelles le paquetage `minitoc` auront beaucoup de mal à fonctionner. L'astuce est de détecter l'absence de certaines commandes internes de LaTeX, `\@inputcheck` et `\reset@font`. Si vous recevez l'un de ces messages, vous êtes dans une mauvaise situation et devriez, *dans l'urgence,* mettre à jour votre installation de LaTeX, qui rouille depuis… pas mal d'années !

```
\@ifundefined{@inputcheck}%
  {\PackageWarningNoLine{minitoc}%
     {Your version of latex.tex is obsolete.\MessageBreak
     Trying to continue..}\newread\@inputcheck\relax}{}
\@ifundefined{reset@font}%
  {\PackageWarningNoLine{minitoc}%
     {Your version of latex.tex is very obsolete.\MessageBreak
     Trying to continue... crossing fingers}%
  \let\reset@font\relax}{}
```

5.25 Une macro pour faire une entrée dans la table des matières sans points de conduite ni numéros de page

\@undottedtocline

La macro (interne) `\@undottedtocline` est une version modifiée de la commande standard `\@dottedtocline`. Elle sera utilisée dans les macros de personnalisation.

```
\def\@undottedtocline#1#2#3#4#5{%
  \ifnum #1>\c@tocdepth \else
  \vskip \z@ plus.2\p@
  {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
   \parindent #2\relax\@afterindenttrue
   \interlinepenalty\@M
   \leavevmode
   \@tempdima #3\relax \advance\leftskip \@tempdima \hbox{}%
   \hskip -\leftskip
    #4\nobreak\hfill \nobreak
          \null\par}%
  \fi}
```

5.26 Valeurs par défaut pour les macros de personnalisation des numéros de page

\if@mtc@memoirLoaded@

Cette section définit quelques macros de personnalisation concernant la présence ou l'absence des numéros de pages dans les mini-tables. Mais si la classe `memoir` [48] est chargée, elle sait faire ce travail. Donc, nous testons d'abord `\if@mtc@memoirLoaded@` pour utiliser les commandes de `memoir` lorsqu'elles sont disponibles.

```
\if@mtc@memoirLoaded@
```

`\mtcpagenumbers` `\nomtcpagenumbers` Pour les entrées dans les minitocs :

```
\def\mtcpagenumbers{%
  \cftpagenumberson{section}
  \cftpagenumberson{subsection}
  \cftpagenumberson{subsubsection}
  \cftpagenumberson{paragraph}
  \cftpagenumberson{subparagraph}}
\def\nomtcpagenumbers{%
  \cftpagenumbersoff{section}
  \cftpagenumbersoff{subsection}
  \cftpagenumbersoff{subsubsection}
  \cftpagenumbersoff{paragraph}
  \cftpagenumbersoff{subparagraph}}
```

`\stcpagenumbers` `\nostcpagenumbers` Pour les entrées dans les secttocs :

```
\def\stcpagenumbers{%
  \cftpagenumberson{subsection}
  \cftpagenumberson{subsubsection}
  \cftpagenumberson{paragraph}
  \cftpagenumberson{subparagraph}}
\def\nostcpagenumbers{%
  \cftpagenumbersoff{subsection}
  \cftpagenumbersoff{subsubsection}
  \cftpagenumbersoff{paragraph}
  \cftpagenumbersoff{subparagraph}}
```

`\ptcpagenumbers` `\noptcpagenumbers` Pour les entrées dans les parttocs :

```
\def\ptcpagenumbers{%
  \cftpagenumberson{chapter}
  \cftpagenumberson{section}
  \cftpagenumberson{subsection}
  \cftpagenumberson{subsubsection}
  \cftpagenumberson{paragraph}
  \cftpagenumberson{subparagraph}}
\def\noptcpagenumbers{%
  \cftpagenumbersoff{chapter}
  \cftpagenumbersoff{section}
  \cftpagenumbersoff{subsection}
  \cftpagenumbersoff{subsubsection}
  \cftpagenumbersoff{paragraph}
  \cftpagenumbersoff{subparagraph}}
```

\mlfpagenumbers
\nomlfpagenumbers
\slfpagenumbers
\noslfpagenumbers
\plfpagenumbers
\noplfpagenumbers

Pour les entrées dans les minilofs, sectlofs et partlofs :

```
 \def\mlfpagenumbers{\cftpagenumberson{figure}}
 \def\nomlfpagenumbers{\cftpagenumbersoff{figure}}
 \def\slfpagenumbers{\cftpagenumberson{figure}}
 \def\noslfpagenumbers{\cftpagenumbersoff{figure}}
 \def\plfpagenumbers{\cftpagenumberson{figure}}
 \def\noplfpagenumbers{\cftpagenumbersoff{figure}}
```

\mltpagenumbers
\nomltpagenumbers
\sltpagenumbers
\nosltpagenumbers
\pltpagenumbers
\nopltpagenumbers

Pour les entrées dans les minilots, sectlots et partlots :

```
 \def\mltpagenumbers{\cftpagenumberson{table}}
 \def\nomltpagenumbers{\cftpagenumbersoff{table}}
 \def\sltpagenumbers{\cftpagenumberson{table}}
 \def\nosltpagenumbers{\cftpagenumbersoff{table}}
 \def\pltpagenumbers{\cftpagenumberson{table}}
 \def\nopltpagenumbers{\cftpagenumbersoff{table}}
```

Sinon, minitoc utilisera ses propres commandes.

```
\else
```

\mtcpagenumbers
\nomtcpagenumbers
\mlfpagenumbers
\nomlfpagenumbers
\mltpagenumbers
\nomltpagenumbers

D'abord, pour les minitocs, secttocs et parttocs :

```
\def\mtcpagenumbers{\let\mtc@pgno\null}
\def\nomtcpagenumbers{\let\mtc@pgno\relax}
\def\stcpagenumbers{\let\stc@pgno\null}
\def\nostcpagenumbers{\let\stc@pgno\relax}
\def\ptcpagenumbers{\let\ptc@pgno\null}
\def\noptcpagenumbers{\let\ptc@pgno\relax}
```

\mlfpagenumbers
\nomlfpagenumbers
\mlfpagenumbers
\nomlfpagenumbers
\mltpagenumbers
\nomltpagenumbers

Puis, pour les minilofs, sectlofs et partlofs :

```
\def\mlfpagenumbers{\let\mlf@pgno\null}
\def\nomlfpagenumbers{\let\mlf@pgno\relax}
\def\slfpagenumbers{\let\slf@pgno\null}
\def\noslfpagenumbers{\let\slf@pgno\relax}
\def\plfpagenumbers{\let\plf@pgno\null}
\def\noplfpagenumbers{\let\plf@pgno\relax}
```

\mltpagenumbers
\nomltpagenumbers
\mltpagenumbers
\nomltpagenumbers
\mltpagenumbers
\nomltpagenumbers

Puis, pour les minilots, sectlots et partlots :

```
\def\mltpagenumbers{\let\mlt@pgno\null}
\def\nomltpagenumbers{\let\mlt@pgno\relax}
\def\sltpagenumbers{\let\slt@pgno\null}
```

```
 \def\nosltpagenumbers{\let\slt@pgno\relax}
 \def\pltpagenumbers{\let\plt@pgno\null}
 \def\nopltpagenumbers{\let\plt@pgno\relax}
\fi
```

\ptcpagenumbers
\plfpagenumbers
\pltpagenumbers
\mtcpagenumbers
\mlfpagenumbers
\mltpagenumbers
\stcpagenumbers
\slfpagenumbers
\sltpagenumbers

Et enfin, les valeurs par défaut sont établies ; les numéros de pages sont présents :

```
\ptcpagenumbers
\plfpagenumbers
\pltpagenumbers
\mtcpagenumbers
\mlfpagenumbers
\mltpagenumbers
\stcpagenumbers
\slfpagenumbers
\sltpagenumbers
```

5.27 « Dispositifs » pour les mini-tables

À chaque type de mini-table sont associés trois « dispositifs » (*features*) : un dispositif « before » (avant), un dispositif « after » (après) et un dispositif « thispagestyle » (style de cette page).

Un dispositif « before » est défini par une macro telle que `\beforeparttoc` qui contient du code à exécuter avant toute mini-table d'un type donné : `\beforeparttoc` est exécutée avant chaque parttoc. Habituellement, de tels dispositifs ne contiennent que des commandes triviales telles que `\clear[double]page` ou `\empty`.

Un dispositif « after » est analogue mais son code est exécuté après chaque mini-table d'un type donné.

Un dispositif « thispagestyle » est défini par une macro comme `\thispagestyleparttoc` qui contient du code pour définir le style de page impliqué par les mini-tables d'un type donné : la commande `\thispagestyleparttoc` peut être définie comme étant `\thispagestyle{...}`. Habituellement, le dispositif « thispagestyle » n'est défini que pour les mini-tables au niveau partie, qui utilisent des sauts de page dans leurs dispositifs « before » et « after ». Pour les mini-tables aux niveaux chapitre et section, le dispositif « thispagestyle » est habituellement défini comme étant `\empty`.

Nous établissons les valeurs par défaut pour les dispositifs au niveau partie selon la présence de la commande `\chapter`, car les documents du genre `article` sont différents des documents du genre `book` ou `report` pour la mise en page des mini-tables au niveau partie.

\chapter \beforeparttoc \beforepartlof \beforepartlot

Si \chapter n'est pas définie, les mini-tables au niveau partie n'ont pas de dispositif « before » (par défaut) :

```
\@ifundefined{chapter}{%
   \let\beforeparttoc\empty
   \let\beforepartlof\empty
   \let\beforepartlot\empty}%
```

\cleardoublepage

Mais si \chapter est définie, elles ont \cleardoublepage comme dispositif « before » par défaut

```
   {\let\beforeparttoc\cleardoublepage
    \let\beforepartlof\cleardoublepage
    \let\beforepartlot\cleardoublepage}
```

\beforeminitoc \beforeminilof \beforeminilot

Les mini-tables au niveau chapitre n'ont (par défaut), pas de dispositif « before » :

```
\let\beforeminitoc\empty
\let\beforeminilof\empty
\let\beforeminilot\empty
```

\beforesecttoc \beforesectlof \beforesectlot

Les mini-tables au niveau section n'ont (par défaut), pas de dispositif « before » :

```
\let\beforesecttoc\empty
\let\beforesectlof\empty
\let\beforesectlot\empty
```

\chapter \afterparttoc \afterpartlof \afterpartlot

Si \chapter n'est pas définie, les mini-tables au niveau partie n'ont pas de dispositif « after » (par défaut) :

```
\@ifundefined{chapter}{%
   \let\afterparttoc\empty
   \let\afterpartlof\empty
   \let\afterpartlot\empty}%
```

\cleardoublepage

Mais si \chapter est définie, elles ont \cleardoublepage comme dispositif « after » par défaut

```
   {\let\afterparttoc\cleardoublepage
    \let\afterpartlof\cleardoublepage
    \let\afterpartlot\cleardoublepage}
```

\afterminitoc \afterminilof \afterminilot

Les mini-tables au niveau chapitre n'ont, par défaut, pas de dispositif « after » :

```
\let\afterminitoc\empty
\let\afterminilof\empty
\let\afterminilot\empty
```

\aftersecttoc
\aftersectlof
\aftersectlot

Les mini-tables au niveau section n'ont, par défaut, pas de dispositif « after » :

```
\let\aftersecttoc\empty
\let\aftersectlof\empty
\let\aftersectlot\empty
```

\thispagestyle
\thispageparttocstyle
\thispagepartlofstyle
\thispagepartlotstyle
\thispageminitocstyle
\thispageminilofstyle
\thispageminilotstyle
\thispagesecttocstyle
\thispagesectlofstyle
\thispagesectlotstyle

Par défaut, tous les dispositifs « thispagestyle » (au niveau partie) utilisent le style de page `empty`. Ceci affecte seulement la première page de la mini-table.

```
\def\thispageparttocstyle{\thispagestyle{empty}}
\def\thispagepartlofstyle{\thispagestyle{empty}}
\def\thispagepartlotstyle{\thispagestyle{empty}}
\def\thispageminitocstyle{\empty}
\def\thispageminilofstyle{\empty}
\def\thispageminilotstyle{\empty}
\def\thispagesecttocstyle{\empty}
\def\thispagesectlofstyle{\empty}
\def\thispagesectlotstyle{\empty}
```

\mtcsetfeature

Dans la section 5.65.8 page 178, nous définirons la macro `\mtcsetfeature` qui est une interface utilisateur bien plus facile pour mettre en place les « dispositifs » des mini-tables.

5.28 Tables des matières invisibles

\faketableofcontents
\fakelistoffigures
\fakelistoftables
\fake@starttoc

Si vous souhaitez ne pas avoir de table des matières, mais avoir quand même des minitocs, il vous faut créer le fichier `.toc`, sans l'insérer dans votre document. Cette commande `\faketableofcontents` est une version restreinte de la commande standard `\tableofcontents`. Nous définissons de la même manière les commandes `\fakelistoffigures` et `\fakelistoftables`, en utilisant en fait une version restreinte `\fake@starttoc` de `\@starttoc`. Mais il est bon de remettre à zéro maintenant les compteurs `ptc`, `mtc` et `stc`, s'ils sont définis[5].

```
\def\faketableofcontents{\fake@starttoc{toc}%
    \@ifundefined{c@ptc}{}{\setcounter{ptc}{0}}%
    \@ifundefined{c@mtc}{}{\setcounter{mtc}{0}}%
    \@ifundefined{c@stc}{}{\setcounter{stc}{0}}%
    }
\def\fakelistoffigures{\fake@starttoc{lof}}
\def\fakelistoftables{\fake@starttoc{lot}}
\def\fake@starttoc#1{\begingroup
  \makeatletter
  \if@filesw \expandafter\newwrite\csname tf@#1\endcsname
             \immediate\openout \csname tf@#1\endcsname
             \jobname.#1\relax
```

[5] Souvenez-vous de l'infâme erreur « `stc0` ».

```
\fi
\global\@nobreakfalse \endgroup}
```

Ce code utilise les mêmes descripteurs de fichier (en écriture) que les commandes d'origine.

5.29 Compteurs de profondeur pour les minilofs et les minilots

`\AtBeginDocument` `\c@lofdepth` `\c@lotdepth`

Si les compteurs `lofdepth` et `lotdepth` sont définis, nous créons les nouveaux compteurs nécessaires : `minilofdepth` et `minilofdepth`. Ces compteurs sont initialisés à 2. Ceci est fait après le chargement des paquetages, dans un bloc `\AtBeginDocument` :

```
\AtBeginDocument{%
 \@ifundefined{c@lofdepth}{}%
   {\newcounter{minilofdepth}\setcounter{minilofdepth}{2}}%
 \@ifundefined{c@lotdepth}{}%
   {\newcounter{minilotdepth}\setcounter{minilotdepth}{2}}%
}
```

5.30 Commandes au niveau chapitre

À partir d'ici, nous définissons les commandes au niveau chapitre.

`\mtc@markboth` `\@mkboth`

Tout d'abord, nous mémorisons les marques (ceci n'est pas encore utilisé, mais...) :

```
\global\let\mtc@markboth\markboth
\global\let\@mkboth\markboth
```

5.31 Parties, chapitres et section étoilés

`\addst@rred` `\addcontentsline` `\stepcounter` `\c@ptc` `\c@mtc` `\c@stc`

Nous définissons des commandes pour traiter les commandes de sectionnement étoilées : `\part*`, `\chapter*` et `\section*`. Le niveau section est différent selon que `\chapter` est définie ou non. Éventuellement, un compteur est incrémenté. Une ligne de contenu est ajoutée dans le fichier `.toc`, avec la bonne profondeur pour l'imprimer (voir `\l@star...` plus loin, dans la section 5.63 page 163).

```
\def\addst@rred#1#2{%
```

```
  \addcontentsline{toc}{star#1}{#2}%
  \@ifundefined{c@ptc}{}{%
    \expandafter\ifx\csname #1\endcsname\part\relax
      \stepcounter{ptc}%
    \fi
  }%
  \@ifundefined{c@mtc}{}{%
    \expandafter\ifx\csname #1\endcsname\chapter\relax
      \stepcounter{mtc}%
    \fi
  }%
  \@ifundefined{c@stc}{}{%
    \expandafter\ifx\csname #1\endcsname\section\relax
%%      \@ifundefined{chapter}{\stepcounter{stc}}{}%
      \stepcounter{stc}%
    \fi
  }%
}%
```

\addstarredsection
\chapter
\addst@rred

Si `\chapter` n'est pas définie, nous définissons simplement `\addstarredsection` :

```
\@ifundefined{chapter}{%
\gdef\addstarredsection#1{\addst@rred{section}{#1}}
}%
```

sinon, nous commençons à définir le matériel pour les commandes au niveau chapitre (branche « sinon » de `\@ifundefined{chapter}`) :

```
{%
```

\The@mtc
\firstchapteris
\if@firstchapteris@used@
\adjustmtc
\decrementmtc
\incrementmtc
\themtc
\mtc@rule
\mlf@rule
\mlt@rule
\plf@rule
\plt@rule
\slf@rule
\slt@rule
\mtcindent

Nous définissons ensuite : le format interne du compteur `mtc` (`\The@mtc`), la commande obsolète `\firstchapteris` (elle ne fait d'émettre un avertissement bénin), le compteur `mtc` (initialisé à 0), la commande `\adjustmtc` (incrémente `mtc`, de 1 par défaut), la commande `\decrementmtc` (décrémente `mtc` de 1), la commande `\incrementmtc` (incrémente `mtc` de 1), le format du compteur `mtc` (`\themtc`), le compteur `minitocdepth`, initialisé à 2, pour la profondeur d'une minitoc (analogue au compteur standard `tocdepth`), les filets horizontaux à tracer avant et après les minitocs (`\mtc@rule`), puis nous copions cette définition dans des macros analogues pour les autres sortes de mini-tables. Nous établissons aussi la valeur par défaut (24pt) de `\mtcindent`, l'indentation pour les minitocs (des deux côtés). Les filets sont épais de 0.4pt. Ils sont définis via `\hrule` pour rester en mode vertical pour le `\kern` final.

```
\def\The@mtc{\arabic{mtc}}
\def\firstchapteris#1%
  {\PackageWarning{minitoc}%
     {*** \string\firstchapteris \space is an obsolete command ***}%
     \@firstchapteris@used@true}
\newcounter{mtc}
\setcounter{mtc}{0}
```

```
\newcommand{\adjustmtc}[1][1]{\addtocounter{mtc}{#1}}
\def\decrementmtc{\addtocounter{mtc}{-1}}
\def\incrementmtc{\addtocounter{mtc}{+1}}
\gdef\themtc{\arabic{mtc}}
\newcounter{minitocdepth}
\setcounter{minitocdepth}{2}
\def\mtc@rule{\kern-3\p@ \hrule \@width\columnwidth \kern2.6\p@}
\let\mlf@rule\mtc@rule
\let\mlt@rule\mtc@rule
\let\plf@rule\mtc@rule
\let\plt@rule\mtc@rule
\let\slf@rule\mtc@rule
\let\slt@rule\mtc@rule
\mtcindent=24\p@
```

5.32 Commandes de fontes pour les mini-tables

\mtcfont
\mtcSfont
\mtcSSfont
\mtcSSSfont
\mtcPfont
\mtcSPfont
\mlffont
\mltfont
\mtifont

Nous définissons ces commandes avec des descriptions NFSS [28] complètes. Ces défintions sont effectives si `\chapter` est définie. Les fontes pour les titres sont aussi définies ici. Voir aussi la macro `\mtcsetfont` (section 5.65.2 page 166) et la macro `\mtcsettitlefont` plus loin (section 5.65.3 page 168).

```
\def\mtcfont{\small\rmfamily\upshape\mdseries}
\def\mtcSfont{\small\rmfamily\upshape\bfseries}
\let\mtcSSfont\mtcfont
\let\mtcSSSfont\mtcfont
\let\mtcPfont\mtcfont
\let\mtcSPfont\mtcfont
\let\mlffont\mtcfont
\let\mltfont\mtcfont
\def\mtifont{\large\rmfamily\upshape\bfseries}
```

\coffeefont

Et `\coffeefont` est utilisée pour les « pauses café » dans le paquetage `minutes` [30].

```
\def\coffeefont{\small\rmfamily\slshape\mdseries}
```

5.33 Commandes internes pour positionner les titres des mini-tables

\df@mtitc
\df@mtilf
\df@mtilt
\do@mtitc
\do@mtilf
\do@mtilt

Les commandes `\miniXXX` et `\dominiXXX` acceptent un argument optionnel pour justifier à gauche, centrer, justifier à droite ou omettre le titre des mini-tables au niveau chapitre. Par défaut, ces titres sont justifiés à gauche. Le choix fait dans une commande `\dominiXXX` est global et mémorisé dans `\df@mtitc`, `\df@mtilf` ou `\df@mtilt` ; le choix fait dans une commande `\miniXXX` est local et rangé dans `\do@mtitc`, `\do@mtilf` ou `\do@mtilt`. Voir la macro `\minitoc@` plus loin (section 5.35.1 page suivante). Un titre vide nécessite une correction verticale (Frank MITTELBACH).

`\c@mti` `\l@mti` `\r@mti` `\e@mti` `\n@mti`

Titres centrés, justifiés à gauche, justifiés à droite ou vides :

```
\def\c@mti#1{\null\hfill #1\hfill\null}
\def\l@mti#1{\null #1\hfill\null}
\def\r@mti#1{\null\hfill #1\null}
\def\e@mti#1{\vspace{-\baselineskip}}
\def\n@mti#1{\vspace{-\baselineskip}}
```

`\l@mti` `\do@mtitc` `\df@mtitc` `\do@mtilf` `\df@mtilf` `\do@mtilt` `\df@mtilt`

Par défaut, les titres sont justifiés à gauche :

```
\let\do@mtitc\l@mti
\let\df@mtitc\l@mti
\let\do@mtilf\l@mti
\let\df@mtilf\l@mti
\let\do@mtilt\l@mti
\let\df@mtilt\l@mti
```

5.34 L'environnement mtc@verse

`mtc@verse` `\iftightmtc` `\ifktightmtc` `\\` `\list` `\itemsep` `\itemindent` `\listparindent` `\topsep` `\parsep` `\mtcindent`

Chaque minitoc est placée à l'intérieur d'un environnement `mtc@verse`. Cet environnement est analogue à l'environnement standard `verse` et donc il est défini à l'aide de deux commandes : `\mtc@verse` et `\endmtc@verse`. Comme c'est un environnement de liste, nous définissons d'abord (de manière locale) `\\`, puis nous appelons `\list{}` et établissons quelques dimensions telles que `\itemsep`, `\itemindent`, `\listparindent`, `\topsep`. `\parsep` est forcée à zéro si l'option `tight` est active (pour réduire l'espacement des lignes). Les deux marges sont établies à `\mtcindent`. `\endmtc@verse` termine la liste et décourage une coupure de page.

```
\def\mtc@verse{\let\\=\@centercr
  \list{}{\itemsep\z@
  \itemindent \z@
  \listparindent \itemindent
  \partopsep\z@
  \iftightmtc \parsep\z@ \fi
  \ifktightmtc \parskip\z@ \fi
  \topsep=1ex
  \leftmargin\mtcindent
  \rightmargin\leftmargin}\item[]}
\def\endmtc@verse{\nopagebreak[4]\endlist}
```

5.35 Les commandes \minitoc, \minilof et \minilot

Ces trois commandes sont très similaires, avec seulement des différences de détail.

5.35.1 La commande \minitoc

\minitoc \chapter

La commande `\minitoc` doit être utilisée après `\chapter` si vous voulez une minitoc (il n'y a pas de minitoc automatique).

\dominitoc \minitoc

Cette commande accepte un argument optionnel, dont la valeur par défaut éventuellement été établie plus tôt par une commande `\dominitoc`. La lettre « `d` » représente cette valeur par défaut. `\dominitoc` a elle-même un argument optionnel qui établit la valeur par défaut de l'argument optionnel de `\minitoc`. La valeur par défaut de l'argument optionnel de la commande `\dominitoc` est « `l` ». Ceci semble tortueux, mais c'est simple à utiliser : nous avons un comportement par défaut (`l`) qui peut être altéré globalement via l'argument optionnel de `\dominitoc`, ou localement via l'argument optionnel de `\minitoc`.

\minitoc \minitoc@ \@ifnextchar

Donc nous définissons `\minitoc` avec un argument optionnel et sa valeur par défaut (courante), puis appelons le vrai code dans la macro the `\minitoc@` (qui a un argument délimité) ; nous utilisons l'astuce `\@ifnextchar` pour détecter le crochet gauche de l'argument optionnel :

```
\def\minitoc{\@ifnextchar[{\minitoc@}{\minitoc@[d]}}
```

Le véritable code de `\minitoc` est dans `\minitoc@`, qui a un argument obligatoire (délimité par les crochets) spécifiant la position du titre.

\if@minitoc@used@

D'abord, nous établissons l'indicateur global `\@minitoc@used@true` pour noter que `\minitoc` a été appelée (ceci sera utilisé par une indication ou *hint* plus tard, section 5.76.2.2 page 199).

```
\def\minitoc@[#1]{%
\global\@minitoc@used@true
```

\@tocfile \if@longextensions

Le nom du fichier contenant la minitoc est construit à partir de `\jobname` et d'un suffixe `\@tocfile`, qui est `mtc` (suffixes longs) ou `M` (suffixes courts) suivi du numéro absolu de la minitoc.

```
\if@longextensions@
\def\@tocfile{mtc\The@mtc}%
\else
\def\@tocfile{M\The@mtc}%
\fi
```

\mtc@CkFile
\if@mtc@FE
\if@mtc@empty@minitoc@

Puis nous testons (via \mtc@CkFile) la vacuité de ce fichier. Un avertissement est émis si le fichier est vide et un indicateur est établi (une indication ou *hint* signalera qu'une minitoc vide a été demandée).

```
      \mtc@CkFile{\jobname.\@tocfile}
      \if@mtc@FE
      \PackageInfo{minitoc}%
         {\jobname.\@tocfile\space is empty}
      \@mtc@empty@minitoc@true
      \else
```

\thispageminitocstyle

Nous appelons \thispageminitocstyle pour établir le style de page (par défaut, ceci ne fait rien car, par défaut, il n'y a pas de saut de page avant une minitoc). Les marques ne sont pas traitées, car habituellement, il n'y a pas de saut de page avant une minitoc.

```
      \thispageminitocstyle
%%       \mtc@markboth{\uppercase{\mtctitle}}{\uppercase{\mtctitle}}%
```

\beforeminitoc
samepage
\do@mtitc
\e@mti
\n@mti
\c@mti
\l@mti
\r@mti
\df@mtic
\mtc@CkStr
\mtctitle
\if@mtc@FE
\mtcfont
\mtifont
\mtc@rule
tabular

Nous appelons \beforeminitoc, puis commençons un environnement samepage (pour essayer de décourager les coupures de page à l'intérieur d'une minitoc) et examinons la position du titre. Si le titre est vide, la mise en page est corrigée. Nous imprimons le titre avec sa fonte (\mtifont), puis le filet du haut de la minitoc (si les filets sont présents), en utilisant un environnement tabular (pour empêcher une coupure de page entre le titre et le filet du haut). La fonte est établie à \mtcfont.

```
      \beforeminitoc
      \relax\begin{samepage}%
      \if #1e\let\do@mtitc\e@mti
      \else\if #1n\let\do@mtitc\n@mti
      \else\if #1c\let\do@mtitc\c@mti
      \else\if #1l\let\do@mtitc\l@mti
      \else\if #1r\let\do@mtitc\r@mti
      \else\if #1d\let\do@mtitc\df@mtitc
      \fi\fi\fi\fi\fi\fi
      \mtc@CkStr{\mtctitle}\if@mtc@FE \let\do@mtitc\e@mti\relax\fi
      \raggedright
      \parskip=\z@%
      \reset@font\mtcfont%
      \parindent=\z@%
      \nopagebreak[4]%
      \kern-0.8\baselineskip\nopagebreak[4]%
      \par\noindent %%
      \ifx\mtc@rule\relax
      \begin{tabular}{@{}p{\columnwidth}@{}}
      \reset@font\mtifont\do@mtitc{\mtc@v\mtctitle}\\
      \end{tabular}%
    \else
      \begin{tabular}{@{}p{\columnwidth}@{}}
      \reset@font\mtifont\do@mtitc{\mtc@v\mtctitle}\\\hline
      \end{tabular}%
```

```
      \fi
```

\mtc@zrule \mtc@BBR \mtcindent mtc@verse

Nous interdisons une coupure de page après le titre et le filet supérieur, puis établissons quelques paramètres de mise en page et commençons un environnement `mtc@verse` :

```
        \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
        \leftmargin\mtcindent
        \rightmargin\mtcindent
        \itemindent=\z@\labelwidth=\z@%
        \labelsep=\z@\listparindent=\z@%
        \begin{mtc@verse}%
```

\c@tocdepth \c@minitocdepth \\ \mtc@BBR

Nous forçons la profondeur effective de la mini-table (`\c@tocdepth`) à la profondeur demandée (`\c@minitocdepth`), dons l'impression est faite à l'intérieur de l'environnement `mtc@verse`, où `tocdepth` a été forcée à `minitocdepth`, pour n'imprimer que les entrées dont le niveau est assez bas, puis nous inhibons une coupure de page. La ligne blanche est nécessaire pour éviter une indentation négative parasite.

```
        \c@tocdepth=\c@minitocdepth
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\mtc@pgno \@dottedtocline \@undottedtocline \mtc@hook@beforeinputfile \mtc@setform \mtcsetformat \mtc@strut

Nous testons la présence de points de conduite et de numéros de pages, puis imprimons la minitoc en insérant le fichier de minitoc. Mais avant de lire ce fichier de minitoc, nous devons appeler la macro d'attache (*hook*) (demandée par Donald Arseneau pour son paquetage `notoccite` [2]) `\mtc@hook@beforeinputfile` et la macro `\mtc@setform` qui ajuste certains paramètres de mise en page (définis par l'utilisateur via `\mtcsetformat`). Nous travaillons dans un groupe pour que certaines redéfinitions de macros restent locales.

```
\begingroup
  \makeatletter
  \@ifundefined{mtc@pgno}%
  {\let\@dottedtocline\@undottedtocline}{}
  \@fileswfalse\mtc@hook@beforeinputfile
  \mtc@setform%
  \@input{\jobname.\@tocfile}
  \vspace{-1ex} \vspace{-\baselineskip}
  \leavevmode\mtc@strut
  \global\@nobreakfalse\endgroup
```

mtc@verse \mtc@bottom@rule \\ samepage \afterminitoc

Nous fermons l'environnement `mtc@verse`, ajoutons le filet inférieur (tout en évitant une coupure de page), puis fermons l'environnement `samepage`, et enfin appelons `\afterminitoc`. La ligne blanche (`\\`) est essentielle.

```
        \end{mtc@verse}%
        \kern0.\baselineskip%
        \nopagebreak[4]\mtc@bottom@rule\null\leavevmode\\%
```

```
        \vskip-1.0\baselineskip\mtc@zrule
        \end{samepage}%
        \par\pagebreak[1]\vspace*{-1ex}\afterminitoc\fi}
```

\mtc@bottom@rule
\mtc@rule

Et nous définissons le filet inférieur pour une minitoc, avec un peu d'espace en dessous de la minitoc :

```
\def\mtc@bottom@rule{%
  \ifx\mtc@rule\relax\relax\else
     \vskip -2.5ex
       \rule[2.4\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}\fi}
```

5.35.2 La commande `\minilof`

\minilof

La commande `\minilof` est très similaire à la commande `\minitoc`.

\minilof
\chapter

La commande `\minilof` doit être utilisée après `\chapter` si vous voulez une minilof (il n'y a pas de minilof automatique).

\dominilof
\minilof

Cette commande accepte un argument optionnel, dont la valeur par défaut éventuellement été établie plus tôt par une commande `\dominilof`. La lettre « `d` » représente cette valeur par défaut. `\dominilof` a elle-même un argument optionnel qui établit la valeur par défaut de l'argument optionnel de `\minilof`. La valeur par défaut de l'argument optionnel de la commande `\dominilof` est « `l` ». Ceci semble tortueux, mais c'est simple à utiliser : nous avons un comportement par défaut (`l`) qui peut être altéré globalement via l'argument optionnel de `\dominilof`, ou localement via l'argument optionnel de `\minilof`.

\minilof
\minilof@
\@ifnextchar

Donc nous définissons `\minilof` avec un argument optionnel et sa valeur par défaut (courante), puis appelons le vrai code dans la macro the `\minilof@` (qui a un argument délimité) ; nous utilisons l'astuce `\@ifnextchar` pour détecter le crochet gauche de l'argument optionnel :

```
\def\minilof{\@ifnextchar[{\minilof@}{\minilof@[d]}}
```

Le véritable code de `\minilof` est dans `\minilof@`, qui a un argument obligatoire (délimité par les crochets) spécifiant la position du titre.

\if@minilof@used@

D'abord, nous établissons l'indicateur global `\@minilof@used@true` pour noter que `\minilof` a été appelée (ceci sera utilisé par une indication ou *hint* plus tard, section 5.76.2.2 page 199).

```
\def\minilof@[#1]{%
\global\@minilof@used@true
```

\@tocfile
\if@longextensions@

Le nom du fichier contenant la minilof est construit à partir de `\jobname` et d'un suffixe `\@tocfile`, qui est `mlf` (suffixes longs) ou `F` (suffixes courts) suivi du numéro absolu de la minilof.

```
\if@longextensions@%
\def\@tocfile{mlf\The@mtc}%
\else
\def\@tocfile{F\The@mtc}%
\fi
```

\mtc@CkFile
\if@mtc@FE
\if@mtc@empty@minilof@

Puis nous testons (via `\mtc@CkFile`) la vacuité de ce fichier. Un avertissement est émis si le fichier est vide et un indicateur est établi (une indication ou *hint* signalera qu'une minilof vide a été demandée).

```
        \mtc@CkFile{\jobname.\@tocfile}
        \if@mtc@FE
        \PackageInfo{minitoc}%
           {\jobname.\@tocfile\space is empty}
        \@mtc@empty@minilof@true
        \else
```

\thispageminilofstyle

Nous appelons `\thispageminilofstyle` pour établir le style de page (par défaut, ceci ne fait rien car, par défaut, il n'y a pas de saut de page avant une minilof). Les marques ne sont pas traitées, car habituellement, il n'y a pas de saut de page avant une minilof.

```
        \thispageminilofstyle
%%        \mtc@markboth{\uppercase{\mlftitle}}{\uppercase{\mlftitle}}%
```

\beforeminilof
samepage
\do@mtitc
\e@mti
\n@mti
\c@mti
\l@mti
\r@mti
\df@mtic
\mtc@CkStr
\mtctitle
\if@mtc@FE
\mlffont
\mtifont
\mtc@rule
tabular

Nous appelons `\beforeminilof`, puis commençons un environnement `samepage` (pour essayer de décourager les coupures de page à l'intérieur d'une minilof) et examinons la position du titre. Si le titre est vide, la mise en page est corrigée. Nous imprimons le titre avec sa fonte (`\mtifont`), puis le filet du haut de la minilof (si les filets sont présents), en utilisant un environnement `tabular` (pour empêcher une coupure de page entre le titre et le filet du haut). La fonte est établie à `\mlffont`.

```
        \beforeminilof
        \relax\begin{samepage}%
        \if #1e\let\do@mtilf\e@mti
        \else\if #1n\let\do@mtilf\n@mti
        \else\if #1c\let\do@mtilf\c@mti
        \else\if #1l\let\do@mtilf\l@mti
        \else\if #1r\let\do@mtilf\r@mti
        \else\if #1d\let\do@mtilf\df@mtilf
        \fi\fi\fi\fi\fi\fi
        \mtc@CkStr{\mlftitle}\if@mtc@FE \let\do@mtilf\e@mti\relax\fi
        \raggedright
        \parskip=\z@
        \reset@font\mlffont
```

```
        \parindent=\z@
        \nopagebreak[4]%
        \kern-0.8\baselineskip\nopagebreak[4]%
        \par\noindent
        \ifx\mlf@rule\relax
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\\
        \end{tabular}%
        \else
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\\\hline
        \end{tabular}%
        \fi
```

\mtc@zrule \mtc@BBR \mtcindent mtc@verse

Nous interdisons une coupure de page après le titre et le filet supérieur, puis établissons quelques paramètres de mise en page et commençons un environnement `mtc@verse` :

```
        \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
        \leftmargin\mtcindent
        \rightmargin\mtcindent
        \itemindent=\z@\labelwidth=\z@%
        \labelsep=\z@\listparindent=\z@%
        \begin{mtc@verse}%
```

\c@tocdepth \c@minilofdepth \\ \mtc@BBR

Nous forçons la profondeur effective de la mini-table (`\c@tocdepth`) à la profondeur demandée (`\c@minitocdepth`), dons l'impression est faite à l'intérieur de l'environnement `mtc@verse`, où `tocdepth` a été forcée à `minilofdepth`, pour n'imprimer que les entrées dont le niveau est assez bas, puis nous inhibons une coupure de page. La ligne blanche est nécessaire pour éviter une indentation négative parasite.

```
        \@ifundefined{c@lofdepth}{}{\c@tocdepth=\c@minilofdepth}
        \ifnum\c@tocdepth<1\relax\c@tocdepth=1\fi
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\mtc@pgno \@dottedtocline \@undottedtocline \mtc@hook@beforeinputfile \mlf@setform \mtcsetformat \mtc@strut

Nous testons la présence de points de conduite et de numéros de pages, puis imprimons la minilof en insérant le fichier de minilof. Mais avant de lire ce fichier de minilof, nous devons appeler la macro d'attache (*hook*) (demandée par Donald Arseneau pour son paquetage `notoccite` [2]) `\mtc@hook@beforeinputfile` et la macro `\mlf@setform` qui ajuste certains paramètres de mise en page (définis par l'utilisateur via `\mtcsetformat`). Nous travaillons dans un groupe pour que certaines redéfinitions de macros restent locales.

```
\begingroup
  \makeatletter
  \@ifundefined{mlf@pgno}%
  {\let\@dottedtocline\@undottedtocline}{}
  \@fileswfalse\mtc@hook@beforeinputfile
  \mlf@setform
```

```
\@input{\jobname.\@tocfile}
\vspace{-1ex} \vspace{-\baselineskip}
\leavevmode\mtc@strut
\global\@nobreakfalse\endgroup
```

mtc@verse \mtc@bottom@rule \\ samepage \afterminilof

Nous fermons l'environnement `mtc@verse`, ajoutons le filet inférieur (tout en évitant une coupure de page), puis fermons l'environnement `samepage`, et enfin appelons `\afterminilof`. La ligne blanche (`\\`) est essentielle.

```
        \end{mtc@verse}%
        \kern-1.\baselineskip%
        \nopagebreak[4]\mlf@rule\null\leavevmode\\%
        \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
        \par\pagebreak[1]\vspace*{-1ex}\afterminilof\fi}%
```

5.35.3 La commande `\minilot`

\minilot

La commande `\minilot` est absolument similaire à la commande `\minilof` :

\minilot \chapter

La commande `\minilot` doit être utilisée après `\chapter` si vous voulez une minilot (il n'y a pas de minilot automatique).

\dominilot \minilot

Cette commande accepte un argument optionnel, dont la valeur par défaut éventuellement été établie plus tôt par une commande `\dominilot`. La lettre « `d` » représente cette valeur par défaut. `\dominilot` a elle-même un argument optionnel qui établit la valeur par défaut de l'argument optionnel de `\minilot`. La valeur par défaut de l'argument optionnel de la commande `\dominilot` est « `l` ». Ceci semble tortueux, mais c'est simple à utiliser : nous avons un comportement par défaut (`l`) qui peut être altéré globalement via l'argument optionnel de `\dominilot`, ou localement via l'argument optionnel de `\minilot`.

\minilot \minilot@ \@ifnextchar

Donc nous définissons `\minilot` avec un argument optionnel et sa valeur par défaut (courante), puis appelons le vrai code dans la macro the `\minilot@` (qui a un argument délimité) ; nous utilisons l'astuce `\@ifnextchar` pour détecter le crochet gauche de l'argument optionnel :

```
\def\minilot{\@ifnextchar[{\minilot@}{\minilot@[d]}}
```

Le véritable code de `\minilot` est dans `\minilot@`, qui a un argument obligatoire (délimité par les crochets) spécifiant la position du titre.

\if@minilot@used@

D'abord, nous établissons l'indicateur global `\@minilot@used@true` pour noter que `\minilot` a été appelée (ceci sera utilisé par une indication ou *hint* plus tard, section 5.76.2.2 page 199).

```
\def\minilot@[#1]{%
\global\@minilot@used@true
```

\@tocfile
\if@longextensions@

Le nom du fichier contenant la minilot est construit à partir de `\jobname` et d'un suffixe `\@tocfile`, qui est `mlt` (suffixes longs) ou `T` (suffixes courts) suivi du numéro absolu de la minilot.

```
\if@longextensions@%
\def\@tocfile{mlt\The@mtc}%
\else
\def\@tocfile{T\The@mtc}%
\fi
```

\mtc@CkFile
\if@mtc@FE
\if@mtc@empty@minilot@

Puis nous testons (via `\mtc@CkFile`) la vacuité de ce fichier. Un avertissement est émis si le fichier est vide et un indicateur est établi (une indication ou *hint* signalera qu'une minilot vide a été demandée).

```
        \mtc@CkFile{\jobname.\@tocfile}
        \if@mtc@FE
        \PackageInfo{minitoc}%
           {\jobname.\@tocfile\space is empty}
        \@mtc@empty@minilot@true
        \else
```

\thispageminilotstyle

Nous appelons `\thispageminilotstyle` pour établir le style de page (par défaut, ceci ne fait rien car, par défaut, il n'y a pas de saut de page avant une minilot). Les marques ne sont pas traitées, car habituellement, il n'y a pas de saut de page avant une minilot.

```
        \thispageminilotstyle
%%        \mtc@markboth{\uppercase{\mlttitle}}{\uppercase{\mlttitle}}%
```

\beforeminilot
samepage
\do@mtitc
\e@mti
\n@mti
\c@mti
\l@mti
\r@mti
\df@mtic
\mtc@CkStr
\mtctitle
\if@mtc@FE
\mltfont
\mtifont
\mtc@rule
tabular

Nous appelons `\beforeminilot`, puis commençons un environnement `samepage` (pour essayer de décourager les coupures de page à l'intérieur d'une minilot) et exminons la position du titre. Si le titre est vide, la mise en page est corrigée. Nous imprimons le titre avec sa fonte (`\mtifont`), puis le filet du haut de la minilot (si les filets sont présents), en utilisant un environnement `tabular` (pour empêcher une coupure de page entre le titre et le filet du haut). La fonte est établie à `\mltfont`.

```
        \beforeminilot
        \relax\begin{samepage}%
        \if #1e\let\do@mtilt\e@mti
        \else\if #1n\let\do@mtilt\n@mti
        \else\if #1c\let\do@mtilt\c@mti
        \else\if #1l\let\do@mtilt\l@mti
        \else\if #1r\let\do@mtilt\r@mti
        \else\if #1d\let\do@mtilt\df@mtilt
        \fi\fi\fi\fi\fi\fi
```

```
        \mtc@CkStr{\mlttitle}\if@mtc@FE \let\do@mtilt\e@mti\relax\fi
        \raggedright
        \parskip=\z@%
        \reset@font\mltfont%
        \parindent=\z@%
        \nopagebreak[4]%
        \kern-0.8\baselineskip\nopagebreak[4]%
        \par\noindent
        \ifx\mlt@rule\relax
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\mtifont\do@mtilt{\mtc@v\mlttitle}\\
        \end{tabular}%
      \else
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\mtifont\do@mtilt{\mtc@v\mlttitle}\\\hline
        \end{tabular}%
      \fi
```

\mtc@zrule
\mtc@BBR
\mtcindent
mtc@verse

Nous interdisons une coupure de page après le titre et le filet supérieur, puis établissons quelques paramètres de mise en page et commençons un environnement mtc@verse :

```
        \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
        \leftmargin\mtcindent
        \rightmargin\mtcindent
        \itemindent=\z@\labelwidth=\z@%
        \labelsep=\z@\listparindent=\z@%
        \begin{mtc@verse}%
```

\c@tocdepth
\c@minilotdepth
\\
\mtc@BBR

Nous forçons la profondeur effective de la mini-table (\c@tocdepth) à la profondeur demandée (\c@minitocdepth), dons l'impression est faite à l'intérieur de l'environnement mtc@verse, où tocdepth a été forcée à minilotdepth, pour n'imprimer que les entrées dont le niveau est assez bas, puis nous inhibons une coupure de page. La ligne blanche est nécessaire pour éviter une indentation négative parasite.

```
        \@ifundefined{c@lotdepth}{}{\c@tocdepth=\c@minilotdepth}
        \ifnum\c@tocdepth<1\relax\c@tocdepth=1\fi
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\mtc@pgno
\@dottedtocline
\@undottedtocline
\mtc@hook@beforeinputfile
\mlt@setform
\mtcsetformat
\mtc@strut

Nous testons la présence de points de conduite et de numéros de pages, puis imprimons la minilot en insérant le fichier de minilot. Mais avant de lire ce fichier de minilot, nous devons appeler la macro d'attache (*hook*) (demandée par Donald Arseneau pour son paquetage notoccite [2]) \mtc@hook@beforeinputfile et la macro \mlt@setform qui ajuste certains paramètres de mise en page (définis par l'utilisateur via \mtcsetformat). Nous travaillons dans un groupe pour que certaines redéfinitions de macros restent locales.

```
\begingroup
  \makeatletter
```

```
\@ifundefined{mlt@pgno}%
{\let\@dottedtocline\@undottedtocline}{}
\@fileswfalse\mtc@hook@beforeinputfile
\mlt@setform
\@input{\jobname.\@tocfile}
\vspace{-1ex} \vspace{-\baselineskip}
\leavevmode\mtc@strut
\global\@nobreakfalse\endgroup
```

mtc@verse \mtc@bottom@rule \\ samepage \afterminilot

Nous fermons l'environnement `mtc@verse`, ajoutons le filet inférieur (tout en évitant une coupure de page), puis fermons l'environnement `samepage`, et enfin appelons `\afterminilot`. La ligne blanche (`\\`) est essentielle.

```
        \end{mtc@verse}%
        \kern-1.\baselineskip%
        \nopagebreak[4]\mlt@rule\null\leavevmode\\%
        \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
        \par\pagebreak[1]\vspace*{-1ex}\afterminilot\fi}%
```

5.36 Modifier la commande \chapter, suite

\l@xchapter \@dottedtocline \l@chapter \xchapter

Tout d'abord, nous définissons `\l@xchapter` qui est semblable à `\l@chapter`, mais avec une profondeur très grande, pour inhiber son impression (sauf si vous trichez) :

```
\def\l@xchapter{\@dottedtocline{\@M}{1em}{2.3em}}
\def\xchapter{xchapter}
```

\@chapter \sv@chapter \addcontentsline \ignorespaces

Puis nous modifions `\@chapter` (la branche non étoilée de `\chapter`) pour ajouter des entrées de pseudo-chapitres dans la liste des figures et la liste des tableaux (ces entrées seront utilisées par les commandes `\dominiXXX` pour découper en tranches la liste des figures et la liste des tableaux).

```
\let\sv@chapter\@chapter
\def\@chapter[#1]#2{\sv@chapter[{#1}]{#2}\relax%
   \addcontentsline{lof}{xchapter}{#1}%
   \addcontentsline{lot}{xchapter}{#1}%
   \ignorespaces}
```

\mtc@schapter \@schapter \addtocontents \chapterbegin \chapterend

Nous modifions aussi `\@schapter` (la branche étoilée de `\chapter`) pour ajouter des marques dans la table des matières afin de délimiter les chapitres ; ces marques seront utilisées par les commandes `\dominiXXX` pour prélever des tranches dans la liste des figures et la liste des tableaux ; comme elles sont définies comme étant `\relax`, elles ne devraient pas perturber d'autres paquetages.

```
\let\mtc@schapter\@schapter
```

```
\def\@schapter{\addtocontents{toc}{\protect\chapterend}\mtc@schapter}
\def\@schapter{\addtocontents{@@@}{\protect\chapterbegin}\mtc@schapter}
\let\chapterbegin\relax
\let\chapterend\relax
```

5.37 Les commandes `\addstarred...`

\addstarredsection
\addstarredchapter
\addstarredpart
\chapter
\section
\part
\addst@rred

Si la commande `\chapter` n'est pas définie, alors nous devons définir la commande `\addstarredsection` (mais seulement si `\section` est définie). Si la commande `\chapter` est définie, nous définissons la commande `\addstarredchapter`. Si la commande `\part` est définie, nous définissons la commande `\addstarredpart`. Nous utilisons la commande utilitaire `\addst@rred` définie dans la section 5.31 page 90.

```
\@ifundefined{chapter}%
   {\@ifundefined{section}%
      {}{\def\addstarredsection#1{\addst@rred{section}{#1}}}}%
   {\def\addstarredchapter#1{\addst@rred{chapter}{#1}}}
\@ifundefined{part}%
   {}{\def\addstarredpart#1{\addst@rred{part}{#1}}}
```

5.38 Entrées de table des matières sans points de conduite

\@Undottedtocline
\coffeefont

Nous définissons deux macros internes pour mettre en forme des entrées de table des matières sans points de conduite. La macro `\@Undottedtocline` n'imprime pas de numéro de page, mais `\@Undottedtoclinep` l'imprime.

```
\def\@Undottedtocline#1#2#3#4#5{%
  \ifnum #1>\c@tocdepth \else
    \vskip \z@ \@plus.2\p@
    {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
     \parindent #2\relax\@afterindenttrue
     \interlinepenalty\@M
     \leavevmode
     \@tempdima #3\relax
     \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
     {\coffeefont #4}\nobreak
     \nobreak\null
     \par}%
  \fi}
```

\@Undottedtoclinep
\coffeefont

La même chose, mais avec le numéro de page :

```
\def\@Undottedtoclinep#1#2#3#4#5{%
  \ifnum #1>\c@tocdepth \else
    \vskip \z@ \@plus.2\p@
    {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
     \parindent #2\relax\@afterindenttrue
     \interlinepenalty\@M
     \leavevmode
     \@tempdima #3\relax
     \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
     {#4}\nobreak
        \hfill
     \nobreak\null
     \hb@xt@\@pnumwidth{\hfil\normalfont \normalcolor #5}%
     \par}%
  \fi}
```

5.39 Mini-tables avec ou sans points de conduite

\minitoc@
\minilof@
\minilot@
\ifundottedmtc
\@dottedtocline
\@Undottedtoclinep
\sv@minitoc@
\sv@minilof@
\sv@minilot@

Ce code crée l'indicateur, puis modifie chaque commande de mini-table (en fait, sa partie interne). Nous modifions les commandes \minitoc@, \minilof@, etc., pour tester l'indicateur \ifundottedmtc et, s'il est vrai, remplacer localement \@dottedtocline par sa version sans points de conduite \@Undottedtoclinep. Bien sûr, nous devons aussi tester la disponibilité des commandes \chapter, \part et \section, pour éviter de définir de nombreuses commandes non nécessaires.

```
\newif\ifundottedmtc\undottedmtcfalse
\@ifundefined{chapter}{}{%
   \let\sv@minitoc@\minitoc@
   \def\minitoc@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
      \sv@minitoc@[#1]}}%
   \let\sv@minilof@\minilof@
   \def\minilof@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
      \sv@minilof@[#1]}}%
   \let\sv@minilot@\minilot@
   \def\minilot@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
      \sv@minilot@[#1]}}}
```

\sv@parttoc@
\sv@partlof@
\sv@partlot@
\ifundottedmtc
\parttoc@
\partlof@
\partlot@

Pour le niveau partie :

```
\@ifundefined{part}{}{%
   \let\sv@parttoc@\parttoc@
   \def\parttoc@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
      \sv@parttoc@[#1]}}%
   \let\sv@partlof@\partlof@
   \def\partlof@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
      \sv@partlof@[#1]}}%
```

```
\let\sv@partlot@\partlot@
\def\partlot@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
   \sv@partlot@[#1]}}}
```

\sv@secttoc@ \sv@sectlof@ \sv@sectlot@ \ifundottedmtc \secttoc@ \sectlof@ \sectlot@

Pour le niveau section :

```
\@ifundefined{chapter}{%
  \@ifundefined{section}{}{%
    \let\sv@secttoc@\secttoc@
    \def\secttoc@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
       \sv@secttoc@[#1]}}%
    \let\sv@sectlof@\sectlof@
    \def\sectlof@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
       \sv@sectlof@[#1]}}%
    \let\sv@sectlot@\sectlot@
    \def\sectlot@[#1]{{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
       \sv@sectlot@[#1]}}}}{}
```

5.40 La commande `\dominitoc` et ses analogues

\dominitoc \dominilof \dominilot \contentsline \chapbegin \starchapter

Les trois commandes `\dominitoc`, `\dominilof` et `\dominilot` sont, bien sûr, très similaires. Elles prennent le fichier `\jobname.toc` (respectivement les fichiers `\jobname.lof` et `\jobname.lot`) produit par l'exécution précédente de LaTeX et l'éclatent en tranches (une tranche par chapitre ou chapitre étoilé) vers les fichiers `\jobname.mtc`⟨*N*⟩ (respectivement, les fichiers `\jobname.mlf`⟨*N*⟩ et `\jobname.mlt`⟨*N*⟩ files), en utilisant des lignes spécifiques dans le fichier `\jobname.toc` (respectivement, `\jobname.lof` et `\jobname.lot`). Ces lignes sont essentiellement des commandes pour des entrées au niveau chapitre (telles que `\contentsline{chapter}...`, `\contentsline{xchapter}...`, `\contentsline{starchapter}...`, `\chapbegin` délimitant les chapitres dans la table des matières (ou dans la liste des figures ou dans la liste des tableaux). Des lignes analogues au niveau partie délimitent les parties, donc aussi les chapitres.

\dominitoc \dominitoc@ \@@dominitoc \if@dominitoc@used@

Puisque `\dominitoc` a un argument optionnel, dont la valeur par défaut est « `l` » (gauche), elle appelle `\dominitoc@` avec un argument délimité par des crochets.

Les macros sont `\dominitoc` (interface utilisateur), qui appelle `\dominitoc@[l]` (ou avec l'argument optionnel de `\dominitoc`). Puis `\dominitoc@[l]` traite son argument et appelle `\@@dominitoc`. `\@@dominitoc` appelle `\@dominitoc` (en passant `\jobname` comme argument) puis ferme le fichier minitoc écrit. `\@dominitoc` remet à zéro le compteur de mini-tables, appelle `\MTC@next#1.toc` (où `#1` est la valeur de `\jobname`, puis remet encore à zéro le compteur de mini-tables. Chaque appel à `\dominitoc@` (c'est-à-dire, à `dominitoc`) établit l'indicateur `\@dominitoc@used@true`. Ceci sera utilisé plus tard pour une indication (*hint*) (qui détecte si vous avez appelé correctement `\minitoc` *après* `\dominitoc` et que les deux ont été appelées). Voir la section 5.76.2.2 page 199. Le code est similaire pour `\dominilof` et `\dominilot`.

\dominitoc
\@dominitoc
\MTC@next

La commande \dominitoc extrait des informations du fichier .toc et crée les fichiers minitocs, avec le suffixe adéquat.

```
\def\@dominitoc#1{{%
  \makeatletter
  \setcounter{mtc}{0}
  \MTC@next#1.toc\relax\\}\setcounter{mtc}{0}}
\def\dominitoc{\@ifnextchar[{\dominitoc@}{\dominitoc@[l]}}
```

\dominilof
\@dominilof
\MLF@next

La commande \dominilof extrait des informations du fichier .lof et crée les fichiers minilofs, avec le suffixe adéquat.

```
\def\@dominilof#1{{%
  \makeatletter
  \setcounter{mtc}{0}
  \MLF@next#1.lof\relax\\}\setcounter{mtc}{0}}
\def\dominilof{\@ifnextchar[{\dominilof@}{\dominilof@[l]}}
```

\dominilot
\@dominilot
\MLT@next

La commande \dominilot extrait des informations du fichier .lot et crée les fichiers minilots, avec le suffixe adéquat.

```
%    \begin{macrocode}
\def\@dominilot#1{{%
  \makeatletter
  \setcounter{mtc}{0}
  \MLT@next#1.lot\relax\\}\setcounter{mtc}{0}}
\def\dominilot{\@ifnextchar[{\dominilot@}{\dominilot@[l]}}
```

\if@dominitoc@used@
\df@mtitc
\e@mti
\n@mti
\c@mti
\l@mti
\r@mti
\@@dominitoc

Un peu de code pour noter que la commande a été utilisée et gérer la position du titre de la minitoc :

```
\def\dominitoc@[#1]{%
\global\@dominitoc@used@true
\if #1e\let\df@mtitc\e@mti%
\else\if #1n\let\df@mtitc\n@mti%
\else\if #1c\let\df@mtitc\c@mti%
\else\if #1l\let\df@mtitc\l@mti%
\else\if #1r\let\df@mtitc\r@mti%
\fi\fi\fi\fi\fi%
\@@dominitoc}
```

\if@dominilof@used@ \df@mtilf \e@mti \n@mti \c@mti \l@mti \r@mti \@@dominilof

Un peu de code pour noter que la commande a été utilisée et gérer la position du titre de la minilof :

```
\def\dominilof@[#1]{%
\global\@dominilof@used@true
\if #1e\let\df@mtilf\e@mti%
\else\if #1n\let\df@mtilf\n@mti%
\else\if #1c\let\df@mtilf\c@mti%
\else\if #1l\let\df@mtilf\l@mti%
\else\if #1r\let\df@mtilf\r@mti%
\fi\fi\fi\fi\fi%
\@@dominilof}
```

\if@dominilot@used@ \df@mtilt \e@mti \n@mti \c@mti \l@mti \r@mti \@@dominilot

Un peu de code pour noter que la commande a été utilisée et gérer la position du titre de la minilot :

```
\def\dominilot@[#1]{%
\global\@dominilot@used@true
\if #1e\let\df@mtilt\e@mti%
\else\if #1n\let\df@mtilt\n@mti%
\else\if #1c\let\df@mtilt\c@mti%
\else\if #1l\let\df@mtilt\l@mti%
\else\if #1r\let\df@mtilt\r@mti%
\fi\fi\fi\fi\fi%
\@@dominilot}
```

\@@dominitoc \@@dominilof \@@dominilot \tf@mtc

Ces macros invoquent les macros `\@domini...` pour créer le fichier de mini-table, puis ferment le descripteur de fichier.

```
\def\@@dominitoc{\@dominitoc{\jobname}\immediate\closeout\tf@mtc}
\def\@@dominilof{\@dominilof{\jobname}\immediate\closeout\tf@mtc}
\def\@@dominilot{\@dominilot{\jobname}\immediate\closeout\tf@mtc}
```

5.40.1 Analyse et éclatement du fichier table des matières

Ceci est fait grâce à une boucle gérée par les macros suivantes [6] :

\MTC@next \MTC@list \MTC@loop

Traite l'entrée suivante dans la liste et la retire de la tête de la liste :

```
\def\MTC@next#1\relax#2\\{%
  \edef\MTC@list{#2}%
```

[6] Ce code est dérivé du paquetage `xr` [12], de David Carlisle, avec sa permission. Les chaînes « HO », « DV » et « BJ » dans les commentaires indiquent des modifications faites par Heiko Oberdiek, Didier Verna et Bernd Jaehne pour le support de `hyperref`, essentiellement en ajoutant un argument à certaines macros, pour utiliser l'argument hyperlien dans les lignes de contenu.

```
  \MTC@loop{#1}%
}
```

\MTC@toc \MTC@list \MTC@explist

Teste si la liste est vide :

```
\def\MTC@toc{%
  \ifx\MTC@list\@empty\else\expandafter\MTC@explist\fi
}
```

\MTC@contentsline \arabic \chapter \themtc \tf@mtc

La macro `\MTC@contentsline` analyse les lignes lues depuis le fichier table des matières et détecte les mots-clés intéressants. Si `\chapter` est trouvée, le compteur `mtc` (qui simule le compteur de chapitre, mais est absolu) est incrémenté et un nouveau fichier minitoc est créé.

```
\def\MTC@contentsline#1#2#3#4{% %%HO/BJ: 4 instead of 3 parameters
  \gdef\themtc{\arabic{mtc}}% %%HO: space removed
  \expandafter\ifx\csname #1\endcsname\chapter
    \stepcounter{mtc}%
```

\if@longextensions@ \themtc \mtcname \tf@mtc \closeout \openout

Nous regardons si des suffixes longs ou courts ont utilisés, pour construire le nom du fichier mini-table, puis l'ouvrir (après avoir fermé le descripteur de fichier :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.mtc\themtc\@gobble}%
      \def\mtcname{\jobname.mtc\themtc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.M\themtc\@gobble}%
      \def\mtcname{\jobname.M\themtc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\mtcname
  \fi
```

\mtc@toks

Maintenant, nous filtrons les lignes de contenu utiles, le registre *token* étant utilisé comme mémoire *verbatim*.

```
  \mtc@toks{\noexpand\leavevmode #2}%
```

\MTC@WriteContentsline \section \subsection \subsubsection \paragraph \subparagraph

Chaque ligne de contenu intéressante est copiée, avec une commande de fonte ajoutée devant elle. Nous commençons par les commandes de sectionnement standard, en dessous de `\chapter` :

```
  \expandafter\ifx\csname #1\endcsname\section
    \MTC@WriteContentsline{#1}{mtcS}{#3}{#4}%
```

```
\fi
\expandafter\ifx\csname #1\endcsname\subsection
  \MTC@WriteContentsline{#1}{mtcSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\subsubsection
  \MTC@WriteContentsline{#1}{mtcSSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\paragraph
  \MTC@WriteContentsline{#1}{mtcP}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\subparagraph
  \MTC@WriteContentsline{#1}{mtcSP}{#3}{#4}%
\fi
```

\coffee
\MTC@WriteCoffeeline

Une ligne de contenu de pause-café est écrite pour `\coffee` :

```
\expandafter\ifx\csname #1\endcsname\coffee
  \MTC@WriteCoffeeline{#1}{#3}%
\fi
```

\starchapter
\stepcounter
\if@longextensions@
\mtcname
\themtc
\tf@mtc
\closeout
\openout

Si c'est `\starchapter` (pour un chapitre étoilé), nous incrémentons le compteur `mtc`, construisons un nouveau nom de fichier minitoc, fermons le descripteur de fichier puis le ré-ouvrons avec ce nouveau fichier.

```
\expandafter\ifx\csname #1\endcsname\starchapter
  \stepcounter{mtc}%
  \if@longextensions@
    \PackageInfo{minitoc}%
       {Writing\space\jobname.mtc\themtc\@gobble}%
    \def\mtcname{\jobname.mtc\themtc}%
  \else
    \PackageInfo{minitoc}%
       {Writing\space\jobname.M\themtc\@gobble}%
    \def\mtcname{\jobname.M\themtc}%
  \fi
  \immediate\closeout\tf@mtc
  \immediate\openout\tf@mtc=\mtcname
\fi
```

\starsection
\MTC@WriteContentsline
\starsubsection
\starsubsubsection
\starparagraph
\starsubparagraph

Pour les commandes de sectionnement étoilées plus basses que `\chapter`, une ligne de contenu est écrite dans le fichier minitoc, en lui ajoutant une commande de fonte :

```
\expandafter\ifx\csname #1\endcsname\starsection
  \MTC@WriteContentsline{#1}{mtcS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\starsubsection
  \MTC@WriteContentsline{#1}{mtcSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\starsubsubsection
  \MTC@WriteContentsline{#1}{mtcSSS}{#3}{#4}%
```

```
  \fi
  \expandafter\ifx\csname #1\endcsname\starparagraph
    \MTC@WriteContentsline{#1}{mtcP}{#3}{#4}%
  \fi
  \expandafter\ifx\csname #1\endcsname\starsubparagraph
    \MTC@WriteContentsline{#1}{mtcSP}{#3}{#4}%
  \fi
}
```

\MTC@explist \MTC@next \MTC@list

La boucle de lecture du fichier table des matières ; elle expanse la liste des entrées et appelle `\MTC@next` pour traiter la première :

```
\def\MTC@explist{\expandafter\MTC@next\MTC@list\\}
```

\MTC@loop \openin \@inputcheck \MTC@toc \MTC@read

Si une entrée est trouvée, on boucle ligne par ligne en cherchant des entrées intéressantes. Sinon, on traite l'entrée suivante dans la liste.

```
\def\MTC@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
       {No file #1.\MessageBreak MINITOCS NOT PREPARED}%
    \expandafter\MTC@toc
  \else
    \PackageInfo{minitoc}{PREPARING MINITOCS FROM #1}%
    \expandafter\MTC@read
  \fi
}
```

\MTC@read \read \@inputcheck \MTC@line

Lire l'entrée suivante dans le fichier `.toc`.

```
\def\MTC@read{%
  \read\@inputcheck to\MTC@line
```

\MTC@test \MTC@line \MTC@

La série de points `.....` garantit que `\MTC@test` a suffisamment d'arguments :

```
  \expandafter\MTC@test\MTC@line.....\MTC@% %%HO: . added
}%
```

\MTC@test \contentsline \MTC@contentsline \mtc@string \@input \MTC@list \chapterend \closeout \tf@mtc \openout \chapterbegin \addtocounter \MTC@toc \MTC@read

La macro `\MTC@test` trouve les commandes « intéressantes » dans le fichier table des matières, principalement pour délimiter les chapitres :

```
%%HO/BJ: now patch \MTC@test,
%%HO/BJ: call \MTC@contentsline with 4 instead of 3 parameters
```

Nous regardons le premier *token* de la ligne. Si c'est une entrée intéressante, nous la traitons. Si c'est `\@input`, le fichier est ajouté à la liste. Sinon, elle est ignorée. Nous recommençons la boucle si la fin du fichier n'est pas atteinte. Enfin, nous traitons le fichier suivant dans la liste.

```
\long\def\MTC@test#1#2#3#4#5#6\MTC@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \MTC@contentsline{#2}{#3}{#4}{#5}%
    %%HO/BJ: 4. parameter added by Tony Roberts
    \let\mtc@string\relax
  \else\ifx#1\@input
    \edef\MTC@list{\MTC@list#2\relax}%
  \else\ifx#1\chapterend
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\jobname.mtc
  \else\ifx#1\chapterbegin
    \addtocounter{mtc}{-1}%
  \fi\fi\fi\fi
  \ifeof\@inputcheck
    \expandafter\MTC@toc
  \else
    \expandafter\MTC@read
  \fi
}%
```

5.41 Mini-listes des figures

Le code est similaire à celui pour les mini-tables des matières, mais avec moins de commandes à reconnaître.

5.41.1 Analyse et éclatement du fichier liste des figures

`\MLF@next` Ceci est fait grâce à une boucle gérée par les macros suivantes :

`\MLF@list`

`\MLF@loop` Traite l'entrée suivante dans la liste et la retire de la tête de la liste :

```
\def\MLF@next#1\relax#2\\{%
  \edef\MLF@list{#2}%
  \MLF@loop{#1}}
```

`\MLF@lof` Teste si la liste est vide :

`\MLF@list`

`\MLF@explist`

```
\def\MLF@lof{%
  \ifx\MLF@list\@empty\else\expandafter\MLF@explist\fi}
```

\MLF@contentsline
\arabic
\xchapter

La macro \MLF@contentsline analyse les lignes du fichier liste des figures et détecte les mots-clés intéressants. Si \xchapter est trouvée, mtc est incrémenté et un nouveau fichier minilof est créé.

```
\def\MLF@contentsline#1#2#3#4{% %%HO: added #4
  \gdef\themtc{\arabic{mtc}}% %%HO: space removed
  \expandafter\ifx\csname #1\endcsname\xchapter
    \stepcounter{mtc}%
```

\if@longextensions@
\themtc
\mlfname
\closeout
\tf@mtc
\openout

Le nom du fichier minilof est construit avec \jobname et un suffixe long ou court :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.mlf\themtc\@gobble}
      \def\mlfname{\jobname.mlf\themtc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.F\themtc\@gobble}
      \def\mlfname{\jobname.F\themtc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\mlfname
  \fi
```

\figure
\mtc@toks
\MTC@WriteContentsline

Si nous trouvons une entrée \figure, nous la copions dans le fichier minilof :

```
  \expandafter\ifx\csname #1\endcsname\figure
    \mtc@toks{\noexpand\leavevmode#2}%
    \MTC@WriteContentsline{#1}{mlf}{#3}{#4}%
  \fi
}
```

\MLF@explist
\MLF@next
\MLF@list

La boucle de lecture du fichier liste des figures ; elle expanse la liste des entrées et appelle \MLF@next pour traiter la première :

```
\def\MLF@explist{\expandafter\MLF@next\MLF@list\\}
```

\MLF@loop
\openin
\@inputcheck
\MLF@lof
\MLF@read

Et maintenant, nous examinons le fichier .lof :

```
\def\MLF@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
       {No file #1.\MessageBreak MINILOFS NOT PREPARED}%
    \expandafter\MLF@lof
  \else
    \PackageInfo{minitoc}%
       {PREPARING MINILOFS FROM #1}%
    \expandafter\MLF@read\fi}
```

\MLF@read \read \@inputcheck

Lire l'entrée suivante dans le fichier `.lof` :

```
\def\MLF@read{%
  \read\@inputcheck to\MLF@line
```

\MLF@line \MLF@test \MLF@

La série de points `.....` garantit que `\MLF@test` a suffisamment d'arguments :

```
  \expandafter\MLF@test\MLF@line.....\MLF@% %%HO: . added
  }%
```

\MLF@test

La macro `\MLF@test` trouve les commandes « intéressantes » dans le fichier liste des figures, principalement pour délimiter les chapitres.

\contentsline \mtc@string \MLF@contentsline \@input \MLF@list \chapterend \closeout \tf@mtc \openout \chapterbegin \addtocounter \MLF@lof \MLF@read

Nous regardons le premier *token* de la ligne. Si c'est une entrée intéressante, nous la traitons. Si c'est `\@input`, le fichier est ajouté à la liste. Sinon, elle est ignorée. Nous recommençons la boucle si la fin du fichier n'est pas atteinte. Enfin, nous traitons le fichier suivant dans la liste.

```
\long\def\MLF@test#1#2#3#4#5#6\MLF@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \MLF@contentsline{#2}{#3}{#4}{#5}% %%HO: #4 added
    \let\mtc@string\relax
  \else\ifx#1\@input
     \edef\MLF@list{\MLF@list#2\relax}%
  \else\ifx#1\chapterend
     \immediate\closeout\tf@mtc
     \immediate\openout\tf@mtc=\jobname.mtc
  \else\ifx#1\chapterbegin
    \addtocounter{mtc}{-1}%
  \fi\fi\fi\fi
  \ifeof\@inputcheck\expandafter\MLF@lof
  \else\expandafter\MLF@read\fi}%
```

5.42 Mini-listes des tableaux

Le code est similaire à celui pour les mini-tables des matières, mais avec moins de commandes à reconnaître.

5.42.1 Analyse et éclatement du fichier liste des tableaux

\MLT@next \MLT@list \MLT@loop

Ceci est fait grâce à une boucle gérée par les macros suivantes :

Traite l'entrée suivante dans la liste et la retire de la tête de la liste :

```
\def\MLT@next#1\relax#2\\{%
  \edef\MLT@list{#2}%
  \MLT@loop{#1}}
```

`\MLT@lot` `\MLT@list` `\MLT@explist`

Teste si la liste est vide :

```
\def\MLT@lot{%
  \ifx\MLT@list\@empty\else\expandafter\MLT@explist\fi}
```

`\MLT@contentsline` `\arabic` `\xchapter`

La macro `\MLT@contentsline` analyse les lignes du fichier liste des tableaux et détecte les mots-clés intéressants. Si `\xchapter` est trouvée, `mtc` est incrémenté et un nouveau fichier minilot est créé.

```
\def\MLT@contentsline#1#2#3#4{% %%HO: added #4
  \gdef\themtc{\arabic{mtc}}% %%HO: space removed
  \expandafter\ifx\csname #1\endcsname\xchapter
    \stepcounter{mtc}%
```

`\if@longextensions@` `\themtc` `\mltname` `\closeout` `\tf@mtc` `\openout`

Le nom du fichier minilot est construit avec `\jobname` et un suffixe long ou court :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.mlt\themtc\@gobble}%
      \def\mltname{\jobname.mlt\themtc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.T\themtc\@gobble}%
      \def\mltname{\jobname.T\themtc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\mltname
  \fi
```

`\table` `\mtc@toks` `\MTC@WriteContentsline`

Si nous trouvons une entrée `\table`, nous la copions dans le fichier minilot :

```
  \expandafter\ifx\csname #1\endcsname\table
    \mtc@toks{\noexpand\leavevmode#2}%
    \MTC@WriteContentsline{#1}{mlt}{#3}{#4}%
  \fi
}
```

`\MLT@explist` `\MLT@next` `\MLT@list`

La boucle de lecture du fichier liste des tableaux ; elle expanse la liste des entrées et appelle `\MLT@next` pour traiter la première :

```
\def\MLT@explist{\expandafter\MLT@next\MLT@list\\}
```

\MLT@loop \openin \@inputcheck \MLT@lot \MLT@read

Et maintenant, nous examinons le fichier `.lot` :

```
\def\MLT@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
       {No file #1.\MessageBreak MINILOTS NOT PREPARED}%
    \expandafter\MLT@lot
  \else
    \PackageInfo{minitoc}%
       {PREPARING MINILOTS FROM #1}%
    \expandafter\MLT@read\fi}
```

\MLT@read \read \@inputcheck

Lire l'entrée suivante dans le fichier `.lot` :

```
\def\MLT@read{%
  \read\@inputcheck to\MLT@line
```

\MLT@line \MLT@test \MLT@

La série de points `.....` garantit que `\MLT@test` a suffisamment d'arguments :

```
  \expandafter\MLT@test\MLT@line.....\MLT@% %%HO: . added
  }%
```

\MLT@test

La macro `\MLT@test` trouve les commandes « intéressantes » dans le fichier liste des tableaux, principalement pour délimiter les chapitres.

\contentsline \mtc@string \MLT@contentsline \@input \MLT@list \chapterend \closeout \tf@mtc \openout \chapterbegin \addtocounter \MLT@lot \MLT@read

Nous regardons le premier *token* de la ligne. Si c'est une entrée intéressante, nous la traitons. Si c'est `\@input`, le fichier est ajouté à la liste. Sinon, elle est ignorée. Nous recommençons la boucle si la fin du fichier n'est pas atteinte. Enfin, nous traitons le fichier suivant dans la liste.

```
\long\def\MLT@test#1#2#3#4#5#6\MLT@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \MLT@contentsline{#2}{#3}{#4}{#5}% %%HO: #4 added
    \let\mtc@string\relax
  \else\ifx#1\@input
     \edef\MLT@list{\MLT@list#2\relax}%
  \else\ifx#1\chapterend
     \immediate\closeout\tf@mtc
     \immediate\openout\tf@mtc=\jobname.mtc
  \else\ifx#1\chapterbegin
    \addtocounter{mtc}{-1}%
  \fi\fi\fi\fi
  \ifeof\@inputcheck\expandafter\MLT@lot
  \else\expandafter\MLT@read\fi}%
```

Notez que nous terminons par une accolade fermante pour achever les macros du niveau chapitre (fin de la branche *sinon* d'une alternative `\@ifundefined{chapter}`).

```
}%
```

5.43 Macro pour écrire une ligne de contenu

\mtc@dot
\MTC@WriteContentsline
\mtc@param
\write
\tf@mtc
\@resetfont
\mtc@string
\contentsline
\mtc@toks

La macro `\MTC@WriteContentsline` rend la définition de `\MTC@contentsline` plus courte. Un niveau supplémentaire de `\edef` est retiré (Heiko Oberdiek) :

Les arguments de `\MTC@WriteContentsline` sont :
#1 : l'argument #1 de `\MTC@contentsline`;
#2 : un raccourci pour la fonte ⟹ `\csname #2font\endcsname`;
#3 : l'argument #3 de `\MTC@contentsline`;
#4 : l'argument #4 de `\MTC@contentsline` (hyperlien).

```
\def\mtc@dot{.}
\def\MTC@WriteContentsline#1#2#3#4{%
  \def\mtc@param{#4}%
   \immediate\write\tf@mtc{%
    {\string\reset@font
     \expandafter\string\csname #2font\endcsname
     \string\mtc@string
     \string\contentsline{#1}%
     {\the\mtc@toks}%
     {\string\reset@font
      \expandafter\string\csname #2font\endcsname
      \space #3%
     }%
      \ifx\mtc@dot\mtc@param
      \else
        {#4}% %%HO/BJ: #4 is hyperlink
      \fi
    }%
  }%
}
```

\MTC@WriteCoffeeline
\write
\tf@mtc
\@resetfont
\coffeefont
\mtc@string
\mtc@toks

Et la même chose pour une ligne de « pause café ». Les arguments de `\MTC@WriteCoffeeline` sont :
#1 : l'argument #1 de `\MTC@contentsline`;
#2 : l'argument #3 de `\MTC@contentsline`.

```
\def\MTC@WriteCoffeeline#1#2#3{%
  \immediate\write\tf@mtc{%
    {\string\reset@font
     \string\coffeefont
     \string\mtc@string
```

```
%% COFFEE \string\textbf{#1}%
     {\the\mtc@toks}%
     {\string\reset@font
      \string\coffeefont
      \space #3%
     }%
    }%
  }%
}
```

5.44 Compteurs de profondeur pour les partlofs et les partlots

\AtBeginDocument \newcounter \setcounter \c@lofdepth \c@lotdepth

Si les compteurs `lofdepth` et `lotdepth` sont définis, nous créons les nouveaux compteurs correspondants : `partlofdepth` et `partlotdepth`. Ces compteurs sont initialisés à 2. Ceci est fait après le chargement des paquetages, dans un bloc `\AtBeginDocument` :

```
\AtBeginDocument{%
 \@ifundefined{c@lofdepth}{}%
   {\newcounter{partlofdepth}\setcounter{partlofdepth}{2}}%
 \@ifundefined{c@lotdepth}{}%
   {\newcounter{partlotdepth}\setcounter{partlotdepth}{2}}%
}
```

5.45 Commandes au niveau partie

\xpart \theptc \Thepart \adjustptc \decrementptc \incrementptc \firstpartis \firstpartis \if@firstpartis@used@ \newcounter \setcounter

Si `\part` est définie, nous définissons quelques commandes utilitaires, un compteur (`ptc`) pour les parttocs et les commandes associées (`\theptc`, `\Thepart`, `\adjustptc`, `\decrementptc`, `\incrementptc`), la commande obsolète `\firstpartis` et le compteur de profondeur `parttocdepth`.

```
\@ifundefined{part}{}%
{%
\def\xpart{xpart}
\def\Thepart{\arabic{ptc}}
\def\firstpartis#1%
  {\PackageWarning{minitoc}%
     {*** \string\firstpartis \space is an obsolete command ***}%
     \@firstpartis@used@true}
\newcounter{ptc}
\setcounter{ptc}{0}
\newcommand{\adjustptc}[1][1]{\addtocounter{ptc}{#1}}
\def\decrementptc{\addtocounter{ptc}{-1}}
\def\incrementptc{\addtocounter{ptc}{+1}}
```

```
\def\theptc{\arabic{ptc}}
\newcounter{parttocdepth}
\setcounter{parttocdepth}{2}
```

\ptc@rule

Mais, parfois, nous voulons faire une différence entre les classes `book/report` et `article` (`\chapter` est-elle définie ?), pour avoir une présentation différente : la définition de `\ptc@rule` est vide sauf si `\chapter` n'est pas définie. Par défaut, il n'y a pas de filets avant et après les parttocs, partlofs et partlots pour les livres. Vous devriez redéclarer `\ptc@rule` si vous voulez ces filets.

```
\@ifundefined{chapter}%
   {\def\ptc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}}%
   {\let\ptc@rule\relax}
```

\ptcindent

Et nous déclarons l'indentation (des deux côtés) par défaut des parttocs :

```
\newlength\ptcindent
\@ifundefined{chapter}{\ptcindent=24\p@}{\ptcindent=\z@}
```

5.46 Fontes pour les parttocs

\ptcfont \ptcSfont \ptcSSfont \ptcSSSfont \ptcPfont \ptcSPfont \plffont \pltfont \ptifont

Nous définissons les fontes pour les parttocs. Notez qu'elles sont plus grandes si `\chapter` est définie (classes de document analogues à `book/report` que lorsqu'elle ne l'est pas (classes de document analogues à `article`) :

```
\@ifundefined{chapter}{%
   \def\ptcfont{\small\rmfamily\upshape\mdseries} % the parttoc
   \def\ptcSfont{\small\rmfamily\upshape\bfseries}% (sections)
   \let\ptcSSfont\ptcfont     % (subsections)
   \let\ptcSSSfont\ptcfont    % (subsubsections)
   \let\ptcPfont\ptcfont      % (paragraphs)
   \let\ptcSPfont\ptcfont     % (subparagraphs)
   \let\plffont\ptcfont       % (figures)
   \let\pltfont\ptcfont       % (tables)
   \def\ptifont{\Large\rmfamily\upshape\bfseries}% titles
}%
```

\ptcfont \ptcCfont \ptcSfont \ptcSSfont \ptcSSSfont \ptcPfont \ptcSPfont \plffont \pltfont \ptifont

Si `\chapter` est définie, les fontes sont plus grandes et `\ptcCfont` doit être définie :

```
{%
   \def\ptcfont{\normalsize\rmfamily\upshape\mdseries} % the parttoc
   \def\ptcCfont{\normalsize\rmfamily\upshape\bfseries}% (chapters)
   \def\ptcSfont{\normalsize\rmfamily\upshape\mdseries}% (sections)
   \let\ptcSSfont\ptcfont     % (subsections)
```

```
  \let\ptcSSSfont\ptcfont    % (subsubsections)
  \let\ptcPfont\ptcfont      % (paragraphs)
  \let\ptcSPfont\ptcfont     % (subparagraphs)
  \let\plffont\ptcfont       % (figures)
  \let\pltfont\ptcfont       % (tables)
  \def\ptifont{\LARGE\rmfamily\upshape\bfseries}% titles
}
```

5.47 Titres par défaut pour les mini-tables au niveau partie

\parttoc \partlof \partlot

Nous définissons la position par défaut, les fontes et la mise en page pour les titres des mini-tables du niveau partie (`\parttoc`, `\partlof` et `\partlot`). Cette mise en page est différentes selon que `\chapter` est non définie ou définie.

\c@pti \l@pti \r@pti \e@pti \n@pti

Si `\chapter` n'est pas définie, les définitions sont très simples, pour des titres centrés, cadrés à gauche, cadrés à droite, ou vides. Les titres vides ont besoin d'une correction verticale (Frank Mittelbach).

```
\@ifundefined{chapter}{%
\def\c@pti#1{\null\hfill #1\hfill\null}
\def\l@pti#1{\null #1\hfill\null}
\def\r@pti#1{\null\hfill #1\null}
\def\e@pti#1{\vspace{-\baselineskip}}
\def\n@pti#1{\vspace{-\baselineskip}}
}%
```

\e@pti \n@pti \if@twocolumn \@topnewpage \@makephead@l \@makephead@r \@makephead@c \@afterheading \ptifont

Mais, si `\chapter` est définie, nous devons simuler la mise en page d'une tête de chapitre, qui est plus complexe. Les titres vides ont besoin d'une correction verticale (Frank Mittelbach).

```
{%
\def\e@pti#1{\vspace{-\baselineskip}}
\def\n@pti#1{\vspace{-\baselineskip}}
```

\if@twocolumn \@topnewpage \@afterheading \ptifont \l@pti \@makephead@l

Pour un titre cadré à gauche, nous devons tester si le corps du texte est sur deux colonnes :

```
\def\l@pti#1{\if@twocolumn
    \@topnewpage[\@makephead@l{#1}]%
    \else
    \@makephead@l{#1}%
    \@afterheading
    \fi}
\def\@makephead@l#1{%
```

```
    \vspace*{50\p@}%
    {\parindent \z@ \raggedright
     \ptifont
     #1\par
     \nobreak
     \vskip 40\p@
    }}
```

\r@pti
\@makephead@l

Pour un titre cadré à droite, nous devons tester si le corps du texte est sur deux colonnes :

```
\def\r@pti#1{\if@twocolumn
    \@topnewpage[\@makephead@r{#1}]%
    \else
    \@makephead@r{#1}%
    \@afterheading
    \fi}
\def\@makephead@r#1{%
    \vspace*{50\p@}%
    {\parindent \z@ \raggedleft
     \ptifont
     #1\par
     \nobreak
     \vskip 40\p@
    }}
```

\c@pti
\@makephead@c

Pour un titre centré, nous devons aussi tester si le corps du texte est sur deux colonnes :

```
\def\c@pti#1{\if@twocolumn
    \@topnewpage[\@makephead@c{#1}]%
    \else
    \@makephead@c{#1}%
    \@afterheading
    \fi}
\def\@makephead@c#1{%
    \vspace*{50\p@}%
    {\parindent \z@ \centering
     \ptifont
     #1\par
     \nobreak
     \vskip 40\p@
    }}%
}
```

\l@pti
\do@ptitc
\df@ptitc
\do@ptilf
\df@ptilf
\do@ptilt
\df@ptilt

Par défaut, les titres sont à gauche :

```
\let\do@ptitc\l@pti
\let\df@ptitc\l@pti
\let\do@ptilf\l@pti
\let\df@ptilf\l@pti
\let\do@ptilt\l@pti
```

```
\let\df@ptilt\l@pti
```

5.48 L'environnement ptc@verse

ptc@verse \ptc@verse \endptc@verse \\ \@centercr \list \itemsep \itemindent \listparindent \topsep \parsep \parskip \partopsep \ptcindent \iftightmtc \ifktightmtc

Chaque parttoc est placée à l'intérieur d'un environnement `ptc@verse`. Cet environnement est analogue à l'environnement standard `verse` est est donc défini à l'aide de deux commandes : `\ptc@verse` et `\endptc@verse`. Puisque c'est un environnement de liste, nous définissons d'abord (de manière locale) `\\`, puis appelons `\list{}` et établissons quelques dimensions telles que `\itemsep`, `\itemindent`, `\listparindent`, `\itemindent`, `\partopsep`, `\topsep`. `\parsep` est forcée à zéro si l'option `tight` est active (ceci réduit l'espacement entre les lignes). `\parskip` est forcée à zéro si l'option `k-tight` est active (ceci réduit l'espacement entre les lignes). Les deux marges sont forcées à `\ptcindent`. `\endptc@verse` termine la liste et décourage une coupure de page.

```
\def\ptc@verse{\let\\=\@centercr
  \list{}{\itemsep\z@
          \itemindent \z@
          \listparindent \itemindent
          \partopsep\z@
          \iftightmtc \parsep\z@ \fi
          \ifktightmtc \parskip\z@ \fi
          \topsep=1ex
          \leftmargin\ptcindent
          \rightmargin\leftmargin}\item[]}
\def\endptc@verse{\nopagebreak[4]\endlist}
```

5.49 Les mini-tables au niveau partie : \parttoc, \partlof et \partlot

\parttoc \partlof \partlot

Ces commandes sont essentiellement similaires à la commande `\minitoc`, sauf qu'elles devraient être placées après une commande `\part` pour produire une parttoc, une partlof ou une partlot, et que la mise en page est différente et dépend de la disponibilité de la commande `\chapter` (pour les fontes et les filets horizontaux). Le code est très similaire. Les commandes `\partlof` et `\partlof` sont les cousines de la commande `\parttoc`. Notez que `\parttoc`, `\partlof` et `\partlot` utilisent des styles de page, car les commandes `\beforepart...` et `\afterpart...` impliquent habituellement une commande `\clear[double]page`, et donc `\markboth{...}{...}` doit être appelée.

5.49.1 La commande \parttoc

\parttoc
\parttoc@

Cette commande doit être utilisée après `\part` si vous souhaitez avoir une parttoc (pas de parttoc automatique). Tout d'abord, `\parttoc` détecte la présence de son argument optionnel, et utilise sa valeur par défaut, `d`, s'il est absent. Puis, `\parttoc@` est appelée avec la position effective comme argument :

```
\def\parttoc{\@ifnextchar[{\parttoc@}{\parttoc@[d]}}
```

\parttoc@
\if@parttoc@used@
\if@longextensions@
\@tocfile
\Thepart

La macro `\parttoc@` fait le véritable travail. Elle positionne d'abord l'indicateur `\if@parttoc@used@` (pour une indication (*hint*) de cohérence) et teste si les suffixes longs sont utilisés ou non (pour construire le nom du fichier parttoc) :

```
\def\parttoc@[#1]{%
\global\@parttoc@used@true
\if@longextensions@%
\def\@tocfile{ptc\Thepart}%
\else
\def\@tocfile{P\Thepart}%
\fi
```

\mtc@CkFile
\if@mtc@FE
\@tocfile

Puis, nous vérifions la présence du fichier parttoc et donnons un avertissement s'il n'est pas là :

```
        \mtc@CkFile{\jobname.\@tocfile}
        \if@mtc@FE
        \PackageInfo{minitoc}%
           {\jobname.\@tocfile\space is empty}
        \@mtc@empty@parttoc@true
        \else
```

\beforeparttoc

Si le fichier est présent, nous pouvons l'insérer, mais devons ajouter un peu de code pour la présentation ; d'abord, `\beforeparttoc`, bien sûr :

```
        \beforeparttoc
```

\mtc@markboth
\@mkboth
\thispageparttocstyle
\ptctitle

Si `\chapter` est définie, vous mettons simplement le titre de la parttoc dans les marques de page et établissons le style de page :

```
        \@ifundefined{chapter}{}{%
        \global\let\mtc@markboth\markboth
        \global\let\@mkboth\markboth
        \thispageparttocstyle
        \mtc@markboth{\uppercase{\ptctitle}}{\uppercase{\ptctitle}}%
        }%
```

\do@ptitc \e@pti \n@pti \c@pti \l@pti \r@pti \df@pti \mtc@CkStr \ptctitle \if@mtc@FE samepage

Nous débutons un environnement samepage, puis l'argument est traité pour établir le position du titre de la parttoc. Si la chaîne de titre est vide, le positionnement est forcé.

```
      \relax\begin{samepage}%
      \if #1e\let\do@ptitc\e@pti
      \else\if #1n\let\do@ptitc\n@pti
      \else\if #1c\let\do@ptitc\c@pti
      \else\if #1l\let\do@ptitc\l@pti
      \else\if #1r\let\do@ptitc\r@pti
      \else\if #1d\let\do@ptitc\df@ptitc
      \fi\fi\fi\fi\fi\fi
      \mtc@CkStr{\ptctitle}\if@mtc@FE \let\do@ptitc\e@pti\relax\fi
```

\raggedright \parskip \ptcfont

Nous ajustons quelques paramètres de mise en page et évitons une coupure de page entre le titre et la parttoc, puis établissons la fonte :

```
      \raggedright
      \parskip=\z@%
      \reset@font\ptcfont%
      \parindent=\z@%
      \nopagebreak[4]%
      \kern-0.8\baselineskip\nopagebreak[4]%
      \par\noindent
      \nopagebreak[4]%
```

\ptc@rule tabular \ptifont \do@ptitc \mtc@v \ptctitle \hline

Le titre de la parttoc est composé dans un environnement tabular (pour empêcher une coupure de page entre le titre et le filet du haut), avec un filet en dessous si nécessaire. Ce filet est une \hline. C'est le filet supérieur de la parttoc.

```
      \ifx\ptc@rule\relax
      \begin{tabular}{@{}p{\columnwidth}@{}}
      \reset@font\ptifont\do@ptitc{\mtc@v\ptctitle}\\
      \end{tabular}%
      \else
      \begin{tabular}{@{}p{\columnwidth}@{}}
      \reset@font\ptifont\do@ptitc{\mtc@v\ptctitle}\\\hline
      \end{tabular}%
      \fi
```

\mtc@zrule \mtc@BBR \ptcindent

Puis, nous ajustons la position en dessous du filet supérieur et établissons l'indentation et quelques paramètres de mise en page :

```
      \nopagebreak[4]\null\leavevmode\mtc@zrule\\*[-\baselineskip]\mtc@BBR
      \leftmargin\ptcindent
      \rightmargin\ptcindent
      \itemindent=\z@\labelwidth=\z@%
      \labelsep=\z@\listparindent=\z@%
```

ptc@verse
\c@tocdepth
\c@parttocdepth
\mtc@BBR

Nous entrons dans un environnement `ptc@verse` pour mettre en page la parttoc. La profondeur de table des matières est forcée (localement) à `parttocdepth`. Une petite astuce est nécessaire pour ajuster la position. Une ligne blanche est nécessaire pour éviter une indentation négative.

```
      \begin{ptc@verse}\c@tocdepth=\c@parttocdepth%
      \leavevmode\\\mtc@BBR
      \vskip -.5\baselineskip
```

\ptc@pgno
\@dottedtocline
\@undottedtocline
\mtc@hook@beforeinputfile
\ptc@setform
\@tocfile
\mtc@strut
ptc@verse

Si les lignes de contenu doivent ne pas avoir de numéros de pages, nous remplaçons la macro `\@dottedtocline` par sa version sans points de conduite. Pour les entrées au niveau chapitre, nous devons invoquer `\l@chapter` en ignorant l'argument numéro de page. Un « *hook* » (point d'attache, commande redéfinissable) est ajouté, et les choix de mise en page de `\mtcsetformat` sont activés `\ptc@setform`. Puis le fichier parttoc est inséré, suivi d'un étai, puis l'environnement `ptc@verse` est terminé.

```
\begingroup
 \makeatletter
 \@ifundefined{ptc@pgno}%
 {\let\@dottedtocline\@undottedtocline}{}
 \@ifundefined{ptc@pgno}%
 {\let\l@chapter@SVPN\l@chapter%
   \def\l@chapter##1##2{\l@chapter@SVPN{##1}{\hbox{}}}}{}
 \@fileswfalse\mtc@hook@beforeinputfile
 \ptc@setform
 \@input{\jobname.\@tocfile}
 \vspace{-1ex} \vspace{-1\baselineskip}
 \leavevmode\mtc@strut
 \global\@nobreakfalse\endgroup
      \end{ptc@verse}%
```

\ptc@rule
\mtc@zrule
samepage
\afterparttoc

La partie finale consiste seulement à ajouter le filet inférieur, si nécessaire, la possibilité d'une coupure de page (si `\chapter` n'est pas définie), et `\afterparttoc`.

```
      \kern-1.\baselineskip%
      \nopagebreak[4]\ptc@rule\null\leavevmode\\%
      \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
      \par\@ifundefined{chapter}{\pagebreak[1]\vspace*{-1ex}}%
      \afterparttoc\fi}%
```

5.49.2 La commande \partlof

\partlof
\partlof@

Cette commande doit être utilisée après `\part` si vous souhaitez avoir une partlof (pas de partlof automatique). Tout d'abord, `\partlof` détecte la présence de son argument optionnel, et utilise sa valeur par défaut, d, s'il est absent. Puis, \partlof@ est appelée avec la position effective comme argument :

```
\def\partlof{\@ifnextchar[{\partlof@}{\partlof@[d]}}
```

\partlof@ \if@partlof@used@ \if@longextensions@ \@tocfile \Thepart

La macro \partlof@ fait le véritable travail. Elle positionne d'abord l'indicateur \if@partlof@used@ (pour une indication (*hint*) de cohérence) et teste si les suffixes longs sont utilisés ou non (pour construire le nom du fichier partlof) :

```
\def\partlof@[#1]{%
\global\@partlof@used@true
\if@longextensions@%
\def\@tocfile{plf\Thepart}%
\else
\def\@tocfile{G\Thepart}%
\fi
```

\mtc@CkFile \if@mtc@FE \@tocfile

Puis, nous vérifions la présence du fichier partlof et donnons un avertissement s'il n'est pas là :

```
        \mtc@CkFile{\jobname.\@tocfile}
        \if@mtc@FE
        \PackageInfo{minitoc}%
           {\jobname.\@tocfile\space is empty}
        \@mtc@empty@partlof@true
        \else
```

\beforepartlof

Si le fichier est présent, nous pouvons l'insérer, mais devons ajouter un peu de code pour la présentation ; d'abord, \beforepartlof, bien sûr :

```
        \beforepartlof
```

\mtc@markboth \@mkboth \thispagepartlofstyle \plftitle

Si \chapter est définie, vous mettons simplement le titre de la partlof dans les marques de page et établissons le style de page :

```
        \@ifundefined{chapter}{}{%
        \global\let\mtc@markboth\markboth
        \global\let\@mkboth\markboth
        \thispagepartlofstyle
        \mtc@markboth{\uppercase{\plftitle}}{\uppercase{\plftitle}}%
        }%
```

\do@ptilf \e@pti \n@pti \c@pti \l@pti \r@pti \df@pti \mtc@CkStr \plftitle \if@mtc@FE samepage

Nous débutons un environnement samepage, puis l'argument est traité pour établir le position du titre de la partlof. Si la chaîne de titre est vide, le positionnement est forcé.

```
        \relax\begin{samepage}%
        \if #1e\let\do@ptilf\e@pti
        \else\if #1n\let\do@ptilf\n@pti
        \else\if #1c\let\do@ptilf\c@pti
        \else\if #1l\let\do@ptilf\l@pti
        \else\if #1r\let\do@ptilf\r@pti
        \else\if #1d\let\do@ptilf\df@ptilf
        \fi\fi\fi\fi\fi\fi
        \mtc@CkStr{\plftitle}\if@mtc@FE \let\do@ptilf\e@pti\relax\fi
```

\raggedright \parskip \plffont

Nous ajustons quelques paramètres de mise en page et évitons une coupure de page entre le titre et la partlof, puis établissons la fonte :

```
        \raggedright
        \parskip=\z@%
        \reset@font\plffont%
```

\plf@rule tabular \ptifont \do@ptilf \mtc@v \plftitle \hline

Le titre de la partlof est composé dans un environnement tabular (pour empêcher une coupure de page entre le titre et le filet du haut), avec un filet en dessous si nécessaire. Ce filet est une \hline. C'est le filet supérieur de la partlof.

```
        \parindent=\z@%
        \nopagebreak[4]%
        \kern-0.8\baselineskip\nopagebreak[4]%
        \par\noindent
        \ifx\plf@rule\relax
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\ptifont\do@ptilf{\mtc@v\plftitle}\\
        \end{tabular}%
        \else
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\ptifont\do@ptilf{\mtc@v\plftitle}\\\hline
        \mtc@hstrut\\
        \end{tabular}%
        \fi
```

\mtc@zrule \mtc@BBR \ptcindent

Puis, nous ajustons la position en dessous du filet supérieur et établissons l'indentation et quelques paramètres de mise en page :

```
        \nopagebreak[4]\null\leavevmode\mtc@zrule\\*[-\baselineskip]\mtc@BBR
        \leftmargin\ptcindent
        \rightmargin\ptcindent
        \itemindent=\z@\labelwidth=\z@%
        \labelsep=\z@\listparindent=\z@%
```

ptc@verse
\mtc@BBR

Nous entrons dans un environnement `ptc@verse` pour mettre en page la partlof. Si nécessaire la profondeur de table des matières est forcée (localement) à `partlofdepth`. Une petite astuce est nécessaire pour ajuster la position. Une ligne blanche est nécessaire pour éviter une indentation négative.

```
        \begin{ptc@verse}%
        \@ifundefined{c@lofdepth}{}{\c@tocdepth=\c@partlofdepth}
        \ifnum\c@tocdepth<1\relax\c@tocdepth=1\fi
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\plf@pgno
\@dottedtocline
\@undottedtocline
\mtc@hook@beforeinputfile
\plf@setform
\@tocfile
\mtc@strut
ptc@verse

Si les lignes de contenu doivent ne pas avoir de numéros de pages, nous remplaçons la macro `\@dottedtocline` par sa version sans points de conduite. Pour les entrées au niveau chapitre, nous devons invoquer `\l@chapter` en ignorant l'argument numéro de page. Un « *hook* » (point d'attache, commande redéfinissable) est ajouté, et les choix de mise en page de `\mtcsetformat` sont activés `\plf@setform`. Puis le fichier partlof est inséré, suivi d'un étai, puis l'environnement `ptc@verse` est terminé.

```
\begingroup
  \makeatletter
  \@ifundefined{plf@pgno}%
  {\let\@dottedtocline\@undottedtocline}{}
  \@fileswfalse\mtc@hook@beforeinputfile
  \plf@setform
  \@input{\jobname.\@tocfile}
  \vspace{-1ex} \vspace{-0\baselineskip}
  \leavevmode\mtc@strut
  \global\@nobreakfalse\endgroup
        \end{ptc@verse}%
```

\plf@rule
\mtc@zrule
samepage
\afterpartlof

La partie finale consiste seulement à ajouter le filet inférieur, si nécessaire, la possibilité d'une coupure de page (si `\chapter` n'est pas définie), et `\afterpartlof`. La ligne blanche (\\) est essentielle.

```
        \kern-1.\baselineskip%
        \nopagebreak[4]\plf@rule\null\leavevmode\\%
        \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
        \par\@ifundefined{chapter}{\pagebreak[1]\vspace*{-1ex}}%
        \afterpartlof\fi}
```

5.49.3 La commande \partlot

\partlot
\partlot@

Cette commande doit être utilisée après `\part` si vous souhaitez avoir une partlot (pas de partlot automatique). Tout d'abord, `\partlot` détecte la présence de son argument optionnel, et utilise sa valeur par défaut, d, s'il est absent. Puis, \partlot@ est appelée avec la position effective comme argument :

```
\def\partlot{\@ifnextchar[{\partlot@}{\partlot@[d]}}
```

\partlot@
\if@partlot@used@
\if@longextensions@
\@tocfile
\Thepart

La macro \partlot@ fait le véritable travail. Elle positionne d'abord l'indicateur \if@partlot@used@ (pour une indication (*hint*) de cohérence) et teste si les suffixes longs sont utilisés ou non (pour construire le nom du fichier partlot) :

```
%
\def\partlot@[#1]{%
\global\@partlot@used@true
\if@longextensions@%
\def\@tocfile{plt\Thepart}%
\else
\def\@tocfile{U\Thepart}%
\fi
```

\mtc@CkFile
\if@mtc@FE
\@tocfile

Puis, nous vérifions la présence du fichier partlot et donnons un avertissement s'il n'est pas là :

```
        \mtc@CkFile{\jobname.\@tocfile}
        \if@mtc@FE
        \PackageInfo{minitoc}%
           {\jobname.\@tocfile\space is empty}
        \@mtc@empty@partlof@true
        \else
```

\beforepartlot

Si le fichier est présent, nous pouvons l'insérer, mais devons ajouter un peu de code pour la présentation ; d'abord, \beforepartlot, bien sûr :

```
        \beforepartlot
```

\mtc@markboth
\@mkboth
\thispagepartlotstyle
\plttitle

Si \chapter est définie, vous mettons simplement le titre de la partlot dans les marques de page et établissons le style de page :

```
        \@ifundefined{chapter}{}{%
        \global\let\mtc@markboth\markboth
        \global\let\@mkboth\markboth
        \thispagepartlotstyle
        \mtc@markboth{\uppercase{\plttitle}}{\uppercase{\plttitle}}%
        }%
```

\do@ptilt \e@pti \n@pti \c@pti \l@pti \r@pti \df@pti \mtc@CkStr \plttitle \if@mtc@FE samepage

Nous débutons un environnement samepage, puis l'argument est traité pour établir le position du titre de la partlot. Si la chaîne de titre est vide, le positionnement est forcé.

```
       \relax\begin{samepage}%
       \if #1e\let\do@ptilt\e@pti
       \else\if #1n\let\do@ptilt\n@pti
       \else\if #1c\let\do@ptilt\c@pti
       \else\if #1l\let\do@ptilt\l@pti
       \else\if #1r\let\do@ptilt\r@pti
       \else\if #1d\let\do@ptilt\df@ptilt
       \fi\fi\fi\fi\fi\fi
       \mtc@CkStr{\plttitle}\if@mtc@FE \let\do@ptilt\e@pti\relax\fi
```

\raggedright \parskip \pltfont

Nous ajustons quelques paramètres de mise en page et évitons une coupure de page entre le titre et la partlot, puis établissons la fonte :

```
       \raggedright
       \parskip=\z@%
       \reset@font\pltfont%
       \parindent=\z@%
       \nopagebreak[4]%
       \kern-0.8\baselineskip\nopagebreak[4]%
       \par\noindent
```

\ptc@rule tabular \ptifont \do@ptilt \mtc@v \plttitle \hline

Le titre de la partlot est composé dans un environnement tabular (pour empêcher une coupure de page entre le titre et le filet du haut), avec un filet en dessous si nécessaire. Ce filet est une \hline. C'est le filet supérieur de la partlot.

```
       \ifx\plt@rule\relax
       \begin{tabular}{@{}p{\columnwidth}@{}}
       \reset@font\ptifont\do@ptilt{\mtc@v\plttitle}\\
       \end{tabular}%
       \else
       \begin{tabular}{@{}p{\columnwidth}@{}}
       \reset@font\ptifont\do@ptilt{\mtc@v\plttitle}\\\hline
       \mtc@hstrut\\
       \end{tabular}%
       \fi
```

\mtc@zrule \mtc@BBR \ptcindent

Puis, nous ajustons la position en dessous du filet supérieur et établissons l'indentation et quelques paramètres de mise en page :

```
       \nopagebreak[4]\null\leavevmode\mtc@zrule\\*[-\baselineskip]\mtc@BBR
       \leftmargin\ptcindent
       \rightmargin\ptcindent
       \itemindent=\z@\labelwidth=\z@%
       \labelsep=\z@\listparindent=\z@%
```

ptc@verse
\mtc@BBR

Nous entrons dans un environnement `ptc@verse` pour mettre en page la partlot. Si nécessaire la profondeur de table des matières est forcée (localement) à `partlotdepth`. Une petite astuce est nécessaire pour ajuster la position. Une ligne blanche est nécessaire pour éviter une indentation négative.

```
        \begin{ptc@verse}%
        \@ifundefined{c@lotdepth}{}{\c@tocdepth=\c@partlotdepth}
        \ifnum\c@tocdepth<1\relax\c@tocdepth=1\fi
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\plt@pgno
\@dottedtocline
\@undottedtocline
\mtc@hook@beforeinputfile
\plt@setform
\@tocfile
\mtc@strut
ptc@verse

Si les lignes de contenu doivent ne pas avoir de numéros de pages, nous remplaçons la macro `\@dottedtocline` par sa version sans points de conduite. Pour les entrées au niveau chapitre, nous devons invoquer `\l@chapter` en ignorant l'argument numéro de page. Un « *hook* » (point d'attache, commande redéfinissable) est ajouté, et les choix de mise en page de `\mtcsetformat` sont activés `\plt@setform`. Puis le fichier partlot est inséré, suivi d'un étai, puis l'environnement `ptc@verse` est terminé.

```
\begingroup
  \makeatletter
  \@ifundefined{plt@pgno}%
  {\let\@dottedtocline\@undottedtocline}{}
  \@fileswfalse\mtc@hook@beforeinputfile
  \plt@setform
  \@input{\jobname.\@tocfile}
  \vspace{-1ex} \vspace{-0\baselineskip}
  \leavevmode\mtc@strut
  \global\@nobreakfalse\endgroup
        \end{ptc@verse}%
```

\ptc@rule
\mtc@zrule
samepage
\afterpartlot

La partie finale consiste seulement à ajouter le filet inférieur, si nécessaire, la possibilité d'une coupure de page (si `\chapter` n'est pas définie), et `\afterpartlot`. La ligne blanche (`\\`) est essentielle.

```
        \kern-1.\baselineskip%
        \nopagebreak[4]\plt@rule\null\leavevmode\\%
        \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
        \par\@ifundefined{chapter}{\pagebreak[1]\vspace*{-1ex}}%
        \afterpartlot\fi}
```

5.50 Commandes auxiliaires pour l'impression des parttocs

\@dottedtocline
\l@xpart
\l@pchapter
\l@psection
\pchapter
\psection

Les commandes auxiliaires suivantes sont utilisées dans l'impression des parttocs. Notez que `\l@xpart` utilise une profondeur énorme pour inhiber l'impression de sa ligne de contenu (sauf si vous trichez). Ces commandes sont similaires à `\l@subsection`, seuls les arguments ont été altérés :

```
\def\l@xpart{\@dottedtocline{\@M}{1.0em}{2.3em}}
\def\l@pchapter{\@dottedtocline{1}{1.0em}{2.3em}}
\def\l@psection{\@dottedtocline{2}{1.0em}{2.3em}}
\def\pchapter{pchapter}
\def\psection{psection}
```

5.51 Modifier la commande \part, suite

`\sv@part` `\mtc@svpart` `\@part` `\addtocontents` `\sv@spart` `\ptc@spart` `\@spart` `\stepcounter` `\partbegin` `\partend`

Nous modifions les deux branches de la commande `\part` : `\@part` (`\part` non étoilée) et `\@spart` (`\part*`). Nous ajoutons l'incrémentation du compteur `ptc` dans les deux branches. Dans la branche non étoilée, nous ajoutons des entrées `xpart` dans les fichiers table des matières, liste des figures et liste des tableaux. Dans la branche étoilée, nous ajoutons une ligne `\partbegin` dans le fichier table des matières. Cette commande est juste un marqueur et ne fait rien de concret (`\relax`).

```
\let\sv@part\mtc@svpart
\def\@part[#1]#2{\sv@part[{#1}]{#2}\relax
\addcontentsline{lof}{xpart}{#1}%
\addcontentsline{lot}{xpart}{#1}%
\addcontentsline{toc}{xpart}{#1}%
\stepcounter{ptc}}
\let\sv@spart\@spart
\def\@spart{\stepcounter{ptc}\sv@spart}
\let\ptc@spart\@spart
\def\@spart{\addtocontents{toc}{\protect\partend}\ptc@spart}
\def\@spart{\addtocontents{toc}{\protect\partbegin}\ptc@spart}
\let\partend\relax
\let\partbegin\relax
```

5.52 La commande \doparttoc et ses cousines

`\doparttoc` `\dopartlof` `\dopartlot`

La commande `\doparttoc` fonctionne comme la commande `\dominitoc`, `\dopartlof` comme `\dominilof` et `\dopartlot` comme `\dominilot`.

`\@doparttoc` `\PTC@next` `\setcounter`

La commande `\doparttoc` extrait des informations du fichier `.toc` et crée les fichiers `.ptc`⟨*N*⟩ (`.ptc` devient `.P` sous MS-DOS).

```
\def\@doparttoc#1{{%
  \makeatletter
  \setcounter{ptc}{0}%
  \PTC@next#1.toc\relax\\}\setcounter{ptc}{0}}%
```

`\@dopartlof`
`\PLF@next`
`\setcounter`

La commande `\dopartlof` extrait des informations du fichier `.lof` et crée les fichiers `.plf`⟨*N*⟩ (`.plf` devient `.G` sous MS-DOS).

```
\def\@dopartlof#1{{%
  \makeatletter
  \setcounter{ptc}{0}%
  \PLF@next#1.lof\relax\\}\setcounter{ptc}{0}}%
```

`\@dopartlot`
`\PLT@next`
`\setcounter`

La commande `\dopartlot` extrait des informations du fichier `.lot` et crée les fichiers `.plt`⟨*N*⟩ (`.plt` devient `.U` sous MS-DOS).

```
\def\@dopartlot#1{{%
  \makeatletter
  \setcounter{ptc}{0}%
  \PLT@next#1.lot\relax\\}\setcounter{ptc}{0}}%
```

`\doparttoc`
`\dopartlof`
`\dopartlot`

Nous définissons les macros pour l'utilisateur, qui détectent l'argument optionnel :

```
\def\doparttoc{\@ifnextchar[{\doparttoc@}{\doparttoc@[l]}}
\def\dopartlof{\@ifnextchar[{\dopartlof@}{\dopartlof@[l]}}
\def\dopartlot{\@ifnextchar[{\dopartlot@}{\dopartlot@[l]}}
```

`\doparttoc@`
`\if@doparttoc@used@`
`\df@ptitc`
`\e@pti`
`\n@pti`
`\c@pti`
`\l@pti`
`\r@pti`

Nous traitons l'argument optionnel de `\doparttoc` (il devient la position par défaut pour les titres des parttocs) et marquons cette macro comme ayant été utilisée :

```
\def\doparttoc@[#1]{%
\global\@doparttoc@used@true
\if #1e\let\df@ptitc\e@pti%
\else\if #1n\let\df@ptitc\n@pti%
\else\if #1c\let\df@ptitc\c@pti%
\else\if #1l\let\df@ptitc\l@pti%
\else\if #1r\let\df@ptitc\r@pti%
\fi\fi\fi\fi\fi%
\@@doparttoc}
```

`\dopartlof@`
`\if@dopartlof@used@`
`\df@ptilf`
`\e@pti`
`\n@pti`
`\c@pti`
`\l@pti`
`\r@pti`

Nous traitons l'argument optionnel de `\dopartlof` (il devient la position par défaut pour les titres des partlofs) et marquons cette macro comme ayant été utilisée :

```
\def\dopartlof@[#1]{%
\global\@dopartlof@used@true
\if #1e\let\df@ptilf\e@pti%
\else\if #1n\let\df@ptilf\n@pti%
\else\if #1c\let\df@ptilf\c@pti%
\else\if #1l\let\df@ptilf\l@pti%
\else\if #1r\let\df@ptilf\r@pti%
\fi\fi\fi\fi\fi%
\@@dopartlof}
```

\dopartlot@ \if@dopartlot@used@ \df@ptilt \e@pti \n@pti \c@pti \l@pti \r@pti

Nous traitons l'argument optionnel de `\dopartlot` (il devient la position par défaut pour les titres des partlots) et marquons cette macro comme ayant été utilisée :

```
\def\dopartlot@[#1]{%
\global\@dopartlot@used@true
\if #1e\let\df@ptilt\e@pti%
\else\if #1n\let\df@ptilt\n@pti%
\else\if #1c\let\df@ptilt\c@pti%
\else\if #1l\let\df@ptilt\l@pti%
\else\if #1r\let\df@ptilt\r@pti%
\fi\fi\fi\fi\fi%
\@@dopartlot}
```

\@@doparttoc \@@dopartlof \@@dopartlot \tf@mtc

Ces macros invoquent les `\@dopart...` pour créer le fichier de mini-table, puis ferment le descripteur de fichier.

```
\def\@@doparttoc{\@doparttoc{\jobname}\immediate\closeout\tf@mtc}
\def\@@dopartlof{\@dopartlof{\jobname}\immediate\closeout\tf@mtc}
\def\@@dopartlot{\@dopartlot{\jobname}\immediate\closeout\tf@mtc}
```

5.52.1 Macros de traitement pour les parttocs

\PTC@next \PTC@list \PTC@loop

Nous traitons l'entrée suivante dans la liste puis la retirons de la tête de liste :

```
\def\PTC@next#1\relax#2\\{%
  \edef\PTC@list{#2}%
  \PTC@loop{#1}}
```

\PTC@toc \PTC@list \PTC@explist

Nous testons si la liste est vide :

```
\def\PTC@toc{%
  \ifx\PTC@list\@empty\else\expandafter\PTC@explist\fi}
```

\PTC@contentsline \part \theptc \tf@mtc \ptcname \MTC@WriteContentsLine

La macro `\PTC@contentsline` analyse les lignes lues dans le fichier table des matières et détecte les mots-clés intéressants. Si `\part` est trouvée, `ptc` est incrémenté et un nouveau fichier parttoc est créé.

```
\def\PTC@contentsline#1#2#3#4{% %%HO/DV: 4 instead of 3 parameters
  \expandafter\ifx\csname #1\endcsname\part
    \stepcounter{ptc}%
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.ptc\theptc\@gobble}%
      \def\ptcname{\jobname.ptc\theptc}%
    \else
      \PackageInfo{minitoc}%
```

```
      {Writing\space\jobname.P\theptc\@gobble}%
    \def\ptcname{\jobname.P\theptc}%
  \fi
  \immediate\closeout\tf@mtc
  \immediate\openout\tf@mtc=\ptcname
\fi
\expandafter\ifx\csname #1\endcsname\starpart\relax
  \stepcounter{ptc}%
```

\if@longextensions@ \ptcname

Nous testons si des suffixes longs ou courts sont utilisés, pour construire le nom du fichier mini-table, puis nous ouvrons ce fichier :

```
  \if@longextensions@%
    \PackageInfo{minitoc}%
      {Writing\space\jobname.ptc\theptc}%
    \def\ptcname{\jobname.ptc\theptc}%
  \else
    \PackageInfo{minitoc}%
      {Writing\space\jobname.P\theptc}%
    \def\ptcname{\jobname.P\theptc}%
  \fi
  \immediate\closeout\tf@mtc
  \immediate\openout\tf@mtc=\ptcname
\fi
```

\mtc@toks

Le registre *token* `\mtc@toks` est utilisé pour passer l'entrée à `\MTC@WriteContentsline` :

```
\mtc@toks{\noexpand\leavevmode #2}%
```

\MTC@WriteContentsline \chapter \pchapter \section \coffee \subsection \subsubsection \paragraph \subparagraph

Puis nous filtrons les lignes de contenu concernées ; ce code extrait et écrit les informations pour les chapitres, sections, etc. :

```
\expandafter\ifx\csname #1\endcsname\chapter
  \MTC@WriteContentsline{#1}{ptcC}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\pchapter
  \MTC@WriteContentsline{#1}{ptcC}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\section
  \MTC@WriteContentsline{#1}{ptcS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\coffee
  \MTC@WriteCoffeeline{#1}{#3}%
\fi
\expandafter\ifx\csname #1\endcsname\subsection
  \MTC@WriteContentsline{#1}{ptcSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\subsubsection
  \MTC@WriteContentsline{#1}{ptcSSS}{#3}{#4}%
```

```
  \fi
  \expandafter\ifx\csname #1\endcsname\paragraph
    \MTC@WriteContentsline{#1}{ptcP}{#3}{#4}%
  \fi
  \expandafter\ifx\csname #1\endcsname\subparagraph
    \MTC@WriteContentsline{#1}{ptcSP}{#3}{#4}%
  \fi
```

\MTC@WriteContentsline \starchapter \starsection \starsubsection \starsubsubsection \starparagraph \starsubparagraph

Et pour les commandes de sectionnement étoilées :

```
  \expandafter\ifx\csname #1\endcsname\starchapter
%%HO: the following line should be disabled: \stepcounter{ptc}%
    \MTC@WriteContentsline{#1}{ptcC}{#3}{#4}%
  \fi
  \expandafter\ifx\csname #1\endcsname\starsection
    \MTC@WriteContentsline{#1}{ptcS}{#3}{#4}%
  \fi
  \expandafter\ifx\csname #1\endcsname\starsubsection
    \MTC@WriteContentsline{#1}{ptcSS}{#3}{#4}%
  \fi
  \expandafter\ifx\csname #1\endcsname\starsubsubsection
    \MTC@WriteContentsline{#1}{ptcSSS}{#3}{#4}%
  \fi
  \expandafter\ifx\csname #1\endcsname\starparagraph
    \MTC@WriteContentsline{#1}{ptcP}{#3}{#4}%
  \fi
  \expandafter\ifx\csname #1\endcsname\starsubparagraph
    \MTC@WriteContentsline{#1}{ptcSP}{#3}{#4}%
  \fi
}
```

\PTC@explist \PTC@next \PTC@list

La boucle de lecture du fichier table des matières ; expanse la liste des entrées et appelle \PTC@next pour traiter la première :

```
\def\PTC@explist{\expandafter\PTC@next\PTC@list\\}
```

\PTC@loop \PTC@toc \PTC@read

Si une entrée est trouvée, on boucle ligne par ligne en recherchant des entrées intéressantes. Sinon, traiter l'entrée suivante de la liste.

```
\def\PTC@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
       {No file #1\MessageBreak PARTTOCS NOT PREPARED}%
    \expandafter\PTC@toc
  \else
    \PackageInfo{minitoc}%
       {PREPARING PARTTOCS FROM #1}%
    \expandafter\PTC@read\fi}
```

\PTC@read
\PTC@line

Lire l'entrée suivante dans le fichier `.toc`.

```
\def\PTC@read{%
  \read\@inputcheck to\PTC@line
```

\PTC@test
\PTC@line

Les `.....` garantissent que `\PTC@test` a suffisamment d'arguments :

```
  \expandafter\PTC@test\PTC@line.....\PTC@% %%HO: . added
  }%
```

\PTC@test

La macro `\PTC@test` trouve les commandes « intéressantes » dans le fichier table des matières, principalement pour délimiter les parties :

```
%%HO/BJ: now patch \PTC@test,
%%HO/BJ: call \PTC@contentsline with 4 instead of 3 parameters
```

\PTC@contentsline
\mtc@string
\PTC@list
\PTC@toc
\PTC@read
\partend

Nous examinons le premier *token* de la ligne. Si c'est une entrée « intéressante », nous la traitons. Si c'est `\@input`, nous ajoutons le fichier à la liste. Sinon, nous l'ignorons. La boucle est recommencée si la fin du fichier n'est pas atteinte. Enfin nous traitons le fichier suivant dans la liste.

```
\long\def\PTC@test#1#2#3#4#5#6\PTC@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \PTC@contentsline{#2}{#3}{#4}{#5}%
    %%HO/DV: 4 instead of 3 parameters
    \let\mtc@string\relax
  \else\ifx#1\@input
     \edef\PTC@list{\PTC@list#2\relax}%
  \else\ifx#1\partend
     \immediate\closeout\tf@mtc
     \immediate\openout\tf@mtc=\jobname.mtc
  \else\ifx#1\partbegin
     \addtocounter{ptc}{-1}%
  \fi\fi\fi\fi
  \ifeof\@inputcheck\expandafter\PTC@toc
  \else\expandafter\PTC@read\fi}%
```

5.52.2 Macros de traitement pour les partlofs

\PLF@next
\PLF@list
\PLF@loop

Nous traitons l'entrée suivante de la liste et le retirons de la tête de la liste :

```
\def\PLF@next#1\relax#2\\{%
  \edef\PLF@list{#2}%
  \PLF@loop{#1}}
```

\PLF@lof \PLF@list \PLF@explist

Nous testons si la liste est vide :

```
\def\PLF@lof{%
  \ifx\PLF@list\@empty\else\expandafter\PLF@explist\fi}
```

\PLF@contentsline \part \theptc \tf@mtc \plfname \PLF@WriteContentsLine

La macro `\PLF@contentsline` analyse les lignes lues dans le fichier liste des figures et détecte les mots-clés intéressants. Si `\part` est trouvée, `ptc` est incrémenté et un nouveau fichier partlof est créé.

```
\def\PLF@contentsline#1#2#3#4{% %%HO: #4 added
  \expandafter\ifx\csname #1\endcsname\xpart
    \stepcounter{ptc}%
```

\if@longextensions@ \plfname

Nous testons si des suffixes longs ou courts sont utilisés, pour construire le nom du fichier mini-table, puis nous ouvrons ce fichier :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.plf\theptc}%
      \def\plfname{\jobname.plf\theptc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.G\theptc}%
      \def\plfname{\jobname.G\theptc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\plfname
  \fi
```

\mtc@toks

Le registre *token* `\mtc@toks` est utilisé pour passer l'entrée à `\MTC@WriteContentsline` :

\figure \mtc@toks \MTC@WriteContentsline

Puis nous filtrons les lignes de contenu concernées :

```
  \expandafter\ifx\csname #1\endcsname\figure
    \mtc@toks{\noexpand\leavevmode#2}%
    \MTC@WriteContentsline{#1}{plf}{#3}{#4}%
  \fi
}
```

\PLF@explist \PLF@next \PLF@list

La boucle de lecture du fichier liste des figures ; expanse la liste des entrées et appelle `\PLF@next` pour traiter la première :

```
\def\PLF@explist{\expandafter\PLF@next\PLF@list\\}
```

\PLF@loop \PLF@lof \PLF@read

Si une entrée est trouvée, on boucle ligne par ligne en recherchant des entrées intéressantes. Sinon, traiter l'entrée suivante de la liste.

```
\def\PLF@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
       {No file #1\MessageBreak PARTLOFS NOT PREPARED}%
    \expandafter\PLF@lof
  \else
    \PackageInfo{minitoc}%
       {PREPARING PARTLOFS FROM #1}%
    \expandafter\PLF@read\fi}
```

\PLF@read
\PLF@line

Lire l'entrée suivante dans le fichier `.lof`.

```
\def\PLF@read{%
  \read\@inputcheck to\PLF@line
```

\PLF@test
\PLF@line

Les `.....` garantissent que `\PLF@test` a suffisamment d'arguments :

```
  \expandafter\PLF@test\PLF@line.....\PLF@% %%HO: . added
  }%
```

\PLF@test

La macro `\PLF@test` trouve les commandes « intéressantes » dans le fichier liste des figures, principalement pour délimiter les parties :

```
%%HO/BJ: now patch \PLF@test,
%%HO/BJ: call \PLF@contentsline with 4 instead of 3 parameters
```

\PLF@contentsline
\mtc@string
\PLF@list
\PLF@lof
\PLF@read
\partend

Nous examinons le premier *token* de la ligne. Si c'est une entrée « intéressante », nous la traitons. Si c'est `\@input`, nous ajoutons le fichier à la liste. Sinon, nous l'ignorons. La boucle est recommencée si la fin du fichier n'est pas atteinte. Enfin nous traitons le fichier suivant dans la liste.

```
\long\def\PLF@test#1#2#3#4#5#6\PLF@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \PLF@contentsline{#2}{#3}{#4}{#5}% %%HO: #4 added
    \let\mtc@string\relax
  \else\ifx#1\@input
     \edef\PLF@list{\PLF@list#2\relax}%
  \else\ifx#1\partend
     \immediate\closeout\tf@mtc
     \immediate\openout\tf@mtc=\jobname.mtc
  \else\ifx#1\partbegin
     \addtocounter{ptc}{-1}%
  \fi\fi\fi\fi
  \ifeof\@inputcheck\expandafter\PLF@lof
  \else\expandafter\PLF@read\fi}%
```

5.52.3 Macros de traitement pour les partlots

\PLT@next \PLT@list \PLT@loop

Nous traitons l'entrée suivante de la liste et le retirons de la tête de la liste :

```
\def\PLT@next#1\relax#2\\{%
  \edef\PLT@list{#2}%
  \PLT@loop{#1}}
```

\PLT@lot \PLT@list \PLT@explist

Nous testons si la liste est vide :

```
\def\PLT@lot{%
  \ifx\PLT@list\@empty\else\expandafter\PLT@explist\fi}
```

\PLT@contentsline \part \theptc \tf@mtc \pltname \PLT@WriteContentsLine

La macro `\PLT@contentsline` analyse les lignes lues dans le fichier liste des tableaux et détecte les mots-clés intéressants. Si `\part` est trouvée, `ptc` est incrémenté et un nouveau fichier partlot est créé.

```
\def\PLT@contentsline#1#2#3#4{% %%HO: #4 added
  \expandafter\ifx\csname #1\endcsname\xpart
    \stepcounter{ptc}%
```

\if@longextensions@ \pltname

Nous testons si des suffixes longs ou courts sont utilisés, pour construire le nom du fichier mini-table, puis nous ouvrons ce fichier :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.plt\theptc}%
      \def\pltname{\jobname.plt\theptc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.U\theptc}%
      \def\pltname{\jobname.U\theptc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\pltname
  \fi
```

\mtc@toks

Le registre *token* `\mtc@toks` est utilisé pour passer l'entrée à `\MTC@WriteContentsline` :

\table \mtc@toks \MTC@WriteContentsline

Puis nous filtrons les lignes de contenu concernées :

```
  \expandafter\ifx\csname #1\endcsname\table
    \mtc@toks{\noexpand\leavevmode#2}%
    \MTC@WriteContentsline{#1}{plt}{#3}{#4}%
  \fi
}
```

\PLT@explist \PLT@next \PLT@list

La boucle de lecture du fichier liste des tableaux ; expanse la liste des entrées et appelle \PLT@next pour traiter la première :

```
\def\PLT@explist{\expandafter\PLT@next\PLT@list\\}
```

\PLT@loop \PLT@lot \PLT@read

Si une entrée est trouvée, on boucle ligne par ligne en recherchant des entrées intéressantes. Sinon, traiter l'entrée suivante de la liste.

```
\def\PLT@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
       {No file #1\MessageBreak PARTLOTS NOT PREPARED}%
    \expandafter\PLT@lot
  \else
    \PackageInfo{minitoc}%
       {PREPARING PARTLOTS FROM #1}%
    \expandafter\PLT@read\fi}
```

\PLT@read \PLT@line

Lire l'entrée suivante dans le fichier .lot.

```
\def\PLT@read{%
  \read\@inputcheck to\PLT@line
```

\PLT@test \PLT@line

Les ..... garantissent que \PLT@test a suffisamment d'arguments :

```
  \expandafter\PLT@test\PLT@line.....\PLT@% %%HO: . added
  }%
```

\PLT@test

La macro \PLT@test trouve les commandes « intéressantes » dans le fichier liste des tableaux, principalement pour délimiter les parties :

```
%%HO/BJ: now patch \PLT@test,
%%HO/BJ: call \PLT@contentsline with 4 instead of 3 parameters
```

\PLT@contentsline \mtc@string \PLT@list \PLT@lot \PLT@read \partend

Nous examinons le premier *token* de la ligne. Si c'est une entrée « intéressante », nous la traitons. Si c'est \@input, nous ajoutons le fichier à la liste. Sinon, nous l'ignorons. La boucle est recommencée si la fin du fichier n'est pas atteinte. Enfin nous traitons le fichier suivant dans la liste.

```
\long\def\PLT@test#1#2#3#4#5#6\PLT@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \PLT@contentsline{#2}{#3}{#4}{#5}% %%HO: #4 added
    \let\mtc@string\relax
  \else\ifx#1\@input
     \edef\PLT@list{\PLT@list#2\relax}%
  \else\ifx#1\partend
```

```
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\jobname.mtc
\else\ifx#1\partbegin
    \addtocounter{ptc}{-1}%
\fi\fi\fi\fi
\ifeof\@inputcheck\expandafter\PLT@lot
\else\expandafter\PLT@read\fi}%
```

Fin du code pour le niveau partie (débuté dans la section 5.45 page 118) :

```
}%
```

5.53 Compteurs de profondeur pour les sectlofs et les sectlots

\AtBeginDocument
\c@lofdepth
\c@lotdepth
\newcounter
\setcounter

Si les compteurs `lofdepth` et `lotdepth` sont définis, nous créons de nouveaux compteurs pour les profondeurs des mini-tables correspondantes : `sectlofdepth` et `sectlotdepth`. Ces compteurs sont initialisés à 2. Ceci est fait après le chargement des paquetages, dans un bloc `\AtBeginDocument` :

```
\AtBeginDocument{%
\@ifundefined{c@lofdepth}{}%
  {\newcounter{sectlofdepth}\setcounter{sectlofdepth}{2}}%
\@ifundefined{c@lotdepth}{}%
  {\newcounter{sectlotdepth}\setcounter{sectlotdepth}{2}}%
}
```

5.54 Commandes au niveau section

\if@mtc@chapter@undef@
\if@mtc@section@def@

Les commandes au niveau section sont définies seulement si `\chapter` *n'est pas* définie, donx dans les classes de document du genre article, et seulement si `\section` est définie :

```
\if@mtc@chapter@undef@
\if@mtc@section@def@
```

\firstsectionis
\adjuststc
\decrementstc
\incrementstc
\stc@rule
\stcindent

Nous définissons la commande obsolète `\firstsectionis` (avec son avertissement bénin), le compteur `stc` de secttocs, les commandes `\adjuststc`, `\decrementstc` et `\incrementstc`, le compteur de profondeur `sectocdepth` et sa valeur par défaut 2 (pour inclure au moins les sous-sections), le filet horizontal `\stc@rule` (filet avant/après la secttoc/sectlof/sectlot), l'indentation (des deux côtés) `\stcindent` pour les secttocs (avec leurs valeurs par défaut).

```
\def\firstsectionis#1%
  {\PackageWarning{minitoc}%
     {*** \string\firstsectionis \space is an obsolete command ***}%
     \@firstsectionis@used@true}
\newcounter{stc}\setcounter{stc}{0}%
\newcommand{\adjuststc}[1][1]{\addtocounter{stc}{#1}}%
\def\decrementstc{\addtocounter{stc}{-1}}%
\def\incrementstc{\addtocounter{stc}{+1}}%
\newcounter{secttocdepth}\setcounter{secttocdepth}{2}%
\def\stc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}%
\newlength\stcindent \stcindent=24\p@
```

5.55 Commands de fontes pour les secttocs et co.

\stcfont \stcSSfont \stcSSSfont \stcPfont \stcSPfont \slffont \sltfont \stifont

Nous définissons les commandes de fontes pour les secttocs, sectlofs et sectlots et pour leurs titres :

```
\def\stcfont{\small\rmfamily\upshape\mdseries}    % secttoc
\def\stcSSfont{\small\rmfamily\upshape\bfseries}  % (subsections)
\let\stcSSSfont\stcfont    % (subsubsections)
\let\stcPfont\stcfont      % (paragraphs)
\let\stcSPfont\stcfont     % (subparagraphs)
\let\slffont\stcfont       % sectlof (figures)
\let\sltfont\stcfont       % sectlot (tables)
\def\stifont{\large\rmfamily\upshape\bfseries}    % titles
```

5.56 Macros internes pour le positionnement du titre

Voici quelques macros internes pour le positionnement du titre, selon les arguments optionnels des commandes \dosecttoc et \secttoc (et leurs cousines) :

\l@sti \c@sti \r@sti \e@sti \n@sti

Titres centrés, cadrés à gauche, cadrés à droite ou vides (avec un correction verticale pour les titres vides, d'après Frank Mittelbach) :

```
\def\c@sti#1{\null\hfill #1\hfill\null}
\def\l@sti#1{\null #1\hfill\null}
\def\r@sti#1{\null\hfill #1\null}
\def\e@sti#1{\vspace{-\baselineskip}}
\def\n@sti#1{\vspace{-\baselineskip}}
```

\do@stitc \df@stitc \do@stilf \df@stilf \do@stilt \df@stilt \l@sti

Par défaut, les titres sur cadrés à gauche.

```
\let\do@stitc\l@sti
```

```
\let\df@stitc\l@sti
\let\do@stilf\l@sti
\let\df@stilf\l@sti
\let\do@stilt\l@sti
\let\df@stilt\l@sti
```

5.57 L'environnement stc@verse

stc@verse
\iftightmtc
\ifktightmtc

L'environnement `stc@verse` est un environnement de liste très simple, analogue à l'environnement standard `verse`. Certains paramètres de mise en page sont ajustés. Les options de paquetage `tight/loose` et `k-tight/k-loose` sont honorées.

```
\def\stc@verse{\let\\=\@centercr
  \list{}{\itemsep\z@
          \itemindent \z@
          \listparindent \itemindent
          \partopsep \z@
          \iftightmtc \parsep \z@ \fi
          \ifktightmtc \parskip \z@ \fi
          \topsep \z@
          \leftmargin\stcindent
          \rightmargin\leftmargin}\item[]}
\def\endstc@verse{\nopagebreak[4]\endlist}
```

5.58 Les commandes \secttoc, \sectlof et \sectlot

Ces trois commandes sont très similaires.

5.58.1 La commande \secttoc

\secttoc
\secttoc@

La commande `\secttoc` doit être utilisée après `\section` si vous désirez une secttoc (pas de secttoc automatique). Son code est similaire à celui de `\minitoc` (mais plus simple). Tout d'abord, `\secttoc` détecte la présence de son argument optionnel, et utilise sa valeur par défaut, `d`, s'il est omis. Puis, `\secttoc@` est appelée avec la position effective comme argument :

```
\def\secttoc{\@ifnextchar[{\secttoc@}{\secttoc@[d]}}
```

\secttoc@
\if@secttoc@used@
\if@longextensions@
\@tocfile
\thestc

La macro `\secttoc@` effectue le vrai travail. Elle positionne d'abord l'indicateur `\if@secttoc@used@` (pour une indication (*hint*) de cohérence) est teste si les longs suffixes sont utilisés ou non (pour construire le nom du fichier secttoc) :

```
\def\secttoc@[#1]{%
\global\@secttoc@used@true
\if@longextensions@%
\def\@tocfile{stc\thestc}%
\else
\def\@tocfile{S\thestc}%
\fi
```

\mtc@CkFile
\if@mtc@FE
\@tocfile

Puis nous testons la présence et la vacuité du fichier secttoc, en donnant un avertissement s'il est absent ou est vide :

```
        \mtc@CkFile{\jobname.\@tocfile}
        \if@mtc@FE
        \PackageInfo{minitoc}%
           {\jobname.\@tocfile\space is empty}
        \@mtc@empty@secttoc@true
        \else
```

\beforesecttoc
\thispagesecttocstyle

Si le fichier secttoc est présent et non vide, nous pouvons l'insérer, mais nous devons au préalable ajouter du code de présentation : d'abord, `\beforesecttoc`, bien sûr, et le dispositif de style de page :

```
        \beforesecttoc
        \thispagesecttocstyle
```

\do@stitc
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\stctitle
\if@mtc@FE
samepage

Nous débutons un environnement `samepage`, puis traitons l'argument de positionnement. Si le titre est vide, nous simulons le positionnement « `e` ».

```
%%      \mtc@markboth{\uppercase{\stctitle}}{\uppercase{\stctitle}}%
        \relax\begin{samepage}%
        \if #1e\let\do@stitc\e@sti
        \else\if #1n\let\do@stitc\n@sti
        \else\if #1c\let\do@stitc\c@sti
        \else\if #1l\let\do@stitc\l@sti
        \else\if #1r\let\do@stitc\r@sti
        \else\if #1d\let\do@stitc\df@stitc
        \fi\fi\fi\fi\fi\fi
        \mtc@CkStr{\stctitle}\if@mtc@FE \let\do@stitc\e@sti\relax\fi
```

\raggedright
\parskip
\stcfont

Nous ajustons quelques paramètres de mise en page et évitons une coupure de page entre le titre et la secttoc, puis nous établissons la fonte :

```
        \raggedright
        \parskip=\z@%
        \reset@font\stcfont%
        \parindent=\z@%
        \nopagebreak[4]%
```

\stc@rule tabular \stifont \do@stitc \mtc@v \stctitle \hline

Le titre de la secttoc est composé dans un environnement `tabular` (pour empêcher une coupure de page entre le titre et le filet du haut), avec un filet en dessous si nécessaire. Ce filet est une `\hline`. C'est le filet supérieur de la secttoc.

```
        \kern-0.8\baselineskip\nopagebreak[4]%
        \par\noindent
        \nopagebreak[4]%
        \ifx\stc@rule\relax
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\stifont\do@stitc{\mtc@v\stctitle}\\
        \end{tabular}%
        \else
        \begin{tabular}{@{}p{\columnwidth}@{}}
        \reset@font\stifont\do@stitc{\mtc@v\stctitle}\\\hline
        \end{tabular}%
        \fi
```

\mtc@zrule \mtc@BBR \stcindent

Puis, nous ajustons la position en dessous du filet supérieur et établissons l'indentation et quelques paramètres de mise en page :

```
        \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
        \leftmargin\stcindent
        \rightmargin\stcindent
        \itemindent=\z@\labelwidth=\z@%
        \labelsep=\z@\listparindent=\z@%
```

stc@verse \c@tocdepth \c@secttocdepth \mtc@BBR

Nous débutons un environnement `stc@verse` pour mettre en page la secttoc. La profondeur de table des matières est forcée (localement) à `secttocdepth`. Une petite astuce est nécessaire pour ajuster le positionnement.

```
        \begin{stc@verse}\c@tocdepth=\c@secttocdepth%
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\stc@pgno \@dottedtocline \@undottedtocline \mtc@hook@beforeinputfile \stc@setform \@tocfile \mtc@strut stc@verse

Si les lignes de contenu ne doivent pas avoir de numéros, nous remplaçons la macro `\@dottedtocline` par sa version sans points de conduite. Un « *hook* » (point d'attache, commande redéfinissable) est ajouté, et les choix de mise en page issus de `\mtcsetformat` sont activés `\stc@setform`. Puis le fichier secttoc est inséré, suivi d'un étai, et l'environnement `stc@verse` est terminé.

```
\begingroup
  \makeatletter
  \stc@setform%
  \@ifundefined{stc@pgno}%
  {\let\@dottedtocline\@undottedtocline}{}
  \@fileswfalse\mtc@hook@beforeinputfile
  \stc@setform%
```

```
\@input{\jobname.\@tocfile}
\vspace{-1ex} \vspace{-\baselineskip}
\leavevmode\mtc@strut
\global\@nobreakfalse\endgroup
      \end{stc@verse}%
```

\stc@rule \mtc@zrule samepage \aftersecttoc

La partie finale ajoute le filet inférieur, si nécessaire, la permission d'une coupure de page et \aftersectoc.

```
      \kern-1.\baselineskip%
      \nopagebreak[4]\stc@rule\null\leavevmode\\%
      \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
      \par\pagebreak[1]\vspace*{-1ex}\aftersecttoc\fi}%
```

5.58.2 La commande \sectlof

\sectlof \sectlof@

La commande \sectlof doit être utilisée après \section si vous désirez une sectlof (pas de sectlof automatique). Son code est similaire à celui de \minilof (mais plus simple). Tout d'abord, \sectlof détecte la présence de son argument optionnel, et utilise sa valeur par défaut, d, s'il est omis. Puis, \sectlof@ est appelée avec la position effective comme argument :

```
\def\sectlof{\@ifnextchar[{\sectlof@}{\sectlof@[d]}}
```

\sectlof@ \if@sectlof@used@ \if@longextensions@ \@tocfile \thestc

La macro \sectlof@ effectue le vrai travail. Elle positionne d'abord l'indicateur \if@sectlof@used@ (pour une indication (*hint*) de cohérence) est teste si les longs suffixes sont utilisés ou non (pour construire le nom du fichier sectlof) :

```
\def\sectlof@[#1]{%
\global\@sectlof@used@true
\if@longextensions@%
\def\@tocfile{slf\thestc}%
\else
\def\@tocfile{H\thestc}%
\fi
```

\mtc@CkFile \if@mtc@FE \@tocfile

Puis nous testons la présence et la vacuité du fichier sectlof, en donnant un avertissement s'il est absent ou est vide :

```
      \mtc@CkFile{\jobname.\@tocfile}
      \if@mtc@FE
      \PackageInfo{minitoc}%
         {\jobname.\@tocfile\space is empty}
      \@mtc@empty@sectlof@true
      \else
```

\beforesectlof
\thispagesectlofstyle

Si le fichier sectlof est présent et non vide, nous pouvons l'insérer, mais nous devons au préalable ajouter du code de présentation : d'abord, `\beforesectlof`, bien sûr, et le dispositif de style de page :

```
      \thispagesectlofstyle
%%      \mtc@markboth{\uppercase{\slftitle}}{\uppercase{\slftitle}}%
      \beforesectlof
```

\do@stilf
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slftitle
\if@mtc@FE
samepage

Nous débutons un environnement `samepage`, puis traitons l'argument de positionnement. Si le titre est vide, nous simulons le positionnement « `e` ».

```
      \relax\begin{samepage}%
      \if #1e\let\do@stilf\e@sti
      \else\if #1n\let\do@stilf\n@sti
      \else\if #1c\let\do@stilf\c@sti
      \else\if #1l\let\do@stilf\l@sti
      \else\if #1r\let\do@stilf\r@sti
      \else\if #1d\let\do@stilf\df@stilf
      \fi\fi\fi\fi\fi\fi
      \mtc@CkStr{\slftitle}\if@mtc@FE \let\do@stilf\e@sti\relax\fi
```

\raggedright
\parskip
\slffont

Nous ajustons quelques paramètres de mise en page et évitons une coupure de page entre le titre et la sectlof, puis nous établissons la fonte :

```
      \raggedright
      \parskip=\z@%
      \reset@font\slffont%
      \parindent=\z@%
      \nopagebreak[4]%
```

\slf@rule
tabular
\stifont
\do@stilf
\mtc@v
\slftitle
\hline

Le titre de la sectlof est composé dans un environnement `tabular` (pour empêcher une coupure de page entre le titre et le filet du haut), avec un filet en dessous si nécessaire. Ce filet est une `\hline`. C'est le filet supérieur de la sectlof.

```
      \kern-0.8\baselineskip\nopagebreak[4]%
      \par\noindent
      \ifx\slf@rule\relax
      \begin{tabular}{@{}p{\columnwidth}@{}}
      \reset@font\stifont\do@stilf{\mtc@v\slftitle}\\
      \end{tabular}%
      \else
      \begin{tabular}{@{}p{\columnwidth}@{}}
      \mtc@hstrut
      \reset@font\stifont\do@stilf{\mtc@v\slftitle}\\\hline
      \end{tabular}%
      \fi
```

\mtc@zrule \mtc@BBR \stcindent

Puis, nous ajustons la position en dessous du filet supérieur et établissons l'indentation et quelques paramètres de mise en page :

```
        \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
        \leftmargin\stcindent
        \rightmargin\stcindent
        \itemindent=\z@\labelwidth=\z@%
        \labelsep=\z@\listparindent=\z@%
```

stc@verse \c@tocdepth \c@sectlofdepth \mtc@BBR

Nous débutons un environnement `stc@verse` pour mettre en page la sectlof. La profondeur de table des matières est forcée (localement) à `sectlofdepth`. Une petite astuce est nécessaire pour ajuster le positionnement.

```
        \begin{stc@verse}%
        \@ifundefined{c@lofdepth}{}{\c@tocdepth=\c@sectlofdepth}
        \ifnum\c@tocdepth<1\relax\c@tocdepth=1\fi
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\slf@pgno \@dottedtocline \@undottedtocline \mtc@hook@beforeinputfile \slf@setform \@tocfile \mtc@strut stc@verse

Si les lignes de contenu ne doivent pas avoir de numéros, nous remplaçons la macro `\@dottedtocline` par sa version sans points de conduite. Un « *hook* » (point d'attache, commande redéfinissable) est ajouté, et les choix de mise en page issus de `\mtcsetformat` sont activés `\slf@setform`. Puis le fichier sectlof est inséré, suivi d'un étai, et l'environnement `stc@verse` est terminé.

```
\begingroup
  \makeatletter
  \@ifundefined{slf@pgno}%
  {\let\@dottedtocline\@undottedtocline}{}
  \@fileswfalse\mtc@hook@beforeinputfile
  \slf@setform%
  \@input{\jobname.\@tocfile}
  \global\@nobreakfalse\endgroup
        \end{stc@verse}%
```

\stc@rule \mtc@zrule samepage \aftersecttoc

La partie finale ajoute le filet inférieur, si nécessaire, la permission d'une coupure de page et `\afterseclof`. La ligne blanche (`\\`) est essentielle.

```
        \kern-0.\baselineskip% ABCD
        \nopagebreak[4]\slf@rule\null\leavevmode\\%
        \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
        \par\pagebreak[1]\vspace*{-1ex}\aftersectlof\fi}%
```

5.58.3 La commande \sectlot

\sectlot
\sectlot@

La commande \sectlot doit être utilisée après \section si vous désirez une sectlot (pas de sectlot automatique). Son code est similaire à celui de \minilot (mais plus simple). Tout d'abord, \sectlot détecte la présence de son argument optionnel, et utilise sa valeur par défaut, d, s'il est omis. Puis, \sectlot@ est appelée avec la position effective comme argument :

```
\def\sectlot{\@ifnextchar[{\sectlot@}{\sectlot@[d]}}
```

\sectlot@
\if@sectlot@used@
\if@longextensions@
\@tocfile
\thestc

La macro \sectlot@ effectue le vrai travail. Elle positionne d'abord l'indicateur \if@sectlot@used@ (pour une indication (*hint*) de cohérence) est teste si les longs suffixes sont utilisés ou non (pour construire le nom du fichier sectlot) :

```
\def\sectlot@[#1]{%
\global\@sectlot@used@true
\if@longextensions@%
\def\@tocfile{slt\thestc}%
\else
\def\@tocfile{I\thestc}%
\fi
```

\mtc@CkFile
\if@mtc@FE
\@tocfile

Puis nous testons la présence et la vacuité du fichier sectlot, en donnant un avertissement s'il est absent ou est vide :

```
        \mtc@CkFile{\jobname.\@tocfile}
        \if@mtc@FE
        \PackageInfo{minitoc}%
           {\jobname.\@tocfile\space is empty}
        \@mtc@empty@sectlot@true
        \else
```

\beforesectlot
\thispagesectlotstyle

Si le fichier sectlot est présent et non vide, nous pouvons l'insérer, mais nous devons au préalable ajouter du code de présentation : d'abord, \beforesectlot, bien sûr, et le dispositif de style de page :

```
        \thispagesectlotstyle
%%      \mtc@markboth{\uppercase{\slttitle}}{\uppercase{\slttitle}}%
        \beforesectlot
```

\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
samepage

Nous débutons un environnement samepage, puis traitons l'argument de positionnement. Si le titre est vide, nous simulons le positionnement « e ».

```
        \relax\begin{samepage}%
        \if #1e\let\do@stilt\e@sti
        \else\if #1n\let\do@stilt\n@sti
        \else\if #1c\let\do@stilt\c@sti
```

```
\else\if #1l\let\do@stilt\l@sti
\else\if #1r\let\do@stilt\r@sti
\else\if #1d\let\do@stilt\df@stilt
\fi\fi\fi\fi\fi\fi
\mtc@CkStr{\slttitle}\if@mtc@FE \let\do@stilt\e@sti\relax\fi
```

`\raggedright` `\parskip` `\sltfont`

Nous ajustons quelques paramètres de mise en page et évitons une coupure de page entre le titre et la sectlot, puis nous établissons la fonte :

```
\raggedright
\parskip=\z@%
\reset@font\sltfont%
\parindent=\z@%
\nopagebreak[4]%
```

`\stc@rule` `tabular` `\stifont` `\do@stilt` `\mtc@v` `\slttitle` `\hline`

Le titre de la sectlot est composé dans un environnement `tabular` (pour empêcher une coupure de page entre le titre et le filet du haut), avec un filet en dessous si nécessaire. Ce filet est une `\hline`. C'est le filet supérieur de la sectlot.

```
\kern-0.8\baselineskip\nopagebreak[4]%
\par\noindent
\ifx\slt@rule\relax
\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\stifont\do@stilt{\mtc@v\slttitle}\\
\end{tabular}%
\else
\begin{tabular}{@{}p{\columnwidth}@{}}
\mtc@hstrut
\reset@font\stifont\do@stilt{\mtc@v\slttitle}\\\hline
\end{tabular}%
\fi
```

`\mtc@zrule` `\mtc@BBR` `\stcindent`

Puis, nous ajustons la position en dessous du filet supérieur et établissons l'indentation et quelques paramètres de mise en page :

```
\nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
\leftmargin\stcindent
\rightmargin\stcindent
\itemindent=\z@\labelwidth=\z@%
\labelsep=\z@\listparindent=\z@%
```

`stc@verse` `\c@tocdepth` `\c@sectlotdepth` `\mtc@BBR`

Nous débutons un environnement `stc@verse` pour mettre en page la sectlot. La profondeur de table des matières est forcée (localement) à `sectlotdepth`. Une petite astuce est nécessaire pour ajuster le positionnement.

```
\begin{stc@verse}%
\@ifundefined{c@lotdepth}{}{\c@tocdepth=\c@sectlotdepth}
\ifnum\c@tocdepth<1\relax\c@tocdepth=1\fi
```

```
        \leavevmode\\\mtc@BBR
        \vskip -.5\baselineskip
```

\slt@pgno \@dottedtocline \@undottedtocline \mtc@hook@beforeinputfile \slt@setform \@tocfile \mtc@strut stc@verse

Si les lignes de contenu ne doivent pas avoir de numéros, nous remplaçons la macro `\@dottedtocline` par sa version sans points de conduite. Un « *hook* » (point d'attache, commande redéfinissable) est ajouté, et les choix de mise en page issus de `\mtcsetformat` sont activés `\slt@setform`. Puis le fichier sectlot est inséré, suivi d'un étai, et l'environnement `stc@verse` est terminé.

```
\begingroup
  \makeatletter
  \@ifundefined{slt@pgno}%
  {\let\@dottedtocline\@undottedtocline}{}
   \gdef\thestc{\arabic{stc}}
  \@fileswfalse\mtc@hook@beforeinputfile
  \slt@setform%
  \@input{\jobname.\@tocfile}
  \global\@nobreakfalse\endgroup
        \end{stc@verse}%
```

\stc@rule \mtc@zrule samepage \aftersectlot

La partie finale ajoute le filet inférieur, si nécessaire, la permission d'une coupure de page et `\afterseclot`.

```
        \kern-0.\baselineskip% ABCD
        \nopagebreak[4]\slt@rule\null\leavevmode\\%
        \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
        \par\pagebreak[1]\vspace*{-1ex}\aftersectlot\fi}%
```

5.59 Commandes internes auxiliaires, niveau section

\l@xsect \@dottedtocline \l@schapter \xsect \schapter

Nous définissons quelques commandes auxiliaires, utilisées pour les mini-tables et comme délimiteurs dans le fichier table des matières (et les fichiers liste des figures et liste des tableaux). La profonteur de `xsect` est énorme pour inhiber l'impression de ses lignes de contenu (sauf si vous trichez).

```
\def\l@xsect{\@dottedtocline{\@M}{1.0em}{2.3em}}
\def\l@schapter{\@dottedtocline{1}{1.0em}{2.3em}}
\def\xsect{xsect}
\def\schapter{schapter}
```

5.60 Modifier la commande \section (suite)

\@sect \addcontentsline

Nous modifions les deux branches de la commande \section : \@sect pour la version non étoilée et \@ssect Pour la version étoilée. Tout d'abord, pour la version non étoilée (\@sect), nous ajoutons une ligne de contenu xsect dans la liste des figures et dans la liste des tableaux. Le test \ifnum #2=1 restreint l'action aux macros de niveau section (parce que \@sect est aussi utilisée par \subsection et en dessous, qui n'ont pas de mini-tables).

```
\let\sv@sect\@sect
\gdef\@sect#1#2#3#4#5#6[#7]#8{%
\ifnum #2=1
   \addcontentsline{lof}{xsect}{#7}%
   \addcontentsline{lot}{xsect}{#7}%
\fi
\sv@sect{#1}{#2}{#3}{#4}{#4}{#5}{#6}[{#7}]{#8}}
```

\section \starsection \addcontentsline

Si c'est une section (non étoilée ou étoilée via \starsection), nous ajoutons une entrée xsect dans la liste des figures et dans la liste des tableaux.

```
\def\@sect#1#2#3#4#5#6[#7]#8{
\expandafter
\ifx\csname #1\endcsname\section\relax
   \addcontentsline{lof}{xsect}{#7}%
   \addcontentsline{lot}{xsect}{#7}%
\fi
\ifx\csname #1\endcsname\starsection\relax
   \addcontentsline{lof}{xsect}{#7}%
   \addcontentsline{lot}{xsect}{#7}%
\fi
```

\@svsec \refstepcounter \@tempskipa \@hangfrom \addcontentsline \numberline \@svsechd \@xsect

Et le reste de la mise en page d'un en-tête de section :

```
\ifnum #2>\c@secnumdepth
   \let\@svsec\@empty
\else
   \refstepcounter{#1}%
   \edef\@svsec{\csname the#1\endcsname\hskip 1em}%
\fi
\@tempskipa #5\relax
\ifdim \@tempskipa>\z@
   \begingroup #6\relax
      \@hangfrom{\hskip #3\relax\@svsec}%
         {\interlinepenalty \@M #8\par}%
   \endgroup
   \csname #1mark\endcsname{#7}\addcontentsline
   {toc}{#1}{\ifnum #2>\c@secnumdepth
               \else
               \protect\numberline{\csname the#1\endcsname}%
               \fi
```

```
            #7}%
\else
   \def\@svsechd{#6\hskip #3\relax
   \@svsec #8\csname #1mark\endcsname
   {#7}\addcontentsline
   {toc}{#1}{\ifnum #2>\c@secnumdepth
              \else
                 \protect\numberline{\csname the#1\endcsname}
              \fi
   #7}}%
\fi
\@xsect{#5}}
```

\@ssect \sectbegin \sectend \stc@ssect \addtocontents

Puis nous modifions la branche étoilée (`\@ssect`). Nous définissons aussi les commandes de délimitation `\sectbegin` et `\sectend`.

```
\let\stc@ssect\@ssect
\def\@ssect{\addtocontents{toc}{\protect\sectend}\stc@ssect}
\def\@ssect{\addtocontents{toc}{\protect\sectbegin}\stc@ssect}
\let\sectend\relax
\let\sectbegin\relax
```

5.61 La commande \dosecttoc et ses cousines

La commande `\dosectoc` est très similaire à `\dominitoc`.

\dosecttoc \@dosecttoc \STC@next

La commande `\dosecttoc` extrait des informations du fichier `.toc` et crée les fichiers `.stc`⟨N⟩ (`.stc` devient `.S` sous MS-DOS).

```
\def\@dosecttoc#1{{%
  \makeatletter
  \setcounter{stc}{0}
  \STC@next#1.toc\relax\\}\setcounter{stc}{0}}
```

\dosectlof \@dosectlof \SLF@next

La commande `\dosectlof` extrait des informations du fichier `.lof` et crée les fichiers `.slf`⟨N⟩ (`.slf` devient `.H` sous MS-DOS).

```
\def\@dosectlof#1{{%
  \makeatletter
  \setcounter{stc}{0}
  \SLF@next#1.lof\relax\\}\setcounter{stc}{0}}
```

\dosectlot
\@dosectlot
\PLT@next

La commande \dosectlot extrait des informations du fichier .lot et crée les fichiers .slt⟨*N*⟩ (.slt devient .V sous MS-DOS).

```
\def\@dosectlot#1{{%
  \makeatletter
  \setcounter{stc}{0}
  \SLT@next#1.lot\relax\\}\setcounter{stc}{0}}
```

\dosecttoc
\dosectlof
\dosectlot

Nous définissons les commandes pour l'utilisateur, commandes qui détectent l'argument optionnel :

```
\def\dosecttoc{\@ifnextchar[{\dosecttoc@}{\dosecttoc@[l]}}
\def\dosectlof{\@ifnextchar[{\dosectlof@}{\dosectlof@[l]}}
\def\dosectlot{\@ifnextchar[{\dosectlot@}{\dosectlot@[l]}}
```

\dosecttoc@
\if@dosecttoc@used@
\df@stitc
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti

Nous traitons l'argument optionnel de \dosecttoc (il devient la position par défaut pour les titres des secttocs) et marquons cette macro comme ayant été utilisée :

```
\def\dosecttoc@[#1]{%
\global\@dosecttoc@used@true
\if #1e\let\df@stitc\e@sti%
\else\if #1n\let\df@stitc\n@sti%
\else\if #1c\let\df@stitc\c@sti%
\else\if #1l\let\df@stitc\l@sti%
\else\if #1r\let\df@stitc\r@sti%
\fi\fi\fi\fi\fi%
\@@dosecttoc}
```

\dosectlof@
\if@dosectlof@used@
\df@stilf
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti

Nous traitons l'argument optionnel de \dosectlof (il devient la position par défaut pour les titres des sectlofs) et marquons cette macro comme ayant été utilisée :

```
\def\dosectlof@[#1]{%
\global\@dosectlof@used@true
\if #1e\let\df@stilf\e@sti%
\else\if #1n\let\df@stilf\n@sti%
\else\if #1c\let\df@stilf\c@sti%
\else\if #1l\let\df@stilf\l@sti%
\else\if #1r\let\df@stilf\r@sti%
\fi\fi\fi\fi\fi%
\@@dosectlof}
```

\dosectlot@
\if@dosectlot@used@
\df@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti

Nous traitons l'argument optionnel de \dosectlot (il devient la position par défaut pour les titres des sectlots) et marquons cette macro comme ayant été utilisée :

```
\def\dosectlot@[#1]{%
\global\@dosectlot@used@true
\if #1e\let\df@stilt\e@sti%
\else\if #1n\let\df@stilt\n@sti%
```

```
\else\if #1c\let\df@stilt\c@sti%
\else\if #1l\let\df@stilt\l@sti%
\else\if #1r\let\df@stilt\r@sti%
\fi\fi\fi\fi\fi%
\@@dosectlot}
```

\@@dosecttoc \@@dosectlof \@@dosectlot \tf@mtc

Ces macros invoquent les commandes `\@dosect...` pour créer le fichier mini-table, puis ferment le descripteur de fichier.

```
\def\@@dosecttoc{\@dosecttoc{\jobname}\immediate\closeout\tf@mtc}
\def\@@dosectlof{\@dosectlof{\jobname}\immediate\closeout\tf@mtc}
\def\@@dosectlot{\@dosectlot{\jobname}\immediate\closeout\tf@mtc}
```

\STC@next \STC@list \STC@loop

Nous traitons l'entrée suivante dans la liste et la retirons de la tête de la liste :

```
\def\STC@next#1\relax#2\\{%
  \edef\STC@list{#2}%
  \STC@loop{#1}}
```

\STC@toc \STC@list \STC@explist

Nous testons si la liste est vide :

```
\def\STC@toc{%
  \ifx\STC@list\@empty\else\expandafter\STC@explist\fi}
```

\STC@contentsline \section \thestc \tf@mtc \stcname \MTC@WriteContentsLine

La macro `\STC@contentsline` analyse les lignes lues dans le fichier table des matières. Si `\section` est trouvée, `stc` est incrémenté et un nouveau fichier secttoc est créé.

```
\def\STC@contentsline#1#2#3#4{% %%HO: #4 added
  \gdef\thestc{\arabic{stc}}% %%HO: space removed
  \expandafter\ifx\csname #1\endcsname\section
    \stepcounter{stc}%
%%    \def\thestc{\arabic{stc}} %% HO: removed
```

\if@longextensions@ \stcname

Nous testons si les suffixes longs ou courts sont utilisés, pour construire le nom du fichier mini-table, puis nous l'ouvrons :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.stc\thestc}%
      \def\stcname{\jobname.stc\thestc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.S\thestc}%
      \def\stcname{\jobname.S\thestc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\stcname
  \fi
```

\mtc@toks
\MTC@WriteCoffeeline
\MTC@WriteContentsline
\coffee
\subsection
\subsubsection
\paragraph
\subparagraph

Puis nous filtrons les lignes de contenu concernées :

```
\mtc@toks{\noexpand\leavevmode #2}%
\expandafter\ifx\csname #1\endcsname\coffee
  \MTC@WriteCoffeeline{#1}{#3}%
\fi
\expandafter\ifx\csname #1\endcsname\subsection
  \MTC@WriteContentsline{#1}{stcSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\subsubsection
  \MTC@WriteContentsline{#1}{stcSSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\paragraph
  \MTC@WriteContentsline{#1}{stcP}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\subparagraph
  \MTC@WriteContentsline{#1}{stcSP}{#3}{#4}%
\fi
```

\starsection
\stepcounter
\thestc
\arabic
\if@longextensions@
\stcname
\closeout
\openout

Une section étoilée termine la section courante et crée un nouveau fichier secttoc :

```
\ifx\csname #1\endcsname\starsection
  \stepcounter{stc}%
  \gdef\thestc{\arabic{stc}}
  \if@longextensions@%
    \PackageInfo{minitoc}%
       {Writing\space\jobname.stc\thestc}%
    \def\stcname{\jobname.stc\thestc}%
  \else
    \PackageInfo{minitoc}%
       {Writing\space\jobname.S\thestc}%
    \def\stcname{\jobname.S\thestc}%
  \fi
  \immediate\closeout\tf@mtc
  \immediate\openout\tf@mtc=\stcname
\fi
```

\MTC@WriteContentsline
\starsubsection
\starsubsubsection
\starparagraph
\starsubparagraph

Nous traitons les entrées pour les commandes de sectionnement étoilées :

```
\expandafter\ifx\csname #1\endcsname\starsubsection
  \MTC@WriteContentsline{#1}{stcSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\starsubsubsection
  \MTC@WriteContentsline{#1}{stcSSS}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\starparagraph
  \MTC@WriteContentsline{#1}{stcP}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\starsubparagraph
  \MTC@WriteContentsline{#1}{stcSP}{#3}{#4}%
\fi
}
```

\STC@explist \STC@next \STC@list

Voici la boucle de lecture des lignes du fichier table des matières ; nous expansons la liste des entrées et appelons `\STC@next` pour traiter la première.

```
\def\STC@explist{\expandafter\STC@next\STC@list\\}
```

\STC@loop \STC@toc \STC@read

Si une entrée est trouvée, on examine ligne par ligne, en recherchant les entrées « intéressantes ». Sinon, nous traitons l'entrée suivante dans la liste.

```
\def\STC@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
      {No file #1\MessageBreak SECTTOCS NOT PREPARED}%
    \expandafter\STC@toc
  \else
    \PackageInfo{minitoc}%
      {PREPARING SECTTOCS FROM #1}%
    \expandafter\STC@read\fi}
```

\STC@read \STC@line

Nous lisons l'entrée suivante dans le fichier `.toc`.

```
\def\STC@read{%
  \read\@inputcheck to\STC@line
```

\STC@test \STC@line

Les `.....` garantissent que `\STC@test` a suffisamment d'arguments :

```
  \expandafter\STC@test\STC@line.....\STC@% %%HO: . added
  }%
```

\STC@test \STC@contentsline \mtc@string \STC@list \STC@toc \STC@read \sectend \sectbegin

La macro `\STC@test` trouve les commandes « intéressantes » dans le fichier table des matières, essentiellement pour délimiter les sections :

```
\long\def\STC@test#1#2#3#4#5#6\STC@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \STC@contentsline{#2}{#3}{#4}{#5}% %%HO: #4 added
    \let\mtc@string\relax
  \else\ifx#1\@input
     \edef\STC@list{\STC@list#2\relax}%
  \else\ifx#1\sectend
     \immediate\closeout\tf@mtc
     \immediate\openout\tf@mtc=\jobname.mtc
  \else\ifx#1\sectbegin
     \addtocounter{stc}{-1}%
  \fi\fi\fi\fi
  \ifeof\@inputcheck\expandafter\STC@toc
  \else\expandafter\STC@read\fi}%
```

\SLF@next
\SLF@list
\SLF@loop

Nous traitons l'entrée suivante dans la liste et la retirons de la tête de la liste :

```
\def\SLF@next#1\relax#2\\{%
  \edef\SLF@list{#2}%
  \SLF@loop{#1}}
```

\SLF@lof
\SLF@list
\SLF@explist

Nous testons si la liste est vide :

```
\def\SLF@lof{%
  \ifx\SLF@list\@empty\else\expandafter\SLF@explist\fi}
```

\SLF@contentsline
\section
\thestc
\tf@mtc
\slfname
\MTC@WriteContentsLine

La macro `\SLF@contentsline` analyse les lignes lues dans le fichier liste des figures. Si `\section` est trouvée, `stc` est incrémenté et un nouveau fichier sectlof est créé.

```
\def\SLF@contentsline#1#2#3#4{% %%HO: #4 added
  \gdef\thestc{\arabic{stc}}% %%HO: space removed
  \expandafter\ifx\csname #1\endcsname\xsect
    \stepcounter{stc}%
%%    \gdef\thestc{\arabic{stc}} %%HO: removed
```

\if@longextensions@
\slfname

Nous testons si les suffixes longs ou courts sont utilisés, pour construire le nom du fichier mini-table, puis nous l'ouvrons :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.slf\thestc}%
      \def\slfname{\jobname.slf\thestc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.H\thestc}%
      \def\slfname{\jobname.H\thestc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\slfname
  \fi
```

\mtc@toks
\MTC@WriteContentsline
\figure

Puis nous filtrons les lignes de contenu concernées :

```
  \mtc@toks{\noexpand\leavevmode #2}%
  \expandafter\ifx\csname #1\endcsname\figure
    \MTC@WriteContentsline{#1}{slf}{#3}{#4}%
  \fi
}
```

\SLF@explist
\SLF@next
\SLF@list

Voici la boucle de lecture des lignes du fichier liste des figures ; nous expansons la liste des entrées et appelons `\SLF@next` pour traiter la première.

```
\def\SLF@explist{\expandafter\SLF@next\SLF@list\\}
```

\SLF@loop \SLF@lof \SLF@read

Si une entrée est trouvée, on examine ligne par ligne, en recherchant les entrées « intéressantes ». Sinon, nous traitons l'entrée suivante dans la liste.

```
\def\SLF@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \PackageWarning{minitoc}%
       {No file #1\MessageBreak SECTLOFS NOT PREPARED}%
    \expandafter\SLF@lof
  \else
    \PackageInfo{minitoc}%
       {PREPARING SECTLOFS FROM #1}%
    \expandafter\SLF@read\fi}
```

Nous lisons l'entrée suivante dans le fichier `.lof`.

\SLF@read \SLF@test \SLF@line

Les `.....` garantissent que `\SLF@test` a suffisamment d'arguments :

```
\def\SLF@read{%
  \read\@inputcheck to\SLF@line
  \expandafter\SLF@test\SLF@line.....\SLF@% %%HO: . added
  }%
```

\SLF@test \SLF@contentsline \mtc@string \SLF@list \SLF@lof \SLF@read \sectend \sectbegin

La macro `\SLF@test` trouve les commandes « intéressantes » dans le fichier liste des figures, essentiellement pour délimiter les sections :

```
\long\def\SLF@test#1#2#3#4#5#6\SLF@{% %%HO: #6 added
  \ifx#1\contentsline
    \let\mtc@string\string
    \SLF@contentsline{#2}{#3}{#4}{#5}% %%HO: #4 added
    \let\mtc@string\relax
  \else\ifx#1\@input
     \edef\SLF@list{\SLF@list#2\relax}%
  \else\ifx#1\sectend
     \immediate\closeout\tf@mtc
     \immediate\openout\tf@mtc=\jobname.mtc
  \else\ifx#1\sectbegin
     \addtocounter{stc}{-1}%
  \fi\fi\fi\fi
  \ifeof\@inputcheck\expandafter\SLF@lof
  \else\expandafter\SLF@read\fi}%
```

\SLT@next \SLT@list \SLT@loop

Nous traitons l'entrée suivante dans la liste et la retirons de la tête de la liste :

```
\def\SLT@next#1\relax#2\\{%
  \edef\SLT@list{#2}%
  \SLT@loop{#1}}
```

\SLT@lot \SLT@list \SLT@explist

Nous testons si la liste est vide :

```
\def\SLT@lot{%
  \ifx\SLT@list\@empty\else\expandafter\SLT@explist\fi}
```

\SLT@contentsline \section \thestc \tf@mtc \sltname \MTC@WriteContentsLine

La macro `\SLT@contentsline` analyse les lignes lues dans le fichier liste des tableaux. Si `\section` est trouvée, `stc` est incrémenté et un nouveau fichier sectlot est créé.

```
\def\SLT@contentsline#1#2#3#4{% %%HO: #4 added
  \gdef\thestc{\arabic{stc}}% %%HO: space removed
  \expandafter\ifx\csname #1\endcsname\xsect
    \stepcounter{stc}%
%%    \gdef\thestc{\arabic{stc}} %%HO: removed
```

\if@longextensions@ \sltname

Nous testons si les suffixes longs ou courts sont utilisés, pour construire le nom du fichier mini-table, puis nous l'ouvrons :

```
    \if@longextensions@%
      \PackageInfo{minitoc}%
         {Writing\space\jobname.slt\thestc}%
      \def\sltname{\jobname.slt\thestc}%
    \else
      \PackageInfo{minitoc}%
         {Writing\space\jobname.V\thestc}%
      \def\sltname{\jobname.V\thestc}%
    \fi
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\sltname
  \fi
```

\mtc@toks \MTC@WriteContentsline \table

Puis nous filtrons les lignes de contenu concernées :

```
  \mtc@toks{\noexpand\leavevmode #2}%
  \expandafter\ifx\csname #1\endcsname\table
    \MTC@WriteContentsline{#1}{slt}{#3}{#4}%
  \fi
}
```

\SLT@explist \SLT@next \SLT@list

Voici la boucle de lecture des lignes du fichier liste des tableaux ; nous expansons la liste des entrées et appelons `\SLT@next` pour traiter la première.

```
\def\SLT@explist{\expandafter\SLT@next\SLT@list\\}
```

\SLT@loop \SLT@lot \SLT@read

Si une entrée est trouvée, on examine ligne par ligne, en recherchant les entrées « intéressantes ». Sinon, nous traitons l'entrée suivante dans la liste.

```
\def\SLT@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
```

```
  \PackageWarning{minitoc}%
     {No file #1\MessageBreak SECTLOTS NOT PREPARED}%
  \expandafter\SLT@lot
\else
  \PackageInfo{minitoc}%
     {PREPARING SECTLOTS FROM #1}%
  \expandafter\SLT@read\fi}
```

Nous lisons l'entrée suivante dans le fichier `.lot`.

\SLT@read \SLT@test \SLT@line

Les `.....` garantissent que `\SLT@test` a suffisamment d'arguments :

```
\def\SLT@read{%
 \read\@inputcheck to\SLT@line
 \expandafter\SLT@test\SLT@line.....\SLT@% %%HO: . added
 }%
```

\SLT@test \SLT@contentsline \mtc@string \SLT@list \SLT@lot \SLT@read \sectend \sectbegin

La macro `\SLT@test` trouve les commandes « intéressantes » dans le fichier liste des tableaux, essentiellement pour délimiter les sections :

```
\long\def\SLT@test#1#2#3#4#5#6\SLT@{% %%HO: #6 added
 \ifx#1\contentsline
   \let\mtc@string\string
   \SLT@contentsline{#2}{#3}{#4}{#5}% %%HO: #4 added
   \let\mtc@string\relax
 \else\ifx#1\@input
    \edef\SLT@list{\SLT@list#2\relax}%
 \else\ifx#1\sectend
    \immediate\closeout\tf@mtc
    \immediate\openout\tf@mtc=\jobname.mtc
 \else\ifx#1\sectbegin
    \addtocounter{stc}{-1}%
 \fi\fi\fi\fi
 \ifeof\@inputcheck\expandafter\SLT@lot
 \else\expandafter\SLT@read\fi}%
```

5.62 Fin des commandes du niveau section

Nous terminons la branche *sinon* du test `\@ifundefined{section}`, la branche *vraie* du test `\@ifundefined{chapter}` et ajoutons une branche *sinon* vide au test `\@ifundefined{chapter}` :

```
\fi% end of \if@mtc@section@def@
\fi% end of \if@mtc@chapter@undef@
```

5.63 Commandes \l@... nécessaires

\l@listof \l@starpart \l@starchapter \l@starsection \l@starsubsection \l@starsubsubsection \l@starparagraph \l@starsubparagraph

Quelques commandes \l@... (analogues à \l@section ou \l@paragraph) sont requises pour mettre en page certaines entrées dans les mini-tables, essentiellement pour les commandes de sectionnement étoilées :

```
\@ifundefined{section}{}{\let\l@listof\l@section}
\@ifundefined{chapter}{}{\let\l@listof\l@chapter}
\@ifundefined{part}{}{\let\l@starpart\l@part}
\@ifundefined{chapter}{}{\let\l@starchapter\l@chapter}
\@ifundefined{section}{}{\let\l@starsection\l@section}
\@ifundefined{subsection}{}{\let\l@starsubsection\l@subsection}
\@ifundefined{subsubsection}{}{\let\l@starsubsubsection\l@subsubsection}
\@ifundefined{paragraph}{}{\let\l@starparagraph\l@paragraph}
\@ifundefined{subparagraph}{}{\let\l@starsubparagraph\l@subparagraph}
```

5.64 Les filets horizontaux et leurs valeurs par défaut

\noptcrule \nomtcrule \nostcrule \ptcrule \mtcrule \stcrule \noplfrule \nomlfrule \noslfrule \plfrule \mlfrule \slfrule \nopltrule \nomltrule \nosltrule \pltrule \mltrule \sltrule

Nous définissons ici les diverses commandes pour activer ou inhiber les filets horizontaux dans les diverses sortes de mini-tables. Chacune de ces commandes est une définition indirecte du filet horizontal correspondant. Les filets sont des filets horizontaux épais de 0.4pt.

```
\def\noptcrule{\let\ptc@rule\relax}
\def\nomtcrule{\let\mtc@rule\relax}
\def\nostcrule{\let\stc@rule\relax}
\def\ptcrule{\def\ptc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\mtcrule{\def\mtc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\stcrule{\def\stc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\ptc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\mtc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\stc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
```

```
\def\noplfrule{\let\plf@rule\relax}
\def\nomlfrule{\let\mlf@rule\relax}
\def\noslfrule{\let\slf@rule\relax}
\def\plfrule{\def\plf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\mlfrule{\def\mlf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\slfrule{\def\slf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\plf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\mlf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\slf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
```

```
\def\nopltrule{\let\plt@rule\relax}
\def\nomltrule{\let\mlt@rule\relax}
\def\nosltrule{\let\slt@rule\relax}
\def\pltrule{\def\plt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
```

```
\def\mltrule{\def\mlt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\sltrule{\def\slt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\plt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\mlt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\slt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
```

5.65 Les commandes `\mtcset...`

Ces commandes[7] ont été introduites pour construire une meilleure interface utilisateur, et chacune d'elles remplace de nombreuses commandes utilisateur, offrant une syntaxe assez unifiée et logique.

5.65.1 Mots-clés pour les commandes `\mtcset...`

`\@namedef` `\@nameuse` Nous définissons quelques mots-clés communs pour les commandes `\mtcset...`. Un mot-clé est créé via le mécanisme `\@namedef` – `\@nameuse` de la manière suivante :

`\@namedef{mtc@`*famille*`@`*nom*`}{`*abréviation*`}`

où *famille* est le nom d'un groupe de mots-clés concernent une ou plusieurs macros `\mtcset...`, *nom* est le mot-clé que l'utilisateur donne comme argument à la macro `\mtcset...`, et *abréviation* est une chaîne utilisée pour construire le nom de la macro effectivement utilisée. Comme certaines macros `\mtcset...` ont plusieurs paramètres mots-clés, cette méthode peut réduire le nombre de macros au niveau de l'utilisateur, au prix de quelques familles de mots-clés.

`\@namedef` Nous définissons une famille (`typetable`) de mots-clés pour les types de mini-tables :

```
\@namedef{mtc@typetable@parttoc}{ptc}\def\mtc@typetable@parttoc{ptc}
\@namedef{mtc@typetable@partlot}{plt}\def\mtc@typetable@partlot{plf}
\@namedef{mtc@typetable@partlof}{plf}\def\mtc@typetable@partlof{plt}
\@namedef{mtc@typetable@minitoc}{mtc}\def\mtc@typetable@minitoc{mtc}
\@namedef{mtc@typetable@minilof}{mlf}\def\mtc@typetable@minilof{mlf}
\@namedef{mtc@typetable@minilot}{mlt}\def\mtc@typetable@minilot{mlt}
\@namedef{mtc@typetable@secttoc}{stc}\def\mtc@typetable@secttoc{stc}
\@namedef{mtc@typetable@sectlof}{slf}\def\mtc@typetable@sectlof{slf}
\@namedef{mtc@typetable@sectlot}{slt}\def\mtc@typetable@sectlot{slt}
```

`\@namedef` Puis une autre famille (`typetitle`) pour les titres des mini-tables :

```
\@namedef{mtc@typetitle@parttoc}{pti}\def\mtc@typetitle@parttoc{pti}
\@namedef{mtc@typetitle@partlof}{pti}\def\mtc@typetitle@partlof{pti}
\@namedef{mtc@typetitle@partlot}{pti}\def\mtc@typetitle@partlot{pti}
\@namedef{mtc@typetitle@minitoc}{mti}\def\mtc@typetitle@minitoc{mti}
```

[7] Le concept général des commandes `\mtcset...` a été proposé par Benjamin BAYART.

```
\@namedef{mtc@typetitle@minilof}{mti}\def\mtc@typetitle@minilof{mti}
\@namedef{mtc@typetitle@minilot}{mti}\def\mtc@typetitle@minilot{mti}
\@namedef{mtc@typetitle@secttoc}{sti}\def\mtc@typetitle@secttoc{sti}
\@namedef{mtc@typetitle@sectlof}{sti}\def\mtc@typetitle@sectlof{sti}
\@namedef{mtc@typetitle@sectlot}{sti}\def\mtc@typetitle@sectlot{sti}
```

`\@namedef` Nous définissons une famille (`YN`) de mots-clés pour reconnaître les mots-clés « `off` » et « `on` », avec leurs nombreux synonymes [8] et signifiant faux ou vrai :

```
\@namedef{mtc@YN@off}{no}\def\mtc@YN@off{no}
\@namedef{mtc@YN@OFF}{no}\def\mtc@YN@OFF{no}
\@namedef{mtc@YN@no}{no}\def\mtc@YN@no{no}
\@namedef{mtc@YN@NO}{no}\def\mtc@YN@NO{no}
\@namedef{mtc@YN@n}{no}\def\mtc@YN@n{no}
\@namedef{mtc@YN@N}{no}\def\mtc@YN@N{no}
\@namedef{mtc@YN@false}{no}\def\mtc@YN@false{no}
\@namedef{mtc@YN@FALSE}{no}\def\mtc@YN@FALSE{no}
\@namedef{mtc@YN@faux}{no}\def\mtc@YN@faux{no}
\@namedef{mtc@YN@FAUX}{no}\def\mtc@YN@FAUX{no}
\@namedef{mtc@YN@f}{no}\def\mtc@YN@f{no}
\@namedef{mtc@YN@F}{no}\def\mtc@YN@F{no}
\@namedef{mtc@YN@NON}{no}\def\mtc@YN@NON{no}
\@namedef{mtc@YN@non}{no}\def\mtc@YN@non{no}
\@namedef{mtc@YN@0}{no}\expandafter\def\csname mtc@YN@0\endcsname{no}
\@namedef{mtc@YN@on}{}\def\mtc@YN@on{}
\@namedef{mtc@YN@ON}{}\def\mtc@YN@ON{}
\@namedef{mtc@YN@yes}{}\def\mtc@YN@yes{}
\@namedef{mtc@YN@YES}{}\def\mtc@YN@YES{}
\@namedef{mtc@YN@y}{}\def\mtc@YN@y{}
\@namedef{mtc@YN@Y}{}\def\mtc@YN@Y{}
\@namedef{mtc@YN@true}{}\def\mtc@YN@true{}
\@namedef{mtc@YN@TRUE}{}\def\mtc@YN@TRUE{}
\@namedef{mtc@YN@t}{}\def\mtc@YN@t{}
\@namedef{mtc@YN@T}{}\def\mtc@YN@T{}
\@namedef{mtc@YN@vrai}{}\def\mtc@YN@vrai{}
\@namedef{mtc@YN@VRAI}{}\def\mtc@YN@VRAI{}
\@namedef{mtc@YN@v}{}\def\mtc@YN@v{}
\@namedef{mtc@YN@V}{}\def\mtc@YN@TRUEV
\@namedef{mtc@YN@OUI}{}\def\mtc@YN@OUI{}
\@namedef{mtc@YN@oui}{}\def\mtc@YN@oui{}
\@namedef{mtc@YN@O}{}\def\mtc@YN@O{}
\@namedef{mtc@YN@o}{}\def\mtc@YN@o{}
\@namedef{mtc@YN@1}{}\expandafter\def\csname mtc@YN@1\endcsname{}
```

[8] Ce cas (extrême) montre la facilité de création de synonymes pour des mots-clés fréquemment utilisés. Notez aussi que lorsqu'un mot-clé contient un caractère qui n'est pas une lettre, nous devons ruser en utilisant `\expandafter \csname ... \endcsname`.

5.65.2 La commande \mtcsetfont

\@namedef Nous définissons les mots-clés pour les niveaux de sectionnement (notez que part n'est pas membre de cette famille (sectlevel), car aucune ligne de contenu pour une partie ne peut apparaître dans une mini-table, la partie étant le plus haut niveau de sectionnement) ; « * » représente « tout niveau », et est utilisée pour établir la fonte globale par défaut pour un genre donné de mini-table.

```
\@namedef{mtc@sectlevel@chapter}{C}\def\mtc@sectlevel@chapter{C}
\@namedef{mtc@sectlevel@section}{S}\def\mtc@sectlevel@section{S}
\@namedef{mtc@sectlevel@subsection}{SS}\def\mtc@sectlevel@subsection{SS}
\@namedef{mtc@sectlevel@subsubsection}{SSS}\def\mtc@sectlevel@subsubsection{SSS}
\@namedef{mtc@sectlevel@paragraph}{P}\def\mtc@sectlevel@paragraph{P}
\@namedef{mtc@sectlevel@subparagraph}{SP}\def\mtc@sectlevel@subparagraph{SP}
\@namedef{mtc@sectlevel@*}{}\expandafter\def\csname mtc@sectlevel@*\endcsname{}
```

\mtcsetfont La commande \mtcsetfont a la syntaxe suivante :

\mtcsetfont{*mini-table*}{*nom-niveau*}{*commandes de fonte*}

Le type de *mini-table* est un mot-clé tel que minitoc, le *nom-niveau* est un niveau de sectionnement tel que subsection (sans contre-oblique). Les *commandes de fonte* sont une spécification de fonte, utilisant habituellement des commandes basiques de NFSS [28].

\if@mtc@setfont@ Tout d'abord, nous déclarons un indicateur, mis à vrai :

```
\newif\if@mtc@setfont@\@mtc@setfont@true
```

\mtcsetfont Puis, nous commençons la commande, qui a trois arguments :

```
\newcommand{\mtcsetfont}[3]{%
```

\mtc@mta@abbrev \if@mtc@setfont@ \@nameuse Les deux premiers arguments sont des mots-clés. Ils doivent être traduits en chaînes effectives. Nous traitons le premier argument qui est un mot-clé de la famille typetable. Le résultat est rangé dans \mtc@mta@abbrev. Exemple : si #1 est minitoc, nous obtenons mtc.

```
\def\mtc@mta@abbrev{X}
\@mtc@setfont@true
\expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
   \@mtc@setfont@false
   \def\mtc@mta@abbrev{X}
   \PackageError{minitoc}%
      {*** \string\mtcsetfont \space has a wrong first argument (#1).
       \MessageBreak
       It should be a mini-table type (parttoc...sectlot)}%
      {Correct the source code.\MessageBreak
```

```
      Type <return> and rerun LaTeX}
\else
   \edef\mtc@mta@abbrev{\@nameuse{mtc@typetable@#1}}
\fi
```

\mtc@level@abbrev
\if@mtc@setfont@
\@nameuse

Le deuxième argument, un mot-clé de la famille `sectlevel`, est traité de ma même manière et le résultat est rangé dans `\mtc@level@abbrev`. Exemple : si `#2` est `subsection`, nous obtenons SS.

```
\def\mtc@level@abbrev{X}
\expandafter\ifx\csname mtc@sectlevel@#2\endcsname\relax
   \@mtc@setfont@false
   \def\mtc@level@abbrev{X}
   \PackageError{minitoc}%
     {*** \string\mtcsetfont \space has a wrong second argument (#2).
      \MessageBreak
      It should be a sectionning level (part...subparagraph) or * }%
     {Correct the source code.\MessageBreak
      Type <return> and rerun LaTeX}
\else
   \edef\mtc@level@abbrev{\@nameuse{mtc@sectlevel@#2}}
\fi
```

\mtc@tmp@name
\mtc@mta@abbrev
\mtc@level@abbrev

Puis nous construisons la macro effective qui sera appliquée :

```
\def\mtc@tmp@name{\mtc@mta@abbrev\mtc@level@abbrev font}
```

Exemple : si `#1` est `minitoc` et `#2` est `subsection`, nous obtenins `mtcSSfont`, qui est le noms de la commande pour la fonte d'une entrée de sous-section dans une minitoc (il manque la contre-oblique, mais nous utiliserons une paire `\csname` ... `\endcsname` pour appliquer la commande construite).

\if@mtc@setfont@

Mais toutes les combinaisons ne sont pas légales (le niveau de l'entrée doit être plus bas que celui de la mini-table), donc nous devons tester :

```
\expandafter\ifx\csname #1\endcsname\minitoc
   \expandafter\ifx\csname #2\endcsname\part\@mtc@setfont@false\fi
   \expandafter\ifx\csname #2\endcsname\chapter\@mtc@setfont@false\fi
\fi
\expandafter\ifx\csname #1\endcsname\minilof
   \expandafter\ifx\csname #2\endcsname\part\@mtc@setfont@false\fi
   \expandafter\ifx\csname #2\endcsname\chapter\@mtc@setfont@false\fi
\fi
\expandafter\ifx\csname #1\endcsname\minilot
   \expandafter\ifx\csname #2\endcsname\part\@mtc@setfont@false\fi
   \expandafter\ifx\csname #2\endcsname\chapter\@mtc@setfont@false\fi
\fi
\expandafter\ifx\csname #1\endcsname\secttoc
   \expandafter\ifx\csname #2\endcsname\part\@mtc@setfont@false\fi
```

```
    \expandafter\ifx\csname #2\endcsname\chapter\@mtc@setfont@false\fi
    \expandafter\ifx\csname #2\endcsname\section\@mtc@setfont@false\fi
\fi
\expandafter\ifx\csname #1\endcsname\sectlof
    \expandafter\ifx\csname #2\endcsname\part\@mtc@setfont@false\fi
    \expandafter\ifx\csname #2\endcsname\chapter\@mtc@setfont@false\fi
    \expandafter\ifx\csname #2\endcsname\section\@mtc@setfont@false\fi
\fi
\expandafter\ifx\csname #1\endcsname\sectlot
    \expandafter\ifx\csname #2\endcsname\part\@mtc@setfont@false\fi
    \expandafter\ifx\csname #2\endcsname\chapter\@mtc@setfont@false\fi
    \expandafter\ifx\csname #2\endcsname\section\@mtc@setfont@false\fi
\fi
```

\if@mtc@setfont@
\mtc@tmp@name
\mtc@mta@abbrev
\mtc@level@abbrev
\mtc@toks

Si la combinaison est légale, nous l'appliquons, c'est-à-dire que nous redéfinissons la signification de la macro construite comme étant la séquence de commandes donnée en troisième argument de `\mtcsetfont` et nous notons cet événement dans le fichier `.log` (nous rangeons le troisième argument dans un registre *token* pour pouvoir l'écrire *verbatim*).

Si la combinaison n'est pas légale, un message d'erreur est affiché.

```
\if@mtc@setfont@
    \def\mtc@tmp@name{\mtc@mta@abbrev\mtc@level@abbrev font}
    \mtc@toks{#3}
    \PackageInfo{minitoc}%
       {\string\mtcsetfont\space redefines the macro
         \mtc@tmp@name \space as "\the\mtc@toks"}
    \expandafter\def\csname\mtc@tmp@name\endcsname{#3}
\else
    \PackageError{minitoc}%
       {\string\mtcsetfont\space has incompatible\MessageBreak
        first (#1) and second (#2) arguments}{}
\fi
}
```

5.65.3 La commande `\mtcsettitlefont`

\mtcsettitlefont

Cette commande est très similaire à la commande `\mtcsetfont`. Sa syntaxe est presque identique :

`\mtcsettitlefont{`*mini-table*`}{`*commandes de fonte*`}`

\if@mtc@settitlefont@

Le type de *mini-table* est un mot-clé tel que `minitoc`. Les *commandes de fonte* sont une spécification de fonte, utilisant habituellement des commandes basiques de NFSS [28]. La différence est l'absence du deuxième argument mot-clé, car les *commandes de fonte* seront appliquées au titre de chaque mini-table du genre donné.

Tout d'abord, nous déclarons un indicateur, mis à vrai :

```
\newif\if@mtc@settitlefont@\@mtc@settitlefont@true
```

\mtcsettitlefont

Et nous commençons la définition de la commande `\mtcsettitlefont`, qui a deux arguments :

```
\newcommand{\mtcsettitlefont}[2]{%
```

\mtc@mtatf@abbrev
\if@mtc@settitlefont@
\@nameuse

Nous traitons le premier argument, un mot-clé de la famille `typetitle`, puis le résultat est rangé dans `\mtc@mtatf@abbrev` :

```
\def\mtc@mtatf@abbrev{X}
\@mtc@settitlefont@true
\expandafter\ifx\csname mtc@typetitle@#1\endcsname\relax
   \@mtc@settitlefont@false
   \def\mtc@mtatf@abbrev{X}
   \PackageError{minitoc}%
      {*** \string\mtcsettitlefont \space has a wrong first argument (#1).
        \MessageBreak
       It should be a mini-table type (parttoc...sectlot)}%
      {Correct the source code.\MessageBreak
       Type <return> and rerun LaTeX}
\else
   \edef\mtc@mtatf@abbrev{\@nameuse{mtc@typetitle@#1}}
\fi
```

\if@mtc@settitlefont@
\mtc@tmptf@name
\mtc@mtatf@abbrev
\mtc@toks

Puis nous construisons le nom de la commande effective et appliquons cette commande :

```
\if@mtc@settitlefont@
   \def\mtc@tmptf@name{\mtc@mtatf@abbrev font}
   \mtc@toks{#2}
   \PackageInfo{minitoc}%
      {\string\mtcsettitlefont\space redefines the macro
       \mtc@tmptf@name \space as "\the\mtc@toks"}
   \expandafter\def\csname\mtc@tmptf@name\endcsname{#2}
\else
   \PackageError{minitoc}%
      {Illegal type of table (#1)}%
      {Correct the source code.\MessageBreak
       Type <return> and rerun LaTeX}{\relax}
\fi
}
```

5.65.4 La commande \mtcsettitle

\mtcsettitle

Cette commande est très similaire à la commande \mtcsettitlefont. Sa syntaxe est presque identique :

\mtcsettitle{*mini-table*}{*texte*}

Le type de *mini-table* est un mot-clé tel que minitoc. Le *texte* est le texte pour un titre de mini-table.

\if@mtc@settitle@

Tout d'abord, nous déclarons un indicateur, mis à vrai :

```
\newif\if@mtc@settitle@\@mtc@settitle@true
```

\mtcsettitle

Puis nous définissons la commande \mtcsettitle, qui a deux arguments :

```
\newcommand{\mtcsettitle}[2]{%
```

\mtc@mtati@abbrev
\if@mtc@settitle@
\@nameuse

Nous traitons le premier argument, qui est un mot-clé de la famille typetable. Le résultat est rangé dans \mtc@mtati@abbrev :

```
\def\mtc@mtati@abbrev{X}
\@mtc@settitle@true
\expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
   \@mtc@settitle@false
   \def\mtc@mtati@abbrev{X}
   \PackageError{minitoc}%
      {*** \string\mtcsettitle \space has a wrong first argument (#1).
       \MessageBreak
       It should be a mini-table type (parttoc...sectlot)}%
      {Correct the source code.\MessageBreak
       Type <return> and rerun LaTeX}
\else
   \edef\mtc@mtati@abbrev{\@nameuse{mtc@typetable@#1}}
\fi
```

\if@mtc@settitle@
\mtc@tmpti@name
\mtc@mtati@abbrev
\mtc@toks

Et nous construisons le nom de la macro effective et nous l'appliquons :

```
\if@mtc@settitle@
   \def\mtc@tmpti@name{\mtc@mtati@abbrev title}
   \mtc@toks{#2}
   \PackageInfo{minitoc}%
      {\string\mtcsettitle\space redefines the macro
       \mtc@tmpti@name \space as "\the\mtc@toks"}
   \expandafter\def\csname\mtc@tmpti@name\endcsname{#2}
\else
   \PackageError{minitoc}%
      {Illegal type of table (#1)}%
```

```
      {Correct the source code.\MessageBreak
       Type <return> and rerun LaTeX}{\relax}
\fi
}
```

5.65.5 La commande \mtcsetformat

\@namedef

Nous définissons d'abord les mots-clés (famille formatparam) pour les trois paramètres de mise en page que cette commande peut altérer :

```
\@namedef{mtc@formatparam@dotinterval}{dotsep}%
\def\mtc@arg@dotinterval{dotsep}
\@namedef{mtc@formatparam@tocrightmargin}{tocrmarg}%
\def\mtc@arg@tocrightmargin{tocrightmargin}
\@namedef{mtc@formatparam@pagenumwidth}{pnumwidth}%
\def\mtc@arg@pagenumwidth{\mtc@arg@pagenumwidth}
%% \@namedef{mtc@arg@numwidth}{numwidth} %not yet available
%% \def\mtc@arg@numwidth{\mtc@arg@numwidth} %not yet available
```

\AtBeginDocument

La commande \mtcsetformat a besoin qu'une initialisation soit faite en début de document pour établir les valeurs par défaut des paramètes de mise en page :

```
\AtBeginDocument{%
```

\@pnumwidth
\ptcpnumwidth
\mtcpnumwidth
\stcpnumwidth
\plfpnumwidth
\mlfpnumwidth
\slfpnumwidth
\pltpnumwidth
\mltpnumwidth
\sltpnumwidth

Nous prenons, si possible, la valeur par défaut de \@pnumwith pour chaque type de mini-tables :

```
\@ifundefined{ptcpnumwidth}{\let\ptcpnumwidth\@pnumwidth}{}%
\@ifundefined{stcpnumwidth}{\let\stcpnumwidth\@pnumwidth}{}%
\@ifundefined{mtcpnumwidth}{\let\mtcpnumwidth\@pnumwidth}{}%
\@ifundefined{plfpnumwidth}{\let\plfpnumwidth\@pnumwidth}{}%
\@ifundefined{mlfpnumwidth}{\let\mlfpnumwidth\@pnumwidth}{}%
\@ifundefined{slfpnumwidth}{\let\slfpnumwidth\@pnumwidth}{}%
\@ifundefined{pltpnumwidth}{\let\pltpnumwidth\@pnumwidth}{}%
\@ifundefined{mltpnumwidth}{\let\mltpnumwidth\@pnumwidth}{}%
\@ifundefined{sltpnumwidth}{\let\sltpnumwidth\@pnumwidth}{}%
```

\@tocrmarg \ptctocrmarg \mtclofrmarg \stclotrmarg \plftocrmarg \mlflofrmarg \slflotrmarg \plttocrmarg \mltlofrmarg \sltlotrmarg

Nous prenons, si possible, la valeur par défaut de `\@tocrmarg` pour chaque type de mini-tables :

```
\@ifundefined{ptctocrmarg}{\let\ptctocrmarg\@tocrmarg}{}%
\@ifundefined{mtctocrmarg}{\let\mtctocrmarg\@tocrmarg}{}%
\@ifundefined{stctocrmarg}{\let\stctocrmarg\@tocrmarg}{}%
\@ifundefined{plftocrmarg}{\let\plftocrmarg\@tocrmarg}{}%
\@ifundefined{mlftocrmarg}{\let\mlftocrmarg\@tocrmarg}{}%
\@ifundefined{slftocrmarg}{\let\slftocrmarg\@tocrmarg}{}%
\@ifundefined{plttocrmarg}{\let\plttocrmarg\@tocrmarg}{}%
\@ifundefined{mlttocrmarg}{\let\mlttocrmarg\@tocrmarg}{}%
\@ifundefined{slttocrmarg}{\let\slttocrmarg\@tocrmarg}{}%
```

\@dotsep \ptcdotsep \mtcdotsep \stcdotsep \plfdotsep \mlfdotsep \slfdotsep \pltdotsep \mltdotsep \sltdotsep

Nous prenons, si possible, la valeur par défaut de `\@dotsep` pour chaque type de mini-tables :

```
\@ifundefined{ptcdotsep}{\let\ptcdotsep\@dotsep}{}%
\@ifundefined{mtcdotsep}{\let\mtcdotsep\@dotsep}{}%
\@ifundefined{stcdotsep}{\let\stcdotsep\@dotsep}{}%
\@ifundefined{plfdotsep}{\let\plfdotsep\@dotsep}{}%
\@ifundefined{mlfdotsep}{\let\mlfdotsep\@dotsep}{}%
\@ifundefined{slfdotsep}{\let\slfdotsep\@dotsep}{}%
\@ifundefined{pltdotsep}{\let\pltdotsep\@dotsep}{}%
\@ifundefined{mltdotsep}{\let\mltdotsep\@dotsep}{}%
\@ifundefined{sltdotsep}{\let\sltdotsep\@dotsep}{}%
```

Et nous terminons le bloc `\AtBeginDocument` :

```
}
```

\mtcsetformat

La partie exécutive est effectuée via les macros suivantes, qui sont invoquées dans les environnements analogues à `\mtc@verse` pour chaque genre de mini-table. Ces commandes activent les valeurs enregisteées par `\mtcsetformat`.

\ptc@setform ptc@verse

La macro `\ptc@setform` est invoquée dans `ptc@verse` pour établir les paramètres de mise en page :

```
\def\ptc@setform{%
\let\@pnumwidth\ptcpnumwidth\relax
\let\@tocrmarg\ptctocrmarg\relax
\let\@dotsep\ptcdotsep\relax
}
```

\mtc@setform mtc@verse

La macro `\mtc@setform` est invoquée dans `mtc@verse` pour établir les paramètres de mise en page :

```
\def\mtc@setform{%
\let\@pnumwidth\mtcpnumwidth\relax
\let\@tocrmarg\mtctocrmarg\relax
\let\@dotsep\mtcdotsep\relax
}
```

\stc@setform stc@verse

La macro `\stc@setform` est invoquée dans `stc@verse` pour établir les paramètres de mise en page :

```
\def\stc@setform{%
\let\@pnumwidth\stcpnumwidth\relax
\let\@tocrmarg\stctocrmarg\relax
\let\@dotsep\stcdotsep\relax
}
```

\plf@setform ptc@verse

La macro `\plf@setform` est invoquée dans `ptc@verse` pour établir les paramètres de mise en page :

```
\def\plf@setform{%
\let\@pnumwidth\plfpnumwidth\relax
\let\@tocrmarg\plftocrmarg\relax
\let\@dotsep\plfdotsep\relax
}
```

\mlf@setform mtc@verse

La macro `\mlf@setform` est invoquée dans `mtc@verse` pour établir les paramètres de mise en page :

```
\def\mlf@setform{%
\let\@pnumwidth\mlfpnumwidth\relax
\let\@tocrmarg\mlftocrmarg\relax
\let\@dotsep\mlfdotsep\relax
}
```

\slf@setform stc@verse

La macro `\slf@setform` est invoquée dans `stc@verse` pour établir les paramètres de mise en page :

```
\def\slf@setform{%
\let\@pnumwidth\slfpnumwidth\relax
\let\@tocrmarg\slftocrmarg\relax
\let\@dotsep\slfdotsep\relax
}
```

\plt@setform ptc@verse

La macro `\plt@setform` est invoquée dans `ptc@verse` pour établir les paramètres de mise en page :

```
\def\plt@setform{%
\let\@pnumwidth\pltpnumwidth\relax
```

```
\let\@tocrmarg\plttocrmarg\relax
\let\@dotsep\pltdotsep\relax
}
```

\mlt@setform mtc@verse

La macro \mlt@setform est invoquée dans mtc@verse pour établir les paramètres de mise en page :

```
\def\mlt@setform{%
\let\@pnumwidth\plfpnumwidth\relax
\let\@tocrmarg\plftocrmarg\relax
\let\@dotsep\plfdotsep\relax
}
```

\slt@setform stc@verse

La macro \slt@setform est invoquée dans stc@verse pour établir les paramètres de mise en page :

```
\def\slt@setform{%
\let\@pnumwidth\plfpnumwidth\relax
\let\@tocrmarg\plftocrmarg\relax
\let\@dotsep\plfdotsep\relax
}
```

\if@mtc@setformat@ \mtcsetformat

Puis nous définissons un indicateur et la commande \mtcsetformat, qui a la syntaxe suivante :

\mtcsetformat{*mini-table*}{*nom-paramètre*}{*valeur*}

où *mini-table* est un mot-clé de la famille typetable, *nom-paramètre* est un mot-clé de la famille formatparam et *valeur*, la valeur de ce paramètre pour le genre de mini-table donné.

```
\newif\if@mtc@setformat@\@mtc@setformat@true
\newcommand{\mtcsetformat}[3]{%
```

\mtc@mtf@abbrev \mtc@fparam@abbrev

Puis nous traitons le premier argument et rangeons le résultat dans \mtc@mtf@abbrev :

```
\def\mtc@mtf@abbrev{X}
\def\mtc@fparam@abbrev{X}
\@mtc@setformat@true
\expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
  \@mtc@setformat@false
  \def\mtc@mtf@abbrev{X}
  \PackageError{minitoc}%
     {*** \string\mtcsetformat \space has a wrong first argument (#1).
     \MessageBreak
     It should be a mini-table type (parttoc...sectlot)}%
     {Correct the source code.\MessageBreak
      Type <return> and rerun LaTeX}
\else
  \edef\mtc@mtf@abbrev{\@nameuse{mtc@typetable@#1}}
\fi
```

\mtc@fparam@abbrev
\@nameuse

Puis le deuxième argument est traité et le résultat est rangé dans `\mtc@fparam@abbrev` :

```
\expandafter\ifx\csname mtc@formatparam@#2\endcsname\relax
 \@mtc@setformat@false
  \def\mtc@fparam@abbrev{X}
  \PackageError{minitoc}%
     {*** \string\mtcsetformat \space has a wrong second argument (#2).
      \MessageBreak
      It should be a formatting param
      (pagenumwidth, tocrightmargin, dotinterval)}%
     {Correct the source code.\MessageBreak
      Type <return> and rerun LaTeX}
\else
  \edef\mtc@fparam@abbrev{\@nameuse{mtc@formatparam@#2}}
\fi
```

\if@mtc@setformat@
\mtc@tmpfm@name
\mtc@mtf@abbrev
\mtc@fparam@abbrev

Le nom de la macro de stockage est construit et elle reçoit le troisième paramètre comme valeur :

```
\if@mtc@setformat@
  \def\mtc@tmpfm@name{\mtc@mtf@abbrev\mtc@fparam@abbrev}
  \PackageInfo{minitoc}%
     {\string\mtcsetformat\space redefines
      the macro \mtc@tmpfm@name \space as \string#3}
  \expandafter\def\csname\mtc@tmpfm@name\endcsname{#3}
\else
  \PackageError{minitoc}%
     {\string\mtcsetformat\space has incompatible
      \MessageBreak
      first (#1) and second (#2) arguments}{}
\fi
}
```

5.65.6 La commande `\mtcsetpagenumbers`

Cette commande active ou inhibe les numéros de pages dans les mini-tables du genre donné. Sa syntaxe est la suivante :

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

où *mini-table* est un mot-clé pour un genre de mini-table (`parttoc`, … `sectlot`) et `on` ou `off` un mot-clé pour activer (`on`) ou inhiber (`off`) les numéros de pages. `on` et `off` ont de nombreux synonymes.

\if@mtc@setpagenumbers@
\if@mtc@spn@ok@

Nous définissons quelques indicateurs :

```
\newif\if@mtc@setpagenumbers@ \@mtc@setpagenumbers@false
\newif\if@mtc@spn@ok@
```

\mtcsetpagenumber

Puis la macro \mtcsetpagenumbers, avec deux arguments :

```
\newcommand{\mtcsetpagenumbers}[2]{%
```

\mtc@mttpn@abbrev
\mtc@pnsw@abbrev

Nous traitons le premier argument, un mot-clé de la famille typetable, et rangeons le résultat dans \mtc@mttpn@abbrev :

```
\def\mtc@mttpn@abbrev{X}
\@mtc@setpagenumbers@true
\def\mtc@pnsw@abbrev{}
\expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
   \@mtc@setpagenumbers@false
   \def\mtc@pnsw@abbrev{X}
   \def\mtc@mttpn@abbrev{X}
   \PackageError{minitoc}%
      {*** \string\mtcsetpagenumbers \space has a wrong first argument (#1).}%
      {It should be a mini-table type (parttoc...sectlot)\MessageBreak
       Correct the source code, type <return> and rerun LaTeX}
\else
   \edef\mtc@mttpn@abbrev{\@nameuse{mtc@typetable@#1}}
\fi
```

\if@mtc@spn@ok@
\mtc@pnsw@abbrev
\mtc@mttpn@abbrev
\if@mtc@setpagenumbers@
\mtc@tmppn@name

Puis le second argument, un mot-clé de la famille YN, et nous rangeons le résultat dans \mtc@pnsw@abbrev. Le nom de la macro effective est construit puis la macro exécutée.

```
\@mtc@spn@ok@true
\expandafter\ifx\csname mtc@YN@#2\endcsname\relax
   \@mtc@spn@ok@false
   \def\mtc@pnsw@abbrev{X}
   \def\mtc@mttpn@abbrev{X}
   \@mtc@setpagenumbers@false
   \def\mtc@mttpn@abbrev{X}
   \PackageError{minitoc}%
      {*** \string\mtcsetpagenumbers \space has a wrong second argument (#2)}%
      {It should be a boolean value (0/1, yes/no, on/off\MessageBreak
       Correct the source code, type <return> and rerun LaTeX}
\else
   \edef\mtc@pnsw@abbrev{\@nameuse{mtc@YN@#2}}
   \def\mtc@pnsw@abbrevX{X}
   \def\mtc@noX{mtc@noX}
   \def\mtc@tmppn@name{\mtc@pnsw@abbrev\mtc@mttpn@abbrev pagenumbers}
   \expandafter\ifx\csname mtc@\mtc@pnsw@abbrev X\endcsname\mtc@noX
      \PackageInfo{minitoc}%
         {Page numbers are inhibited for the #1s}
   \else
      \PackageInfo{minitoc}%
         {Page numbers are activated for the #1s}
   \fi
   \csname\mtc@tmppn@name\endcsname{}
\fi
}
```

5.65.7 La commande `\mtcsetrules`

Cette macro ressemble beaucoup à `\mtcsetpagenumbers` et sa syntaxe est la même :

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

où *mini-table* est un mot-clé pour un genre de mini-table (`parttoc`, ... `sectlot`) et `on` ou `off` un mot-clé pour activer (`on`) ou inhiber (`off`) les filets horizontaux. `on` et `off` ont de nombreux synonymes.

Donc le code est similaire.

`\if@mtc@setrules@`
`\if@mtc@sru@ok@`

Nous définissons quelques indicateurs :

```
\newif\if@mtc@setrules@ \@mtc@setrules@false
\newif\if@mtc@sru@ok@
```

`\mtcsetrules`

Puis la macro `\mtcsetrules`, qui a deux arguments :

```
\newcommand{\mtcsetrules}[2]{%
```

`\mtc@mttru@abbrev`
`\if@mtc@setrules@`
`\mtc@rusw@abbrev`
`\@nameuse`

Nous traitons le premier argument, un mot-clé de la famille `typetable` puis rangeons le résultat dans `\mtc@mttru@abbrev` :

```
\def\mtc@mttru@abbrev{X}
\@mtc@setrules@true
\def\mtc@rusw@abbrev{}
\expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
   \@mtc@setrules@false
   \def\mtc@rusw@abbrev{X}
   \def\mtc@mttru@abbrev{X}
   \PackageError{minitoc}%
      {*** \string\mtcsetrules \space has a wrong first argument (#1).}%
      {It should be a mini-table type (parttoc...sectlot)
       \MessageBreak
       Correct the source code, type <return> and rerun LaTeX}
\else
   \edef\mtc@mttru@abbrev{\@nameuse{mtc@typetable@#1}}
\fi
```

`\if@mtc@sru@ok@`
`\mtc@rusw@abbrev`
`\mtc@mttru@abbrev`
`if@setrules@false`
`\mtc@noX`
`\mtc@tmppn@name`

Puis le second argument, un mot-clé de la famille `YN`, et nous rangeons le résultat dans `\mtc@rusw@abbrev`. Le nom de la macro effective est construit et cette macro est exécutée.

```
\@mtc@sru@ok@true
\expandafter\ifx\csname mtc@YN@#2\endcsname\relax
   \@mtc@sru@ok@false
   \def\mtc@rusw@abbrev{X}
   \def\mtc@mttru@abbrev{X}
```

```
   \@mtc@setrules@false
   \PackageError{minitoc}%
      {*** \string\mtcsetrules \space has a wrong second argument (#2)}%
      {It should be a boolean value (0/1, yes/no, on/off
       \MessageBreak
       Correct the source code, type <return> and rerun LaTeX}
\else
   \edef\mtc@rusw@abbrev{\@nameuse{mtc@YN@#2}}
   \def\mtc@rusw@abbrevX{X}
   \def\mtc@noX{mtc@noX}
   \def\mtc@tmppn@name{\mtc@rusw@abbrev\mtc@mttru@abbrev rule}
   \expandafter\ifx\csname mtc@\mtc@rusw@abbrev X\endcsname\mtc@noX
      \PackageInfo{minitoc}%
         {Horizontal rules are inhibited for the #1s}
   \else
      \PackageInfo{minitoc}%
         {Horizontal rules are activated for the #1s}
   \fi
   \csname\mtc@tmppn@name\endcsname{}
\fi
}
```

5.65.8 La commande \mtcsetfeature

Pour cette commande, nous devons définir trois familles de mots-clés, mais la troisième ne sert qu'à ajouter le mot « `style` » pour le « `pagestyle` » lorsque « `thispage` » est utilisé.

Une famille pour les noms longs des types de mini-tables :

```
\@namedef{mtc@ltypetable@parttoc}{parttoc}\def\mtc@ltypetable@parttoc{parttoc}
\@namedef{mtc@ltypetable@partlot}{partlot}\def\mtc@ltypetable@partlot{partlot}
\@namedef{mtc@ltypetable@partlof}{partlof}\def\mtc@ltypetable@partlof{partlof}
\@namedef{mtc@ltypetable@minitoc}{minitoc}\def\mtc@ltypetable@minitoc{minitoc}
\@namedef{mtc@ltypetable@minilot}{minilot}\def\mtc@ltypetable@minilot{minilot}
\@namedef{mtc@ltypetable@minilof}{minilof}\def\mtc@ltypetable@minilof{minilof}
\@namedef{mtc@ltypetable@secttoc}{secttoc}\def\mtc@ltypetable@secttoc{secttoc}
\@namedef{mtc@ltypetable@sectlof}{sectlof}\def\mtc@ltypetable@sectlof{sectlof}
\@namedef{mtc@ltypetable@sectlot}{sectlot}\def\mtc@ltypetable@sectlot{sectlot}
```

Une famille pour le type de dispositif :

```
\@namedef{mtc@featureparam@before}{before}%
   \def\mtc@featureparam@before{before}
\@namedef{mtc@featureparam@after}{after}%
   \def\mtc@featureparam@after{after}
\@namedef{mtc@featureparam@pagestyle}{thispage}%
   \def\mtc@featureparam@pagestyle{thispage}
```

Et une famille pour ajouter « style » si c'est un dispositif « pagestyle » :

```
\@namedef{mtc@ft3@before}{}\expandafter\def\csname mtc@ft3@before\endcsname{}
\@namedef{mtc@ft3@after}{}\expandafter\def\csname mtc@ft3@after\endcsname{}
\@namedef{mtc@ft3@pagestyle}{style}%
    \expandafter\def\csname mtc@ft3@pagestyle\endcsname{style}
```

La commande `\mtcsetfeature` a la syntaxe suivante :

`\mtcsetfeature{`*mini-table*`}{`*nom-dispositif*`}{`*commandes*`}`

où *mini-table* est une mot-clé de la famille `ltypetable` family, *nom-dispositif* est un mot-clé de la famille `featureparam` (mais aussi de la famille `ft3`), et *commandes* sont les commandes qui constituent le dispositif choisi.

`\if@mtc@setfeature@`
`\mtcsetfeature`

Nous définissons un indicateur et la commande `\mtcsetfeature`, qui a trois arguments :

```
\newif\if@mtc@setfeature@\@mtc@setfeature@true
\newcommand{\mtcsetfeature}[3]{%
```

`\mtc@mtfeat@abbrev`
`\mtc@featparam@abbrev`

Nous traitons le premier argument, un mot-clé de la famille `ltypetable` et rangeons le résultat dans `\mtc@mtfeat@abbrev` :

```
\def\mtc@mtfeat@abbrev{X}
\def\mtc@featparam@abbrev{X}
\@mtc@setfeature@true
\expandafter\ifx\csname mtc@ltypetable@#1\endcsname\relax
  \@mtc@setfeature@false
  \def\mtc@mtfeat@abbrev{X}
  \PackageError{minitoc}%
     {*** \string\mtcsetfeature \space has a wrong first argument (#1).
      \MessageBreak
      It should be a mini-table type (parttoc...sectlot)}%
     {Correct the source code.\MessageBreak
      Type <return> and rerun LaTeX}
\else
  \edef\mtc@mtfeat@abbrev{\@nameuse{mtc@ltypetable@#1}}
\fi
```

`\if@mtc@setfeature@`
`\mtc@featparam@abbrev`
`\mtc@featparam@third`
`\@nameuse`

Le deuxième argument est un mot-clé de la famille `featureparam`, le résultat est rangé dans `\mtc@featparam@`; et le complément est calculé d'aprés le premier argument, interprété comme un mot-clé de la famille `ft3` et dont le résultat rangé dans `\mtc@featparam@third`.

```
\expandafter\ifx\csname mtc@featureparam@#2\endcsname\relax
 \@mtc@setfeature@false
  \def\mtc@featparam@abbrev{X}
  \def\mtc@featparam@third{X}
  \PackageError{minitoc}%
     {*** \string\mtcsetfeature \space has a wrong second argument (#2).
```

```
      \MessageBreak
      It should be a feature param (before, after, pagestyle)}%
     {Correct the source code.\MessageBreak
      Type <return> and rerun LaTeX}
\else
  \edef\mtc@featparam@abbrev{\@nameuse{mtc@featureparam@#2}}
  \edef\mtc@featparam@third{\@nameuse{mtc@ft3@#2}}
\fi
```

\if@mtc@setfeature@
\mtc@tmpfeat@name
\mtc@featparam@abbrev
\mtc@mtfeat@abbrev
\mtc@featparam@third

Le nom de la macro effective est construit en concaténant ces trois morceaux (nommés `\mtc@featparam@abbrev`, `\mtc@mtfeat@abbrev` et `\mtc@featparam@third` respectivement), puis cette macro est exécutée :

```
\if@mtc@setfeature@
  \def\mtc@tmpfeat@name%
     {\mtc@featparam@abbrev\mtc@mtfeat@abbrev\mtc@featparam@third}
  \PackageInfo{minitoc}%
     {\string\mtcsetfeature\space redefines the macro
      "\csname mtc@tmpfeat@name\endcsname" as "\string#3"}
  \expandafter\def\csname\mtc@tmpfeat@name\endcsname{#3}
\else
  \PackageError{minitoc}%
     {\string\mtcsetfeature\space has incompatible\MessageBreak
      first (#1) and second (#2) arguments}{}
\fi}
```

5.65.9 La commande `\mtcsetdepth`

Cette commande est très similaire à la commande `\mtcsettitle`. Sa syntaxe est presque identique :

`\mtcsetdepth`{*mini-table*}{*profondeur*}

Le type de *mini-table* est un mot-clé tel que `minitoc`. La *profondeur* est la profondeur pour une mini-table. Si c'est une mini-table pour une liste de figures ou de tableaux, le compteur profondeur correspondant *doit être disponible*, c'est-à-dire qu'il doit avoir été créé (souvent par un paquetage adéquat, tel que le paquetage `subfig` [13]).

\if@mtc@setdepth@

Tout d'abord, nous déclarons un indicateur, vrai par défaut :

```
\newif\if@mtc@setdepth@\@mtc@setdepth@true
```

\mtcsetdepth

Puis nous définissons la commande `\mtcsetdepth`, avec deux arguments :

```
\newcommand{\mtcsetdepth}[2]{%
```

\mtc@mtade@abbrev \if@mtc@setdepth@ \@nameuse

Nous traitons le premier argument, qui est un mot-clé de la famille `ltypetable`. Le résultat est rangé dans `\mtc@mtade@abbrev` :

```
\def\mtc@mtade@abbrev{X}
\@mtc@setdepth@true
\expandafter\ifx\csname mtc@ltypetable@#1\endcsname\relax
   \@mtc@setdepth@false
   \def\mtc@mtade@abbrev{X}
   \PackageError{minitoc}%
      {*** \string\mtcsetdepth \space has a wrong first argument (#1).
       \MessageBreak
       It should be a mini-table type (parttoc...sectlot)}%
      {Correct the source code.\MessageBreak
       Type <return> and rerun LaTeX}
\else
   \edef\mtc@mtade@abbrev{\@nameuse{mtc@ltypetable@#1}}
\fi
```

\if@mtc@setdepth@ \mtc@tmpde@name \mtc@mtade@abbrev \mtc@toks \setcounter

Et nous construisons le nom du compteur effectif et lui donnons la valeur :

```
\if@mtc@setdepth@
   \def\mtc@tmpde@name{\mtc@mtade@abbrev depth}
   \expandafter\noexpand\@ifundefined{c@\mtc@mtade@abbrev depth}%
      {\PackageError{minitoc}%
       {***  \string\mtcsetdepth \space attempt to use\MessageBreak
             an undefined counter (#1depth).}%
      {Correct the source code.\MessageBreak
       Type <return> and rerun LaTeX}}{%
   \mtc@toks{#2} % trick for explicit message using \the.
   \PackageInfo{minitoc}%
      {\string\mtcsetdepth\space redefines the counter
       \mtc@tmpde@name \space as "\the\mtc@toks"}%
    \expandafter\setcounter{\mtc@tmpde@name}{#2}}%
\else
   \PackageError{minitoc}%
      {\string\mtcsetdepth:\space Illegal type of table (#1)}%
      {Correct the source code.\MessageBreak
       Type <return> and rerun LaTeX}{\relax}
\fi}% end of \mtcsettitle
```

5.66 L'environnement mtchideinmaintoc

\if@mtc@Himtoc@

L'indicateur `\if@mtc@Himtoc@` est utilisé pour détecter une imbrication incorrecte de cet environnement :

```
\newif\if@mtc@Himtoc@ \@mtc@Himtoc@false
```

mtchideinmaintoc \if@mtc@Himtoc@ \mtc@sv@tocdepth \arabic \addtocontents \setcounter

Puis nous définissons cet environnement, qui écrit dans le fichier table des matières des commandes de la forme `\setcounter{tocdepth}{...}`. Il sauvegarde et restaure aussi la valeur du compteur `tocdepth`, car l'argument optionnel est la profondeur de masquage des entrées dans la table des matières principale.

```
\newenvironment{mtchideinmaintoc}[1][-1]%
{\if@mtc@Himtoc@\PackageError{minitoc}%
    {Imbrication of mtchideinmaintoc environments}%
    {The hiding in main ToC could be incorrect}\fi
 \global\@mtc@Himtoc@true
 \def\mtc@sv@tocdepth{\arabic{tocdepth}}%
 \addtocontents{toc}{\protect\setcounter{tocdepth}{#1}}}%
{\if@mtc@Himtoc@\else\PackageError{minitoc}%
    {Unbalanced mtchideinmaintoc environment}%
    {The hiding in main ToC could be incorrect}\fi
 \global\@mtc@Himtoc@false
 \addtocontents{toc}{\protect\setcounter{tocdepth}{\mtc@sv@tocdepth}}}
```

\AtBeginDocument \if@mtc@Himlof@ mtchideinmainlof \mtc@sv@lofdepth \mtc@sv@tocdepth \mtc@svf@tocdepth \arabic \addtocontents \setcounter

Les environnements `mtchideinmainlof` et `mtchideinmainlot` sont similaires, mais nous devons véfifier la présence du compteur de profondeur associé, donc nous avons deux versions de ces environnements. Ceci doit être fait *après* le chargement des paquetages.

D'abord, pour la liste des figures :

```
\newif\if@mtc@Himlof@ \@mtc@Himlof@false
\AtBeginDocument{%
\@ifundefined{c@lofdepth}{%
   \newenvironment{mtchideinmainlof}[1][-1]%
   {\if@mtc@Himlof@\PackageError{minitoc}%
       {Imbrication of mtchideinmainlof environments}%
       {The hiding in main LoF could be incorrect}\fi
    \global\@mtc@Himlof@true
    \def\mtc@sv@tocdepth{\arabic{tocdepth}}%
    \def\mtc@sv@lofdepth{\arabic{tocdepth}}%
    \addtocontents{lof}{\protect\setcounter{tocdepth}{#1}}}%
   {\if@mtc@Himlof@\else\PackageError{minitoc}%
       {Unbalanced mtchideinmainlof environment}%
       {The hiding in main LoF could be incorrect}\fi
    \global\@mtc@Himtoc@false
    \addtocontents{lof}{\protect\setcounter{tocdepth}{\mtc@sv@lofdepth}}}
}%
{%
   \newenvironment{mtchideinmainlof}[1][-1]%
   {\if@mtc@Himlof@\PackageError{minitoc}%
       {Imbrication of mtchideinmainlof environments}%
       {The hiding in main LoF could be incorrect}\fi
    \global\@mtc@Himlof@true
    \def\mtc@svf@tocdepth{\arabic{tocdepth}}%
    \addtocontents{lof}{\protect\setcounter{tocdepth}{#1}}}%
   {\if@mtc@Himlof@\else\PackageError{minitoc}%
       {Unbalanced mtchideinmainlof environment}%
```

```
      {The hiding in main LoF could be incorrect}\fi
    \global\@mtc@Himlof@false
    \addtocontents{lof}{\protect\setcounter{tocdepth}{\mtc@svf@tocdepth}}}}
```

\AtBeginDocument \if@mtc@Himlot@ mtchideinmainlot \mtc@sv@lotdepth \mtc@sv@tocdepth \mtc@svt@tocdepth \arabic \addtocontents \setcounter

Puis pour la liste des tableaux :

```
\newif\if@mtc@Himlot@ \@mtc@Himlot@false
\AtBeginDocument{%
\@ifundefined{c@lotdepth}{%
   \newenvironment{mtchideinmainlot}[1][-1]%
   {\if@mtc@Himlot@\PackageError{minitoc}%
      {Imbrication of mtchideinmainlot environments}%
      {The hiding in main LoT could be incorrect}\fi
    \global\@mtc@Himlot@true
    \def\mtc@sv@tocdepth{\arabic{tocdepth}}%
    \def\mtc@sv@lotdepth{\arabic{tocdepth}}%
    \addtocontents{lot}{\protect\setcounter{tocdepth}{#1}}}%
   {\if@mtc@Himlot@\else\PackageError{minitoc}%
      {Unbalanced mtchideinmainlot environment}%
      {The hiding in main LoT could be incorrect}\fi
    \global\@mtc@Himlot@false
    \addtocontents{lot}{\protect\setcounter{tocdepth}{\mtc@sv@lotdepth}}}
}%
{%
   \newenvironment{mtchideinmainlot}[1][-1]%
   {\if@mtc@Himlot@\PackageError{minitoc}%
      {Imbrication of mtchideinmainlot environments}%
      {The hiding in main LoT could be incorrect}\fi
    \global\@mtc@Himlot@true
    \def\mtc@svt@tocdepth{\arabic{tocdepth}}%
    \addtocontents{lot}{\protect\setcounter{tocdepth}{#1}}}%
   {\if@mtc@Himlot@\else\PackageError{minitoc}%
      {Unbalanced mtchideinmainlot environment}%
      {The hiding in main LoT could be incorrect}\fi
    \global\@mtc@Himlot@false
    \addtocontents{lot}{\protect\setcounter{tocdepth}{\mtc@svt@tocdepth}}}}}
```

5.67 Correction de l'entrée « Index » dans la table des matières

Cette macro est complexe. Sa syntaxe est la suivante :

`\mtcfixindex[part|chapter|section]`

\@ifundefined \mtc@ixfix@level

Selon la classe du document, l'entrée « Index » dans la table des matières est traitée comme un chapitre étoilé ou une section étoilée. Donc nous devons d'abord déterminer la valeur par défaut de l'argument optionnel. La valeur par défaut est alors rangée dans la macro \mtc@ixfix@level. Ceci est fait par le code suivant, qui produit éventuellement un message d'avertissement :

```
\@ifundefined{chapter}{%
   \@ifundefined{section}%
      {\PackageWarningNoLine{minitoc}%
       {\string\chapter\space and \string\section\space are undefined.%
        \MessageBreak Cannot use \string\mtcfixindex\space %
           without optional argument [part]}%
       \@ifundefined{part}%
         {\PackageError{minitoc}%
            {But \string\part\space is undefined}%
            {\string\mtcfixindex\space not usable}}%
       {\PackageWarningNoLine{minitoc}%
          {\string\mtcfixindex\space can only be used with the [part]
           optional argument, which becomes the default}%
        \def\mtc@ixfix@level{part}%
       }%
       {\def\mtc@ixfix@level{section}}}}%
{\def\mtc@ixfix@level{chapter}}
```

\if@mtcfixindex@ \mtcfixindex \addcontentsline

Puis nous définissons un indicateur et la commande \mtcfixindex, qui ajoute les lignes nécessaires dans la table des matières, la liste des figures et la liste des tableaux.

```
\newif\if@mtcfixindex@ \@mtcfixindex@false
\newcommand{\mtcfixindex}[1][\mtc@ixfix@level]{%
   \@mtcfixindex@false
   \expandafter%
    \ifx\csname #1\endcsname\part\relax\@mtcfixindex@true\fi
   \expandafter%
    \ifx\csname #1\endcsname\chapter\relax\@mtcfixindex@true\fi
   \expandafter%
    \ifx\csname #1\endcsname\section\relax\@mtcfixindex@true\fi
   \if@mtcfixindex@
   \addcontentsline{lof}{x\mtc@ixfix@level}{}%
   \addcontentsline{lot}{x\mtc@ixfix@level}{}%
   \csname mtcadd\mtc@ixfix@level\endcsname\relax
   \else
   \PackageError{minitoc}%
      {The optional argument of \string\mtcfixindex\space is wrong}%
      {It must be omitted (\mtc@ixfix@level),
       or be part, chapter or section}%
   \fi
}%
```

5.68 Correction de l'entrée « Glossaire » dans la table des matières

Cette macro est complexe. Sa syntaxe est la suivante :

`\mtcfixglossary[part|chapter|section]`

\@ifundefined
\mtc@glofix@level

Selon la classe du document, l'entrée « Glossaire » dans la table des matières est traitée comme un chapitre étoilé ou une section étoilée. Donc nous devons d'abord déterminer la valeur par défaut de l'argument optionnel. La valeur par défaut est alors rangée dans la macro `\mtc@glofix@level`. Ceci est fait par le code suivant, qui produit éventuellement un message d'avertissement :

```
\@ifundefined{chapter}{%
   \@ifundefined{section}%
      {\PackageWarningNoLine{minitoc}%
       {\string\chapter\space and \string\section\space are undefined.%
        \MessageBreak Cannot use \string\mtcfixglossary\space %
           without optional argument [part]}%
       \@ifundefined{part}%
         {\PackageError{minitoc}%
            {But \string\part\space is undefined}%
            {\string\mtcfixglossary\space not usable}}%
       {\PackageWarningNoLine{minitoc}%
          {\string\mtcfixglossary\space can only be used with the [part]
           optional argument, which becomes the default}%
        \def\mtc@glofix@level{part}%
       }%
       {\def\mtc@glofix@level{section}}}}%
{\def\mtc@glofix@level{chapter}}
```

\if@mtcfixglossary@
\mtcfixglossary
\addcontentsline

Puis nous définissons un indicateur et la commande `\mtcfixglossary`, qui ajoute les lignes nécessaires dans la table des matières, la liste des figures et la liste des tableaux.

```
\newif\if@mtcfixglossary@ \@mtcfixglossary@false
\newcommand{\mtcfixglossary}[1][\mtc@glofix@level]{%
   \@mtcfixglossary@false
   \expandafter%
    \ifx\csname #1\endcsname\part\relax\@mtcfixglossary@true\fi
   \expandafter%
    \ifx\csname #1\endcsname\chapter\relax\@mtcfixglossary@true\fi
   \expandafter%
    \ifx\csname #1\endcsname\section\relax\@mtcfixglossary@true\fi
   \if@mtcfixglossary@
   \addcontentsline{lof}{x\mtc@glofix@level}{}%
   \addcontentsline{lot}{x\mtc@glofix@level}{}%
   \csname mtcadd\mtc@glofix@level\endcsname\relax
   \else
   \PackageError{minitoc}%
      {The optional argument of \string\mtcfixglossary\space is wrong}%
```

```
    {It must be omitted (\mtc@glofix@level),
     or be part, chapter or section}%
  \fi
}%
```

5.69 La commande \mtcselectlanguage

\mtcselectlanguage
\if@mtc@insellang@
\IfFileExists
\@input

Cette commande charge un fichier minitoc de définition de langue *langue*`.mld` pour établir les titres dépendants de la langue pour les mini-tables. Mais au préalable, nous vérifions que ce fichier existe. L'indicateur `\if@mtc@insellang@` est vrai tant que nous smmes dans cette macro.

```
\newif\if@mtc@insellang@ \@mtc@insellang@false
\def\mtcselectlanguage#1{%
    \InputIfFileExists{#1.mld}%
      {\PackageInfo{minitoc}{#1 language selected.%
        \MessageBreak}}%
      {\PackageError{minitoc}%
         {#1 is not a known language, \MessageBreak
          #1.mld not found. \MessageBreak
          Command ignored}%
         {See the minitoc documentation. \MessageBreak
          Correct the source using a valid language name. \MessageBreak
          Press RETURN}}%
}
```

5.70 La commande interne \mtcloadmlo

\mtcloadmlo
\if@mtc@insellang@
\IfFileExists
\@input

Cette commande charge un fichier minitoc objet de langue *langue*`.mlo` pour établir les titres dépendants de la langue pour les mini-tables lorsque des caractères exotiques sont nécessaires. Cette commande n'est utilisée que dans certains fichiers `.mld`, lorsque les chaînes de titres ne peuvent pas être produites par le traitement normal de `minitoc.dtx`. Les fichiers `.mlo` sont produits par des environnements `filecontents` dans le fichier `minitoc.ins`. Mais au préalable, nous vérifions que ce fichier `.mlo` existe.

Cette commande ne devrait pas être invoquée directement par l'utilisateur. Ceci est vérifié grâce à l'indicateur `\if@mtc@insellang@`.

```
\def\mtcloadmlo#1{%
    \if@mtc@insellang@
    \InputIfFileExists{#1.mlo}%
      {\PackageInfo{minitoc}{#1 minitoc language object selected.%
        \MessageBreak}}%
      {\PackageError{minitoc}%
```

```
        {#1 is not a known minitoc language object file (.mlo), \MessageBreak
         #1.mlo not found. \MessageBreak
         Command ignored}%
        {See the minitoc documentation. \MessageBreak
         Correct the source using a valid language name. \MessageBreak
         Press RETURN}}%
  \else
    \PackageError{minitoc}%
        {You are using the \string\mtcloadmlo\space command\MessageBreak
         outside of a .mld file}%
        {It will be ignored}
    \@mtc@insellang@false
  \fi
}
```

5.71 Les « pauses café »

\addcoffeeline
\addtocontents
\coffeeline
\l@coffee
\@Undottedtocline

Pour le paquetage `minutes` [30] (de Knut Lickert), nous avons besoin de quelques commandes pour insérer des entrées spéciales, sans points de conduite, dans la table des matières pour marquer les « pauses café » dans une conférence. Donc nous définissons `\addcoffeeline`, `\coffeeline` et `\l@coffee`, et des commandes internes analogues aux commandes internes standard pour mettre en forme la table des matières.

```
\def\addcoffeeline#1#2#3{%
  \addtocontents{#1}{\protect\coffeeline{#2}{#3}{\null}}}
\def\coffeeline#1{\csname l@#1\endcsname}
\newcommand*\l@coffee{\@Undottedtocline{1}{1.5em}{2.3em}}
```

5.72 Initialisation des compteurs

\AtBeginDocument
\@ifundefined
\setcounter

Au début du document, nous initialisons les compteurs absolus pour les parties, chapitres et sections, s'ils sont définis.

```
\AtBeginDocument{%
\@ifundefined{c@ptc}{}{\setcounter{ptc}{0}}
\@ifundefined{c@mtc}{}{\setcounter{mtc}{0}}
\@ifundefined{c@stc}{}{\setcounter{stc}{0}}}
```

5.73 Déclarations pour les options simples

Ces options ne font que positionner un indicateur.

5.73.1 Options tight et loose, k-tight et k-loose

\DeclareOption
\iftightmtc
\kiftightmtc

Ces options influencent l'interlignage dans les mini-tables.

```
\DeclareOption{tight}{\tightmtctrue}
\DeclareOption{loose}{\tightmtcfalse} % default
\DeclareOption{k-tight}{\ktightmtctrue}
\DeclareOption{k-loose}{\ktightmtcfalse} % default
```

5.73.2 Options checkfiles et nocheckfiles

\DeclareOption
\if@mtc@checkfiles

Ces options activent ou inhibent les tests de vacuité sur les fichiers des mini-tables.

```
\DeclareOption{checkfiles}{\@mtc@checkfilestrue} % default
\DeclareOption{nocheckfiles}{\@mtc@checkfilesfalse}
```

5.73.3 Options dotted et undotted

\DeclareOption
\ifundottedmtc

Ces options activent ou inhibent les points de conduite dans les mini-tables.

```
\DeclareOption{undotted}{\undottedmtctrue}
\DeclareOption{dotted}{\undottedmtcfalse} % default
```

5.73.4 Option notoccite

Cette option chargera plus loin le paquetage notoccite [2].

\DeclareOption
\if@mtc@notoccite@

```
\DeclareOption{notoccite}{\@mtc@notoccite@true}
```

5.73.5 Option shortext

\DeclareOption
\if@longextensions@

Cette option force l'utilisation de suffixes courts.

```
\DeclareOption{shortext}{\@longextensions@false
\PackageWarningNoLine{minitoc}%
   {*** You have forced the use of short extensions ***}}
```

5.74 L'option insection

\if@mtc@ss@insection@
\@ifundefined
\DeclareOption

Cette option n'est disponible que si `\chapter` n'est pas définie et `\section` est définie. Il faudra la revoir si un jour les commandes aux niveaux chapitre et section sont éventuellement permises ensemble, peut-être dans un avenir lointain.

```
\newif\if@mtc@ss@insection@ \@mtc@ss@insection@false
\@ifundefined{chapter}{%
   \@ifundefined{section}{}%
                         {%
                          \DeclareOption{insection}%
                           {\@mtc@ss@insection@true}%
                         }%
}{}
```

5.75 Options pour les langues

\DeclareOption
\mtcselectlanguage

Chaque option de langue lit le fichier *langue*`.mld` correspondant via la macro spécialisée `\mtcselectlanguage` macro. Si le fichier n'existe pas, un message d'erreur standard est affiché. Les options de langue sont (devraient) être par ordre alphabétique. Plusieurs options pourraient charger le même fichier, mais, par convention, il devrait y avoir un fichier *langue*`.mld` pour chaque option de langue, mais ce fichier peut en charger un autre (ainsi `francais.mld` charge `french.mld`).

```
\DeclareOption{acadian}{\mtcselectlanguage{acadian}}%
\DeclareOption{acadien}{\mtcselectlanguage{acadien}}%
\DeclareOption{afrikaan}{\mtcselectlanguage{afrikaan}}%
\DeclareOption{afrikaans}{\mtcselectlanguage{afrikaans}}%
\DeclareOption{american}{\mtcselectlanguage{american}}%
\DeclareOption{arab}{\mtcselectlanguage{arab}}%
\DeclareOption{arabic}{\mtcselectlanguage{arabic}}%
\DeclareOption{armenian}{\mtcselectlanguage{armenian}}%
\DeclareOption{austrian}{\mtcselectlanguage{austrian}}%
\DeclareOption{bahasa}{\mtcselectlanguage{bahasa}}%
\DeclareOption{bangla}{\mtcselectlanguage{bangla}}%
\DeclareOption{basque}{\mtcselectlanguage{basque}}%
\DeclareOption{bicig}{\mtcselectlanguage{bicig}}%
\DeclareOption{brazil}{\mtcselectlanguage{brazil}}%
\DeclareOption{brazilian}{\mtcselectlanguage{brazilian}}%
\DeclareOption{breton}{\mtcselectlanguage{breton}}%
\DeclareOption{british}{\mtcselectlanguage{british}}%
\DeclareOption{bulgarian}{\mtcselectlanguage{bulgarian}}%
\DeclareOption{bulgarianb}{\mtcselectlanguage{bulgarianb}}%
\DeclareOption{buryat}{\mtcselectlanguage{buryat}}%
\DeclareOption{canadian}{\mtcselectlanguage{canadian}}%
\DeclareOption{canadien}{\mtcselectlanguage{canadien}}%
\DeclareOption{castillan}{\mtcselectlanguage{castillan}}%
\DeclareOption{castillian}{\mtcselectlanguage{castillian}}%
```

```
\DeclareOption{catalan}{\mtcselectlanguage{catalan}}%
\DeclareOption{chinese1}{\mtcselectlanguage{chinese1}}%
\DeclareOption{chinese2}{\mtcselectlanguage{chinese2}}%
\DeclareOption{croatian}{\mtcselectlanguage{croatian}}%
\DeclareOption{czech}{\mtcselectlanguage{czech}}%
\DeclareOption{danish}{\mtcselectlanguage{danish}}%
\DeclareOption{dutch}{\mtcselectlanguage{dutch}}%
\DeclareOption{english}{\mtcselectlanguage{english}}%
\DeclareOption{esperant}{\mtcselectlanguage{esperant}}%
\DeclareOption{esperanto}{\mtcselectlanguage{esperanto}}%
\DeclareOption{estonian}{\mtcselectlanguage{estonian}}%
\DeclareOption{ethiopia}{\mtcselectlanguage{ethiopia}}%
\DeclareOption{ethiopian}{\mtcselectlanguage{ethiopian}}%
\DeclareOption{farsi1}{\mtcselectlanguage{farsi1}}%
\DeclareOption{farsi2}{\mtcselectlanguage{farsi2}}%
\DeclareOption{finnish}{\mtcselectlanguage{finnish}}%
\DeclareOption{finnish2}{\mtcselectlanguage{finnish2}}%
\DeclareOption{francais}{\mtcselectlanguage{francais}}%
\DeclareOption{french}{\mtcselectlanguage{french}}%
\DeclareOption{frenchb}{\mtcselectlanguage{frenchb}}%
\DeclareOption{frenchle}{\mtcselectlanguage{frenchle}}%
\DeclareOption{frenchpro}{\mtcselectlanguage{frenchpro}}%
\DeclareOption{galician}{\mtcselectlanguage{galician}}%
\DeclareOption{german}{\mtcselectlanguage{german}}%
\DeclareOption{germanb}{\mtcselectlanguage{germanb}}%
\DeclareOption{greek}{\mtcselectlanguage{greek}}%
\DeclareOption{greek-mono}{\mtcselectlanguage{greek-mono}}%
\DeclareOption{greek-polydemo}{\mtcselectlanguage{greek-polydemo}}%
\DeclareOption{greek-polykatha}{\mtcselectlanguage{greek-polykatha}}%
\DeclareOption{guarani}{\mtcselectlanguage{guarani}}%
\DeclareOption{hangul1}{\mtcselectlanguage{hangul1}}%
\DeclareOption{hangul2}{\mtcselectlanguage{hangul2}}%
\DeclareOption{hangul3}{\mtcselectlanguage{hangul3}}%
\DeclareOption{hangul4}{\mtcselectlanguage{hangul4}}%
\DeclareOption{hanja1}{\mtcselectlanguage{hanja1}}%
\DeclareOption{hanja2}{\mtcselectlanguage{hanja2}}%
\DeclareOption{hebrew}{\mtcselectlanguage{hebrew}}%
\DeclareOption{hungarian}{\mtcselectlanguage{hungarian}}%
\DeclareOption{icelandic}{\mtcselectlanguage{icelandic}}%
\DeclareOption{interlingua}{\mtcselectlanguage{interlingua}}%
\DeclareOption{irish}{\mtcselectlanguage{irish}}%
\DeclareOption{italian}{\mtcselectlanguage{italian}}%
\DeclareOption{japanese}{\mtcselectlanguage{japanese}}%
\DeclareOption{japanese2}{\mtcselectlanguage{japanese2}}%
\DeclareOption{japanese3}{\mtcselectlanguage{japanese3}}%
\DeclareOption{japanese4}{\mtcselectlanguage{japanese4}}%
\DeclareOption{japanese5}{\mtcselectlanguage{japanese5}}%
\DeclareOption{latin}{\mtcselectlanguage{latin}}%
\DeclareOption{latin2}{\mtcselectlanguage{latin2}}%
\DeclareOption{latvian}{\mtcselectlanguage{latvian}}%
\DeclareOption{letton}{\mtcselectlanguage{letton}}%
\DeclareOption{lithuanian}{\mtcselectlanguage{lithuanian}}%
\DeclareOption{lsorbian}{\mtcselectlanguage{lsorbian}}%
\DeclareOption{magyar}{\mtcselectlanguage{magyar}}%
```

```
\DeclareOption{magyar2}{\mtcselectlanguage{magyar2}}%
\DeclareOption{malayalam-keli}{\mtcselectlanguage{malayalam-keli}}%
\DeclareOption{malayalam-rachana}{\mtcselectlanguage{malayalam-rachana}}%
\DeclareOption{malayalam-rachana2}{\mtcselectlanguage{malayalam-rachana2}}%
\DeclareOption{mongol}{\mtcselectlanguage{mongol}}%
\DeclareOption{naustrian}{\mtcselectlanguage{naustrian}}%
\DeclareOption{ngerman}{\mtcselectlanguage{ngerman}}%
\DeclareOption{ngermanb}{\mtcselectlanguage{ngermanb}}%
\DeclareOption{norsk}{\mtcselectlanguage{norsk}}%
\DeclareOption{nynorsk}{\mtcselectlanguage{nynorsk}}%
\DeclareOption{polish}{\mtcselectlanguage{polish}}%
\DeclareOption{polish2}{\mtcselectlanguage{polish2}}%
\DeclareOption{portuges}{\mtcselectlanguage{portuges}}%
\DeclareOption{portuguese}{\mtcselectlanguage{portuguese}}%
\DeclareOption{romanian}{\mtcselectlanguage{romanian}}%
\DeclareOption{russian}{\mtcselectlanguage{russian}}%
\DeclareOption{russianb}{\mtcselectlanguage{russianb}}%
\DeclareOption{russianc}{\mtcselectlanguage{russianc}}%
\DeclareOption{russian2m}{\mtcselectlanguage{russian2m}}%
\DeclareOption{russian2o}{\mtcselectlanguage{russian2o}}%
\DeclareOption{samin}{\mtcselectlanguage{samin}}%
\DeclareOption{scottish}{\mtcselectlanguage{scottish}}%
\DeclareOption{serbian}{\mtcselectlanguage{serbian}}%
\DeclareOption{serbianc}{\mtcselectlanguage{serbianc}}%
\DeclareOption{slovak}{\mtcselectlanguage{slovak}}%
\DeclareOption{slovene}{\mtcselectlanguage{slovene}}%
\DeclareOption{spanish}{\mtcselectlanguage{spanish}}%
\DeclareOption{spanish2}{\mtcselectlanguage{spanish2}}%
\DeclareOption{spanish3}{\mtcselectlanguage{spanish3}}%
\DeclareOption{swedish}{\mtcselectlanguage{swedish}}%
\DeclareOption{thai}{\mtcselectlanguage{thai}}%
\DeclareOption{turkish}{\mtcselectlanguage{turkish}}%
\DeclareOption{UKenglish}{\mtcselectlanguage{UKenglish}}%
\DeclareOption{ukraineb}{\mtcselectlanguage{ukraineb}}%
\DeclareOption{USenglish}{\mtcselectlanguage{USenglish}}%
\DeclareOption{usorbian}{\mtcselectlanguage{usorbian}}%
\DeclareOption{vietnam}{\mtcselectlanguage{vietnam}}%
\DeclareOption{vietnamese}{\mtcselectlanguage{vietnamese}}%
\DeclareOption{welsh}{\mtcselectlanguage{welsh}}%
```

5.76 L'option hints

\if@mtc@hints@
\if@mtc@hints@w@
\if@mtc@hints@giben@

Tout d'abord, nous définissons quelques indicateurs :
- L'indicateur \if@mtc@hints@ est vrai si l'option hints est demandée (défaut).
- L'indicateur \if@mtc@hints@w@ est forcé à vrai si nous détectons que certaines commandes de sectionnement ont été altérées depuis le chargement de la classe de document.
- L'indicateur \if@mtc@hints@given@ est forcé à vrai si l'option hints détecte quelque chose de curieux et écrit des messages dans le fichier .log.

```
\newif\if@mtc@hints@ \@mtc@hints@true
\newif\if@mtc@hints@w@ \@mtc@hints@w@false
\newif\if@mtc@hints@given@ \@mtc@hints@given@false
```

\DeclareOption
\if@mtc@hints@

Nous déclarons les options `hints` (défaut) et `nohints` :

```
\DeclareOption{hints}{\@mtc@hints@true}
\DeclareOption{nohints}{\@mtc@hints@false}
```

\mtc@hints@begindoc
\AtBeginDocument

L'option `hints` est constituée de trois parties : la première, `\mtc@hints@begindoc`, est exécutée via `\AtBeginDocument` et regarde si certains paquetages ou classes sont chargés, puis donne des avertissements à propos de leur compatibilité avec `minitoc`.

La deuxième partie est faite de petits fragments de code insérés dans le code de `minitoc`, pour vérifier que certaines macros sont appelées dans le bon ordre.

\mtc@hints@enddoc
\AtEndDocument

La troisième et dernière partie, `\mtc@hints@enddoc`, est exécutée via `\AtEndDocument` et examine certains indicateurs positionnés par les première et deuxième parties. Puis, si nécessaire, elle écrit quelques informations dans le fichier `.log` et/ou des avertissements sur l'écran et dans le fichier `.log`. Notez que l'option `hints` ne signale pas d'erreurs, seulement des informations et des avertissements, et donc elle n'interrompt pas l'exécution de LaTeX.

5.76.1 Première partie : `\mtc@hints@begindoc`

\mtc@hints@begindoc

Nous déclarons la première partie de l'option `hints` (pour `\AtBeginDocument`) :

```
\def\mtc@hints@begindoc{%
\PackageInfo{minitoc(hints)}%
  {***** You requested the hints option *****\MessageBreak
   ***** Some hints are eventually given below *****\@gobble}%
```

5.76.1.1 *Hint* sur le paquetage appendix

\@ifpackageloaded
\if@mtc@hints@given@

Nous testons la présence du paquetage `appendix` [46] :

```
\@ifpackageloaded{appendix}{%
\@mtc@hints@given@true
\PackageInfo{minitoc(hints)}%
  {--- The appendix package is loaded. \MessageBreak
   See the minitoc package documentation\MessageBreak
   for specific precautions\@gobble}}{}%
```

5.76.1.2 *Hint* sur le paquetage `tocbibind`

\@ifpackageloaded
\if@mtc@hints@given@

Nous testons la présence du paquetage `tocbibind` [47] :

```
\@ifpackageloaded{tocbibind}%
   {\@mtc@hints@given@true
    \PackageInfo{minitoc(hints)}%
       {--- The tocbibind package is loaded. \MessageBreak
        See the minitoc package documentation
        for specific precautions\@gobble}}{}%
```

5.76.1.3 *Hint* sur le paquetage `tocloft`

\@ifpackageloaded
\if@mtc@hints@given@

Nous testons la présence du paquetage `tocloft` [44] :

```
\@ifpackageloaded{tocloft}%
   {\@mtc@hints@given@true
    \PackageInfo{minitoc(hints)}%
       {--- The tocloft package is loaded. \MessageBreak
        See the minitoc package documentation
        for specific precautions\@gobble}}{}%
```

5.76.1.4 *Hint* sur le paquetage `titletoc`

\@ifpackageloaded
\if@mtc@hints@given@

Nous testons la présence du paquetage `titletoc` [8], et émettons un avertissement, puisque ce paquetage est incompatible avec `minitoc` :

```
\@ifpackageloaded{titletoc}%
   {\@mtc@hints@given@true
    \PackageWarningNoLine{minitoc(hints)}%
       {--- The titletoc package is loaded. \MessageBreak
        It is incompatible with the minitoc package}}{}%
```

5.76.1.5 *Hint* sur le paquetage `placeins`

\@ifpackageloaded
\if@mtc@ss@insection@
\@ifpackagewith
\if@mtc@hints@given@

Nous testons si le paquetage `placeins` [3] est chargé et, si oui, nous regardons si les bonnes options ont été choisies :

```
\@ifpackageloaded{placeins}%
   {\if@mtc@ss@insection@
    \@ifpackagewith{placeins}{section,below}{}%
       {\@mtc@hints@given@true
        \PackageWarningNoLine{minitoc(hints)}%
```

```
          {--- The placeins package is loaded without
           \MessageBreak
           section and below options, but minitoc used
           \MessageBreak
           the insection option which implies them.
           \MessageBreak
           Try to inverse the loading order and use coherent options.
           \MessageBreak
           You may have got a message
             ! LaTeX Error: Option clash for package placeins}%
      }%
   \fi
  \@ifpackagelater{placeins}{2005/04/18}{}{%
     \@mtc@hints@given@true
     \PackageWarningNoLine{minitoc(hints)}%
{--- The placeins package loaded is
 \MessageBreak
 too old. It should have a version date of
 \MessageBreak
 2005/04/18 at least}%
  }%
  }{}%
```

5.76.1.6 *Hint* sur la classe memoir

\@ifclassloaded
\if@mtc@hints@given@

Nous testons si la classe memoir [48] est chargée :

```
\@ifclassloaded{memoir}%
  {\@mtc@hints@given@true
   \PackageInfo{minitoc(hints)}%
      {--- The memoir class is loaded. \MessageBreak
       See the minitoc package documentation for specific precautions\@gobble}{}%
  }{}%
```

5.76.1.7 *Hint* sur les classes amsart et amsproc

\@ifclassloaded
\if@mtc@hints@given@

Nous regardons si l'une des classes amsart ou amsproc a été chargée et émettons un avertissement, puisque ces classes sont incompatibles avec minitoc :

```
\@ifclassloaded{amsart}%
  {\@mtc@hints@given@true
   \PackageWarningNoLine{minitoc(hints)}%
      {--- The amsart class is loaded. \MessageBreak
       It is incompatible with the minitoc package}}{}%
\@ifclassloaded{amsproc}%
  {\@mtc@hints@given@true
   \PackageWarningNoLine{minitoc(hints)}%
```

```
      {--- The amsproc class is loaded. \MessageBreak
       It is incompatible with the minitoc package}}{}%
```

5.76.1.8 *Hint* sur la classe amsbook

\@ifclassloaded
\if@mtc@hints@given@

Nous testons si la classe amsbook est chargée :

```
\@ifclassloaded{amsbook}%
  {\@mtc@hints@given@true
   \PackageInfo{minitoc(hints)}%
      {--- The amsbook class is loaded. \MessageBreak
       See the minitoc package documentation for specific precautions\@gobble}{}%
  }{}%
```

5.76.1.9 *Hint* sur le paquetage abstract

\@ifpackageloaded
\if@mtc@abstract@loaded@
\@ifpackagewith
\if@mtc@hints@given@

Nous testons la présence du paquetage abstract [45], puis ses options :

```
\@ifpackageloaded{abstract}%
  {\@mtc@abstract@loaded@true%
  \@ifpackagewith{abstract}{addtotoc}%
     {\@mtc@hints@given@true
      \PackageInfo{minitoc(hints)}%
         {The ‘‘abstract’’ package has been loaded with\MessageBreak
          the ‘‘addtotoc’’ option.\MessageBreak
          You need to look at the documentation to adjust.}
     }}{}
```

5.76.1.10 *Hint* sur l'altération des commandes de sectionnement

\part
\chapter
\section
\AtBeginDocument
\mtc@hints@part
\mtc@hints@chapter
\mtc@hints@section

Pour vérifier si les commandes de sectionnement \part, \chapter ou \section ont été altérées par un paquetage ou dans le préambule, nous les comparons (lors de l'exécution d'un bloc \AtBeginDocument) avec leurs versions sauvegardées (par le paquetage minitoc lors de son chargement) \mtc@hints@part, \mtc@hints@chapter et \mtc@hints@section. Pour chacune de ces commandes de sectionnement, nous devons effectuer la comparaison pour la commande elle-même, sa branche non étoilée et sa branche étoilée. Mais le paquetage hyperref [39] peut interférer, d'où les précautions formelles dans les messages.

5.76.1.10.1 Altération de \part :

\if@mtc@hints@w@ \@ifundefined \part \mtc@hints@part \@part \mtc@hints@@part \@spart \mtc@hints@@spart \if@mtc@hints@given@ \if@mtc@hyper@used@

Nous regardons si \part, \@part and \@spart ont été altérées :

```
\@mtc@hints@w@false
\@ifundefined{part}{}{\ifx\part\mtc@hints@part\relax
   \else\@mtc@hints@w@true\fi}
\@ifundefined{part}{}{\ifx\@part\mtc@hints@@part\relax
   \else\@mtc@hints@w@true\fi}
\@ifundefined{part}{}{\ifx\@spart\mtc@hints@@spart\relax
   \else\@mtc@hints@w@true\fi}
\if@mtc@hints@w@\@mtc@hints@given@true%
   \PackageWarningNoLine{minitoc(hints)}%
      {--- The \string\part\space command is altered after minitoc}
   \if@mtc@hyper@used@
      \PackageWarningNoLine{minitoc(hints)}%
         {--- It may be the consequence of loading the ‘‘hyperref’’ package}
   \fi
\fi
```

5.76.1.10.2 Altération de \chapter :

\if@mtc@hints@w@ \@ifundefined \chapter \mtc@hints@chapter \@chapter \mtc@hints@@chapter \@schapter \mtc@hints@@schapter \if@mtc@hints@given@ \if@mtc@hyper@used@

Nous regardons si \chapter, \@chapter and \@schapter ont été altérées :

```
\@mtc@hints@w@false
\@ifundefined{chapter}{}{\ifx\chapter\mtc@hints@chapter\relax
   \else\@mtc@hints@w@true\fi}%
\@ifundefined{chapter}{}{\ifx\@chapter\mtc@hints@@chapter\relax
   \else\@mtc@hints@w@true\fi}%
\@ifundefined{chapter}{}{\ifx\@schapter\mtc@hints@@schapter\relax
   \else\@mtc@hints@w@true\fi}%
\if@mtc@hints@w@\@mtc@hints@given@true%
   \PackageWarningNoLine{minitoc(hints)}%
      {--- The \string\chapter\space command is altered after minitoc}
   \if@mtc@hyper@used@
      \PackageWarningNoLine{minitoc(hints)}%
         {--- It may be the consequence of loading the ‘‘hyperref’’ package}
   \fi
\fi
```

5.76.1.10.3 Altération de \section :

\if@mtc@hints@w@ \@ifundefined \section \mtc@hints@section \@sect \mtc@hints@@sect \@ssect \mtc@hints@@ssect \if@mtc@hints@given@ \if@mtc@hyper@used@

Nous regardons si \section, \@sect and \@ssect ont été altérées :

```
\@mtc@hints@w@false
\@ifundefined{chapter}%
   {\@ifundefined{section}{}{\ifx\section\mtc@hints@section\relax\else
```

```
    \@mtc@hints@w@true\fi}
  \@ifundefined{section}{}{\ifx\@sect\mtc@hints@@sect\relax\else
    \@mtc@hints@w@true\fi}
  \@ifundefined{section}{}{\ifx\@ssect\mtc@hints@@ssect\relax\else
    \@mtc@hints@w@true\fi}
\if@mtc@hints@w@\@mtc@hints@given@true%
  \PackageWarningNoLine{minitoc(hints)}%
    {--- The \string\section\space command is altered after minitoc}
  \if@mtc@hyper@used@
    \PackageWarningNoLine{minitoc(hints)}%
      {--- It may be the consequence of loading the ‘‘hyperref’’ package}
  \fi
  \relax\else\fi}{}%
```

5.76.1.11 *Hint* sur la cohérence des séquences d'appel de certaines commandes

Et enfin, nous préparons les tests de cohérence sur les séquences d'appel de triplets de commandes associées telles que `\doparttoc`, `\parttoc` et `\[fake]listofcontents`, et similaires : pour pouvoir utiliser `\parttoc`, un fichier table des matières doit avoir été créé via `\[fake]listofcontents` et éclaté en fichiers parttocs via `\doparttoc`.

`\if@mtc@hints@` `\if@mtc@toc@used@` `\if@mtc@lof@used@` `\if@mtc@lot@used@`

Donc nous déclarons quelques indicateurs :

```
\if@mtc@hints@
\newif\if@mtc@toc@used@ \global\@mtc@toc@used@false
\newif\if@mtc@lof@used@ \global\@mtc@lof@used@false
\newif\if@mtc@lot@used@ \global\@mtc@lot@used@false
```

`\mtc@sv@tableofcontents` `\tableofcontents` `\if@mtc@toc@used@` `\mtc@sv@listoffigures` `\listoffigures` `\if@mtc@lof@used@` `\mtc@sv@listoftables` `\listoftables` `\if@mtc@lot@used@`

Puis nous modifions les commandes concernées pour positionner l'indicateur correspondant lorsqu'elles sont utilisées. Nous commençons par les commandes des sommaires principaux :

```
\let\mtc@sv@tableofcontents\tableofcontents
\def\tableofcontents%
  {\global\@mtc@toc@used@true\mtc@sv@tableofcontents}
\let\mtc@sv@listoffigures\listoffigures
\def\listoffigures%
  {\global\@mtc@lof@used@true\mtc@sv@listoffigures}
\let\mtc@sv@listoftables\listoftables
\def\listoftables%
  {\global\@mtc@lot@used@true\mtc@sv@listoftables}
```

`\mtc@sv@faketableofcontents` `\faketableofcontents` `\if@mtc@toc@used@` `\mtc@sv@fakelistoffigures` `\fakelistoffigures` `\if@mtc@lof@used@` `\mtc@sv@fakelistoftables` `\fakelistoftables` `\if@mtc@lot@used@`

Puis, leurs cousines « fake » :

```
\let\mtc@sv@faketableofcontents\faketableofcontents
\def\faketableofcontents%
  {\global\@mtc@toc@used@true\mtc@sv@faketableofcontents}
```

```
\let\mtc@sv@fakelistoffigures\fakelistoffigures
\def\fakelistoffigures%
   {\global\@mtc@lof@used@true\mtc@sv@fakelistoffigures}
\let\mtc@sv@fakelistoftables\fakelistoftables
\def\fakelistoftables%
   {\global\@mtc@lot@used@true\mtc@sv@fakelistoftables}
\fi
```

\mtc@hints@begindoc

Et la définition de \mtc@hints@begindoc est terminée (elle commence à la section 5.76.1 page 192) :

```
}
```

5.76.2 Partie finale : \mtc@hints@enddoc

\mtc@hints@enddoc
\AtEndDocument

La partie finale de l'option hints est exécutée via \AtEndDocument. Son code est dans la macro \mtc@hints@enddoc. C'est une séquence de tests sur les paquetages et classes chargés et sur les indicateurs positionnés durant les première et deuxième parties de cette option. Tout d'abord, nous déclarons la macro \mtc@hints@enddoc :

```
\def\mtc@hints@enddoc{%
```

5.76.2.1 *Hint* sur \sect-lof|lot et l'option insection

\if@mtc@sect@floats@
\if@dosectlof@used@
\if@dosectlot@used@
\if@sectlof@used@
\if@sectlot@used@
\if@mtc@section@def@

Nous regardons si des listes de figures ou de tableaux ont été demandées au niveau section.

```
\if@dosectlof@used@\@mtc@sect@floats@true\fi
\if@dosectlot@used@\@mtc@sect@floats@true\fi
\if@sectlof@used@\@mtc@sect@floats@true\fi
\if@sectlot@used@\@mtc@sect@floats@true\fi
\if@mtc@section@def@
```

\if@mtc@placeinsLoaded@
\if@mtc@sect@floats@
\if@mtc@hints@w@
\if@mtc@hints@given@

Si oui, nous vérifions que le paquetage placeins [3] a été chargé avec les options correctes ou que l'option insection du paquetage minitoc a été invoquée. Sinon, un avertissement est émis.

```
\if@mtc@placeinsLoaded@
\else
  \if@mtc@sect@floats@%
   \PackageWarningNoLine{minitoc(hints)}%
      {***** You are using \string\dosectlof\space and/or \MessageBreak
       \string\dosectlot, \string\sectlof\space and/or \string\sectlot,
       \MessageBreak
       hence the ‘‘insection’’ package option is recommended. *****}%
```

```
  \@mtc@hints@w@true
  \@mtc@hints@given@true
 \fi
\fi
```

5.76.2.2 Partie finale des tests de cohérence

Nous testons si \parttoc a été utilisée \doparttoc, etc., pour chaque paire de commandes préparation/insertion.

\if@mtc@part@def@ \if@parttoc@used@ \if@doparttoc@used@ \if@mtc@hints@given@ \if@partlof@used@ \if@dopartlof@used@ \if@partlot@used@ \if@dopartlot@used@

Pour les commandes au niveau partie :

```
\if@mtc@part@def@
 \if@parttoc@used@
  \if@doparttoc@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\parttoc\space
      but not \string\doparttoc\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@partlof@used@
  \if@dopartlof@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\partlof\space
      but not \string\dopartlof\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@partlot@used@
  \if@dopartlot@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\partlot\space
      but not \string\dopartlot\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
\fi
```

\if@mtc@chapter@def@ \if@minitoc@used@ \if@dominitoc@used@ \if@mtc@hints@given@ \if@minilof@used@ \if@dominilof@used@ \if@minilot@used@ \if@dominilot@used@

Pour les commandes au niveau chapitre :

```
\if@mtc@chapter@def@
 \if@minitoc@used@
  \if@dominitoc@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\minitoc\space
      but not \string\dominitoc\space*****}
  \@mtc@hints@given@true
```

```
  \fi
 \fi
 \if@minilof@used@
  \if@dominilof@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\minilof\space
      but not \string\dominilof\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@minilot@used@
  \if@dominilot@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\minilot\space
      but not \string\dominilot\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
\fi
```

\if@mtc@section@def@
\if@secttoc@used@
\if@dosecttoc@used@
\if@mtc@hints@given@
\if@sectlof@used@
\if@dosectlof@used@
\if@sectlot@used@
\if@dosectlot@used@

Pour les commandes au niveau section :

```
\if@mtc@section@def@
 \if@secttoc@used@
  \if@dosecttoc@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\secttoc\space
      but not \string\dosecttoc\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@sectlof@used@
  \if@dosectlof@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\sectlof\space
      but not \string\dosectlof\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@sectlot@used@
  \if@dosectlot@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\sectlot\space
      but not \string\dosectlot\space*****}
  \fi
 \fi
\fi
```

5.76.2.3 Vérifier si les tables principales ont été préparées (première partie)

Nous testons maintenant si une macro `\doparttoc` a été appelée sans aucune commande `\parttoc` correspondante, car sinon cet appel est vain. Nous faisons de même pour chaque commande analogue.

\if@mtc@part@def@
\if@doparttoc@used@
\if@parttoc@used@
\if@mtc@hints@given@
\if@dopartlof@used@
\if@partlof@used@
\if@dopartlot@used@
\if@partlot@used@

Commandes au niveau partie :

```
\if@mtc@part@def@
 \if@doparttoc@used@
   \if@parttoc@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\doparttoc\space
       but not \string\parttoc\space*****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@dopartlof@used@
   \if@partlof@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\dopartlof\space
       but not \string\partlof\space*****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@dopartlot@used@
   \if@partlot@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\dopartlot\space
       but not \string\partlot\space*****}
   \@mtc@hints@given@true
   \fi
 \fi
\fi
```

\if@mtc@chapter@def@
\if@dominitoc@used@
\if@minitoc@used@
\if@mtc@hints@given@
\if@dominilof@used@
\if@minilof@used@
\if@dominilot@used@
\if@minilot@used@

Commandes au niveau chapitre :

```
\if@mtc@chapter@def@
 \if@dominitoc@used@
   \if@minitoc@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\dominitoc\space
       but not \string\minitoc\space*****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@dominilof@used@
   \if@minilof@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\dominilof\space
```

```
      but not \string\minilof\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@dominilot@used@
  \if@minilot@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\dominilot\space
      but not \string\minilot\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
\fi
```

\if@mtc@section@def@
\if@dosecttoc@used@
\if@secttoc@used@
\if@mtc@hints@given@
\if@dosectlof@used@
\if@sectlof@used@
\if@dosectlot@used@
\if@sectlot@used@

Commandes au niveau section :

```
\if@mtc@section@def@
 \if@dosecttoc@used@
  \if@secttoc@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\dosecttoc\space
      but not \string\secttoc\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@dosectlof@used@
  \if@sectlof@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\dosectlof\space
      but not \string\sectlof\space*****}
  \@mtc@hints@given@true
  \fi
 \fi
 \if@dosectlot@used@
  \if@sectlot@used@\else
  \PackageWarningNoLine{minitoc(hints)}%
     {**** You have used \string\dosectlot\space
      but not \string\sectlot\space*****}
  \fi
 \fi
\fi
\fi
```

5.76.2.4 Vérifier si les tables principales ont été préparées (seconde partie)

Un autre test de cohérence vérifie que si `\parttoc` a été appelée, alors la macro `\tableofcontents` ou `\faketableofcontents` a aussi été appelée (pour créer le fichier de contenu nécessaire) ; et des tests similaires sont effectués pour les autres commandes de mini-tables.

\if@mtc@part@def@
\if@mtc@hints@given@
\if@parttoc@used@
\ifmtc@toc@used@
\if@partlof@used@
\ifmtc@lof@used@
\if@partlot@used@
\ifmtc@lot@used@

Commandes au niveau partie :

```
\if@mtc@part@def@
 \if@parttoc@used@
   \if@mtc@toc@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\parttoc\space but not\MessageBreak
       \string\tableofcontents\space nor
       \string\faketableofcontents\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@partlof@used@
   \if@mtc@lof@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\partlof\space but not\MessageBreak
       \string\listoffigures\space nor
       \string\fakelistoffigures\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@partlot@used@
   \if@mtc@lot@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\partlot\space but not\MessageBreak
       \string\listoftables\space nor
       \string\fakelistoftables\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
\fi
```

\if@mtc@chapter@def@
\if@mtc@hints@given@
\if@minitoc@used@
\ifmtc@toc@used@
\if@minilof@used@
\ifmtc@lof@used@
\if@minilot@used@
\ifmtc@lot@used@

Commandes au niveau chapitre :

```
\if@mtc@chapter@def@
 \if@minitoc@used@
   \if@mtc@toc@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\minitoc\space but not\MessageBreak
       \string\tableofcontents\space nor
       \string\faketableofcontents\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@minilof@used@
   \if@mtc@lof@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\minilof\space but not\MessageBreak
       \string\listoffigures\space nor
       \string\fakelistoffigures\space *****}
   \@mtc@hints@given@true
   \fi
```

```
 \fi
 \if@minilot@used@
   \if@mtc@lot@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\minilot\space but not\MessageBreak
       \string\listoftables\space nor
       \string\fakelistoftables\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
\fi
```

\if@mtc@section@def@
\if@mtc@hints@given@
\if@secttoc@used@
\ifmtc@toc@used@
\if@sectlof@used@
\ifmtc@lof@used@
\if@sectlot@used@
\ifmtc@lot@used@

Commandes au niveau section :

```
\if@mtc@section@def@
 \if@secttoc@used@
   \if@mtc@toc@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\secttoc\space but not\MessageBreak
       \string\tableofcontents\space nor
       \string\faketableofcontents\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@sectlof@used@
   \if@mtc@lof@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\sectlof\space but not\MessageBreak
      \string\listoffigures\space nor
      \string\fakelistoffigures\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
 \if@sectlot@used@
   \if@mtc@lot@used@\else
   \PackageWarningNoLine{minitoc(hints)}%
      {**** You have used \string\sectlot\space but not\MessageBreak
       \string\listoftables\space nor
       \string\fakelistoftables\space *****}
   \@mtc@hints@given@true
   \fi
 \fi
\fi
```

5.76.2.5 Tester le nombre de mini-tables, dans le cas de suffixes courts

\mtc@hints@checklongext
\if@longextensions@
\if@mtc@part@def@
\value
\if@mtc@hints@given@true
\if@mtc@chapter@def@
\if@mtc@section@def@

Si les suffixes courts sont utilisés, vous ne pouvez utiliser que 99 mini-tables de chaque genre. Si davantage de mini-tables sont créées, les fichiers auxiliaires seront écrasés ; le centième fichier minitoc `\jobname.U100` a son nom tronqué à `\jobname.U10`, qui est déjà le dixième fichier minitoc. Donc nous avons besoin d'une indication (*hint*) pour signaler cette situation. Le code est assez simple, mais le remède est amer et coûteux : soit utiliser un meilleur système d'exploitation[9], soit revoir la conception du document.

```
\def\mtc@hints@checklongext{%
\if@longextensions@
\else
  \if@mtc@part@def@
    \ifnum 99 < \value{ptc}
       \@mtc@hints@given@true
       \PackageWarningNoLine{minitoc(hints)}%
          {**** You have used short extensions
           and more than 99 parts *****}
    \fi
  \fi
  \if@mtc@chapter@def@
    \ifnum 99 < \value{mtc}
       \@mtc@hints@given@true
       \PackageWarningNoLine{minitoc(hints)}%
          {**** You have used short extensions
           and more than 99 chapters *****}
    \fi
  \else
    \if@mtc@section@def@
      \ifnum 99 < \value{stc}
         \@mtc@hints@given@true
         \PackageWarningNoLine{minitoc(hints)}%
            {**** You have used short extensions
             and more than 99 sections *****}
      \fi
    \fi
  \fi
\fi}
\mtc@hints@checklongext
```

5.76.2.6 Partie finale du *hint* sur le paquetage `sectsty`

`\if@mtc@sectstyLoaded@`
`\if@mtc@sectstyLoaded@a@`
`\if@mtc@hints@given@`

Nous testons si `sectsty` a été chargé avant (correct) ou après (incorrect) `minitoc`. Voir la section 5.9 page 74.

```
\if@mtc@sectstyLoaded@\else
  \if@mtc@sectstyLoaded@a@
     \PackageWarningNoLine{minitoc(hints)}%
     {**** The sectsty package should be loaded \MessageBreak
           **before** the minitoc package ****}
     \@mtc@hints@given@true
  \fi
\fi
```

[9]Dans le long terme, un bon investissement.

5.76.2.7 Tester si des mini-tables vides ont été détectées

Nous testons pour chaque sorte de mini-tables.

\if@mtc@empty@parttoc@ Pour les parttocs :

```
\if@mtc@empty@parttoc@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty parttocs ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@partlof@ Pour les partlofs :

```
\if@mtc@empty@partlof@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty partlofs ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@partlot@ Pour les partlots :

```
\if@mtc@empty@partlot@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty partlots ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@minitoc@ Pour les minitocs :

```
\if@mtc@empty@minitoc@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty minitocs ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@minilof@ Pour les minilofs :

```
\if@mtc@empty@minilof@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty minilofs ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@minilot@ Pour les minilots :

```
\if@mtc@empty@minilot@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty minilots ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@secttoc@ Pour les secttocs :

```
\if@mtc@empty@secttoc@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty secttocs ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@sectlof@ Pour les sectlofs :

```
\if@mtc@empty@sectlof@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty sectlofs ****}
  \@mtc@hints@given@true
\fi
```

\if@mtc@empty@sectlot@ Pour les sectlots :

```
\if@mtc@empty@sectlot@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have attempted to insert empty sectlots ****}
  \@mtc@hints@given@true
\fi
```

5.76.2.8 Tester si des commandes obsolètes ont été utilisées

Ce *hint* est juste un rappel dans le cas où vous auriez utilisé des commandes obsolètes, qui sont aussi signalées dans le fichier `.log`.

\if@firstpartis@used@ Macro obsolète `\firstpartis` :

```
\if@firstpartis@used@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have invoked an obsolete command: \string\firstpartis\space ****}
  \@mtc@hints@given@true
\fi
```

\if@firstchapteris@used@

Macro obsolète \firstchapteris :

```
\if@firstchapteris@used@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have invoked an obsolete command: \string\firstchapteris\space ****}
  \@mtc@hints@given@true
\fi
```

\if@firstsectionis@used@

Macro obsolète \firstsectionis :

```
\if@firstsectionis@used@
  \PackageWarningNoLine{minitoc(hints)}%
  {**** You have invoked an obsolete command: \string\firstsectionis\space ****}
  \@mtc@hints@given@true
\fi
```

5.76.2.9 Tester si des *hints* ont été écrits

\if@mtc@hints@given@
\mtc@hints@enddoc

Nous arrivons à la fin de la troisième partie de l'option hints : si des problèmes ont été détectés, un avertissement est affiché ; cet avertissement n'est pas affiché mais seulement écrit dans le fichier .log si aucun problème n'a été détecté. Et nous terminons la macro \mtc@hints@enddoc par une accolade fermante.

```
\if@mtc@hints@given@
  \PackageWarningNoLine{minitoc(hints)}%
    {***** Some hints have been written\MessageBreak
     in the \jobname.log file. *****}
\else
  \PackageInfo{minitoc(hints)}%
    {***** No hints have been written\MessageBreak
     in the \jobname.log file. *****\@gobble}
\fi
}
```

5.77 Traitement des options

\ExecuteOptionsenglish

Tout d'abord, nous exécutons l'option de langue par défaut, english :

```
\ExecuteOptions{english}
```

\ProcessOptions*

Puis nous exécutons toutes les options demandées ; pour la plupart des options, ceci consiste seulement à positionner un indicateur, ou à charger un fichier pour les options de langue.

```
\ProcessOptions*
```

Nous examinons maintenant les indicateurs pour certaines options et exécutons les actions nécessaires.

5.77.1 Traitement de l’option insection

\if@mtc@ss@insection@
\RequirePackage
\@ifpackageloaded
\if@mtc@placeinsLoaded@

Pour l’option `insection`, nous chargeons le paquetage `placeins` [3] avec ses options `verbose`, `section` et `below` ; le chargement correct est vérifié :

```
\if@mtc@ss@insection@
   \RequirePackage[section,below,verbose]{placeins}[2005/04/18]%
   \@ifpackageloaded{placeins}%
      {\@mtc@placeinsLoaded@true}%
      {\@mtc@placeinsLoaded@false}%
\fi
```

5.77.2 Traitement de l’option notoccite

\if@mtc@notoccite@
\RequirePackage

Pour l’option `notoccite`, nous chargeons simplement le paquetage `notoccite` [2] :

```
\if@mtc@notoccite@
   \RequirePackage{notoccite}%
\fi
```

5.77.3 Traitement de l’option hints

\if@mtc@hints@
\AtBeginDocument
\mtc@hints@begindoc
\AtEndDocument
\mtc@hints@enddoc

Pour l’option `hints`, nous plaçons sa première partie dans un bloc `\AtBeginDocument` et sa troisième (dernière) partie dans un bloc `\AtEndDocument` :

```
\if@mtc@hints@
   \AtBeginDocument{\mtc@hints@begindoc}%
   \AtEndDocument{\mtc@hints@enddoc}%
\fi
```

5.77.4 Sauvegarde des commandes de sectionnement

Et, enfin, nous sauvegardons les définitions des commandes de sectionnement (avec leurs branches non étoilées et étoilées), pour des comparaisons (ceci est une partie de l’option `hints` exécutée dans le préambule).

\@ifundefined \mtc@hints@part \part \mtc@hints@@part \@part \mtc@hints@@spart \@spart

Pour la commande `\part` :

```
\@ifundefined{part}{}{\let\mtc@hints@part\part
                      \let\mtc@hints@@part\@part
                      \let\mtc@hints@@spart\@spart}
```

\@ifundefined \mtc@hints@chapter \chapter \mtc@hints@@chapter \@chapter \mtc@hints@@schapter \@schapter

Pour la commande `\chapter` :

```
\@ifundefined{chapter}{}{\let\mtc@hints@chapter\chapter
                      \let\mtc@hints@@chapter\@chapter
                      \let\mtc@hints@@schapter\@schapter}
```

\@ifundefined \mtc@hints@section \section \mtc@hints@@sect \@sect \mtc@hints@@ssect \@ssect

Pour la commande `\section` :

```
\@ifundefined{section}{}{\let\mtc@hints@section\section
                      \let\mtc@hints@@sect\@sect
                      \let\mtc@hints@@ssect\@ssect}
```

Et le paquetage est terminé.

```
⟨/minitoc⟩
```

5.78 Le fichier fminitoc.dtx

\jobname \input

Ce court fichier est nécessaire pour créer la documentation en français. Son rôle est de forcer `\jobname` à `fminitoc` au lieu de `minitoc`. Comme `minitoc.ins` engendre les fichiers `minitoc.lan` et `fminitoc.lan` qui établissent un numéro de langue `\LANG`, et comme `minitoc.dtx` lit `\jobname.lan`, la documentation peut être en plusieurs langues (anglais et français) dans `minitoc.dtx`, la langue étant choisie par des constructions `\ifcase\LANG\relax` ... `\or\relax` ... `\fi`. Les primitives `\relax` sont nécessaires pour éviter de mauvaises surprises.

```
⟨*fminitoc⟩
\ProvidesFile{fminitoc.dtx}[2005/09/16 minitoc french documentation start file]
```

```
\input minitoc.dtx
⟨/fminitoc⟩
```

Chapitre 6

Code commenté de `mtcoff.sty`

Sommaire

6.1 Pourquoi `mtcoff.sty` ?

Le paquetage `minitoc` [20, 19] requiert que l'utilisateur insère de nombreuses commandes dans le code source de son document, mais pas seulement dans le préambule du document. D'où le concept d'un paquetage de remplacement, `mtcoff` (c'est-à-dire, « *minitoc off* »), qui substitue à toutes les commandes et à tous les environnements du paquetage `minitoc` d'autres commandes et environnements ayant les mêmes noms et syntaxes, mais ne faisant rien (sauf émettre quelques avertissements bénins, dans des cas spéciaux). De cette manière, pour inhiber facilement le paquetage `minitoc`, il vous suffit d'écrire, dans le préambule de votre document, quelque chose comme ceci :

```
\usepackage[...options...]{minitoc}
%\usepackage{mtcoff}
```

alors le paquetage `minitoc` est activé avec les options spécifiées. Si vous modifiez ces deux lignes de la manière suivante :

```
%\usepackage[...options...]{minitoc}
\usepackage{mtcoff}
```

alors le paquetage `minitoc` est désactivé et toutes ses commandes et environnements sont ignorés. Ceci est beaucoup plus facile, rapide et sûr que de mettre en commentaire toutes les commandes et environnements de `minitoc`. De plus, cette opération est réversible.

6.2 Identification du paquetage

`\NeedsTeXFormat`
`\ProvidesPackage`

Tout d'abord, nous identifions le paquetage et vérifions la version de LaTeX[1] :

```
⟨*mtcoff⟩
\NeedsTeXFormat{LaTeX2e}%
\ProvidesPackage{mtcoff}[2005/09/16 v43 The mtcoff package]
```

6.3 Simuler compteurs et dimensions

`\count@`
`\dimen@`
`\c@minitocdepth`
`\mtcindent`
`\mtcskipamount`
`\bigskipamount`
`\c@parttocdepth`
`\mtcindent`
`\c@secttocdepth`
`\stcindent`
`\c@mtc`
`\c@ptc`
`\c@stc`

Puisque `minitoc` déclare quelques registres compteurs et dimensions, nous les simulons en utilisant `\count@` ou `\dimen@`. Pour `\mtcskipamount`, nous utilisons sa définition par défaut, `\bigskipamount`.

```
\let\c@minitocdepth\count@
\let\mtcindent\dimen@
\let\mtcskipamount\bigskipamount
\let\c@parttocdepth\count@
\let\ptcindent\dimen@
\let\c@secttocdepth\count@
\let\stcindent\dimen@
\let\c@mtc\count@
\let\c@ptc\count@
\let\c@stc\count@
```

[1]Cette vérification n'est pas vraiment utile pour le paquetage `mtcoff` lui-même, mais il est bon de vérifier que votre version de LaTeX n'est pas trop ancienne pour supporter `minitoc`.

6.4 Simuler les commandes utilisateur simples

\mtcskip \faketableofcontents \fakelistoffigures \fakelistoftables \adjustptc \adjustmtc \adjuststc \decrementptc \decrementmtc \decrementstc \incrementptc \incrementmtc \incrementstc

Certaines commandes utilisateur sont faciles à simuler :

```
\let\mtcskip\relax
\let\faketableofcontents\relax
\let\fakelistoffigures\relax
\let\fakelistoftables\relax
\newcommand{\adjustptc}[1][1]{\relax}
\newcommand{\adjustmtc}[1][1]{\relax}
\newcommand{\adjuststc}[1][1]{\relax}
\let\decrementptc\relax
\let\decrementmtc\relax
\let\decrementstc\relax
\let\incrementptc\relax
\let\incrementmtc\relax
\let\incrementstc\relax
```

\partend \partbegin \chapterend \chapterbegin \sectend \sectbegin

Les commandes suivantes ne sont pas appelées directement par l'utilisateur, dans des circonstances normales, mais il faut les simuler :

```
\let\partend\relax
\let\partbegin\relax
\let\chapterend\relax
\let\chapterbegin\relax
\let\sectend\relax
\let\sectbegin\relax
```

6.5 Simulation des commandes utilisateur ayant un argument optionnel

\gobbleopt@

Les commandes utilisateur ayant un argument optionnel sont simulées en utilisant la macro interne LaTeX `\@ifnextchar` (pour prendre l'argument optionnel) et la nouvelle commande utilitaire `\gobbleopt@`.

```
\def\gobbleopt@[#1]{\relax}
```

\@ifnextchar \doparttoc \dopartlof \dopartlot \parttoc \partlof \partlot

Commandes pour les mini-tables au niveau partie :

```
\def\doparttoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\dopartlof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\dopartlot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\parttoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
```

```
\def\partlof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\partlot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
```

\@ifnextchar \dominitoc \dominilof \dominilot \minitoc \minilof \minilot

Commandes pour les mini-tables au niveau chapitre :

```
\def\dominitoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\dominilof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\dominilot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\minitoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\minilof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\minilot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
```

\@ifnextchar \dosecttoc \dosectlof \dosectlot \secttoc \sectlof \sectlot

Commandes pour les mini-tables au niveau section :

```
\def\dosecttoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\dosectlof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\dosectlot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\secttoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\sectlof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
\def\sectlot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[l]}}
```

6.6 Inhibition des commandes internes

\@gobbletwo \tf@mtc \mtc@string \appendixmtc \l@xchapter \xchapter \pchapter \psection \l@xpart \xpart \l@xsect \xsect

Il nous faut aussi inhiber quelques commandes de minitoc, avec \relax (macros sans argument) ou \@gobbletwo (macros avec arguments) :

```
\let\tf@mtc\count@
\let\mtc@string\relax
\let\appendixmtc\relax
\let\l@xchapter\@gobbletwo
\let\xchapter\relax
\let\pchapter\relax
\let\psection\relax
\let\l@xpart\@gobbletwo
\let\xpart\relax
\let\l@xsect\@gobbletwo
\let\xsect\relax
```

6.7 Désactivation des commandes de fonte

\empty

Nous désactivons commandes de fonte de minitoc (telles que \mtcSSfont) avec \empty, car certains utilisateurs pourraient avoir utilisé :

```
\renewcommand{\mtcSSfont}{...}
```

ce qui ne marcherait pas si nous utilisions `\relax` ici.

`\ptcfont` `\ptcCfont` `\ptcSfont` `\ptcSSfont` `\ptcSSSfont` `\ptcPfont` `\ptcSPfont` `\plffont` `\pltfont` `\ptifont`

Fontes pour les mini-tables au niveau partie :

```
\let\ptcfont\empty
\let\ptcCfont\empty
\let\ptcSfont\empty
\let\ptcSSfont\empty
\let\ptcSSSfont\empty
\let\ptcPfont\empty
\let\ptcSPfont\empty
\let\plffont\empty
\let\pltfont\empty
\let\ptifont\empty
```

`\mtcfont` `\mtcSfont` `\mtcSSfont` `\mtcSSSfont` `\mtcPfont` `\mtcSPfont` `\mlffont` `\mltfont` `\mtifont`

Fontes pour les mini-tables au niveau chapitre :

```
\let\mtcfont\empty
\let\mtcSfont\empty
\let\mtcSSfont\empty
\let\mtcSSSfont\empty
\let\mtcPfont\empty
\let\mtcSPfont\empty
\let\mlffont\empty
\let\mltfont\empty
\let\mtifont\empty
```

`\stcfont` `\stcSSfont` `\stcSSSfont` `\stcPfont` `\stcSPfont` `\slffont` `\sltfont` `\stifont`

Fontes pour les mini-tables au niveau section :

```
\let\stcfont\empty
\let\stcSSfont\empty
\let\stcSSSfont\empty
\let\stcPfont\empty
\let\stcSPfont\empty
\let\slffont\empty
\let\sltfont\empty
\let\stifont\empty
```

`\coffeefont`

Fonte pour les lignes de « pause-café » :

```
\let\coffeefont\empty
```

6.8 Inhibition des commandes \mtcset...

\@gobbletwo \mtcsetdepth \mtcsetfont \mtcsettitlefont \mtcsettitle \mtcsetformat \mtcsetfeature \mtcsetpagenumbers \mtcsetrules

Ces commandes utilisent deux ou trois arguments obligatoires :

```
\let\mtcsetdepth\@gobbletwo
\def\mtcsetfont#1#2#3{\empty}
\let\mtcsettitlefont\@gobbletwo
\let\mtcsettitle\@gobbletwo
\def\mtcsetformat#1#2#3{\empty}
\def\mtcsetfeature#1#2#3{\empty}
\let\mtcsetpagenumbers\@gobbletwo
\let\mtcsetrules\@gobbletwo
```

6.9 Inhibition des nouvelles commandes \l@...

\l@starpart \l@starchapter \l@starsection \l@starsubsection \l@starsubsubsection \l@starparagraph \l@starsubparagraph

Le paquetage `minitoc` définit les commandes `\l@starXXX` pour la mise en forme d'entrées de la table des matières pour des commandes de sectionnement étoilées. Nous revenons à la version non étoilée, lorsque cela est nécessaire :

```
\@ifundefined{part}{}{\let\l@starpart\l@part}
\@ifundefined{chapter}{}{\let\l@starchapter\l@chapter}
\@ifundefined{section}{}{\let\l@starsection\l@section}
\@ifundefined{subsection}{}{\let\l@starsubsection\l@subsection}
\@ifundefined{subsubsection}{}{\let\l@starsubsubsection\l@subsubsection}
\@ifundefined{paragraph}{}{\let\l@starparagraph\l@paragraph}
\@ifundefined{subparagraph}{}{\let\l@starsubparagraph\l@subparagraph}
```

6.10 Ignorer les commandes obsolètes

\@gobble \firstpartis \firstchapteris \firstsectionis

Nous ignorons simplement les commandes obsolètes (qui ont un seul argument, obligatoire) :

```
\let\firstpartis\@gobble
\let\firstchapteris\@gobble
\let\firstsectionis\@gobble
```

6.11 Inhibition de la commande \mtcselectlanguage

\@gobble \mtcselectlanguage

Cette commande a un seul argument, obligatoire :

```
\let\mtcselectlanguage\@gobble
```

6.12 Inhibition de la commande \mtcloadmlo

\@gobble
\mtcloadmlo

Cette commande a un seul argument, obligatoire :

```
\let\mtcloadmlo\@gobble
```

6.13 Inhibition des commandes pour les filets horizontaux

\ptcrule
\noptcrule
\mtcrule
\nomtcrule
\stcrule
\nostcrule
\plfrule
\noplfrule
\mlfrule
\nomlfrule
\slfrule
\noslfrule
\pltrule
\nopltrule
\mltrule
\nomltrule
\sltrule
\nosltrule

Ces commandes n'ont aucun argument :

```
\let\ptcrule\relax
\let\noptcrule\relax
\let\mtcrule\relax
\let\nomtcrule\relax
\let\stcrule\relax
\let\nostcrule\relax
\let\plfrule\relax
\let\noplfrule\relax
\let\mlfrule\relax
\let\nomlfrule\relax
\let\slfrule\relax
\let\noslfrule\relax
\let\pltrule\relax
\let\nopltrule\relax
\let\mltrule\relax
\let\nomltrule\relax
\let\sltrule\relax
\let\nosltrule\relax
```

6.14 Inhibition des commandes pour les numéros de page

\ptcpagenumbers \noptcpagenumbers \mtcpagenumbers \nomtcpagenumbers \stcpagenumbers \nostcpagenumbers \plfpagenumbers \noplfpagenumbers \mlfpagenumbers \nomlfpagenumbers \slfpagenumbers \noslfpagenumbers \pltpagenumbers \nopltpagenumbers \mltpagenumbers \nomltpagenumbers \sltpagenumbers \nosltpagenumbers

Ces commandes n'ont aucun argument :

```
\let\mtcpagenumbers\relax
\let\nomtcpagenumbers\relax
\let\stcpagenumbers\relax
\let\nostcpagenumbers\relax
\let\ptcpagenumbers\relax
\let\noptcpagenumbers\relax
\let\mlfpagenumbers\relax
\let\nomlfpagenumbers\relax
\let\slfpagenumbers\relax
\let\noslfpagenumbers\relax
\let\plfpagenumbers\relax
\let\noplfpagenumbers\relax
\let\mltpagenumbers\relax
\let\nomltpagenumbers\relax
\let\sltpagenumbers\relax
\let\nosltpagenumbers\relax
\let\pltpagenumbers\relax
\let\nopltpagenumbers\relax
```

6.15 Inhibition des commandes pour les dispositifs des mini-tables

Nous inhibons les commandes pour les dispositifs (telles que `\beforeparttoc`) avec `\empty`, car certains utilisateurs ont pu avoir fait :

```
\renewcommand{\beforeparttoc}{...}
```

ce qui ne marcherait pas si nous utilisions `\relax` ici. Ces commandes n'ont aucun argument,

\beforeparttoc \beforepartlof \beforepartlot \afterparttoc \afterpartlof \afterpartlot \thispageparttocstyle \thispagepartlofstyle \thispagepartlotstyle

Commandes pour des mini-tables au niveau partie :

```
\let\beforeparttoc\empty
\let\beforepartlof\empty
\let\beforepartlot\empty
\let\afterparttoc\empty
\let\afterpartlof\empty
\let\afterpartlot\empty
\let\thispageparttocstyle\empty
\let\thispagepartlofstyle\empty
\let\thispagepartlotstyle\empty
```

\beforeminitoc
\beforeminilof
\beforeminilot
\afterminitoc
\afterminilof
\afterminilot
\thispageminitocstyle
\thispageminilofstyle
\thispageminilotstyle

Commandes pour des mini-tables au niveau chapitre :

```
\let\beforeminitoc\empty
\let\beforeminilof\empty
\let\beforeminilot\empty
\let\afterminitoc\empty
\let\afterminilof\empty
\let\afterminilot\empty
\let\thispageminitocstyle\empty
\let\thispageminilofstyle\empty
\let\thispageminilotstyle\empty
```

\beforesecttoc
\beforesectlof
\beforesectlot
\aftersecttoc
\aftersectlof
\aftersectlot
\thispagesecttocstyle
\thispagesectlofstyle
\thispagesectlotstyle

Commandes pour des mini-tables au niveau section :

```
\let\beforesecttoc\empty
\let\beforesectlof\empty
\let\beforesectlot\empty
\let\aftersecttoc\empty
\let\aftersectlof\empty
\let\aftersectlot\empty
\let\thispagesecttocstyle\empty
\let\thispagesectlofstyle\empty
\let\thispagesectlotstyle\empty
```

6.16 Inhibition de divers indicateurs et commandes

\if@longextensions@
\iftightmtc
\ifktightmtc
\ifundottedmtc
\l@listof
\chapter

Il y a quelques indicateurs et commandes qu'il est bon de déclarer :

```
\newif\if@longextensions@ \@longextensions@true
\newif\iftightmtc \tightmtcfalse
\newif\ifktightmtc \ktightmtcfalse
\newif\ifundottedmtc \undottedmtcfalse
\let\l@listof\chapter
```

6.17 Précaution pour certaines commandes

\AtBeginDocument
\ifmtcoffwarn@

Certaines commandes de `minitoc` devraient éventuellement être remplacées si vous décidez d'arrêter *définitivement* l'utilisation du paquetage `minitoc` avec votre document. Donc nous déclarons un indicateur et un bloc `\AtEndDocument` pour signaler que vous avez utilisé ces commandes :

```
\newif\ifmtcoffwarn@ \mtcoffwarn@false
\AtEndDocument{\ifmtcoffwarn@
```

```
   \PackageWarningNoLine{mtcoff}%
      {You should scan (backwards) your .log file to find
       \MessageBreak
       some commands needing to be replaced if you decide to
       \MessageBreak
       DEFINITELY stop using minitoc for this document.
       \MessageBreak
       It is more wise to keep the \string\usepackage\space lines for
       \MessageBreak
       minitoc and mtcoff and to comment out only one of them}
\fi}
```

\mtcaddchapter
\mtcaddsection
\mtcaddpart
\ifmtcoffwarn@
\mtc@ck
\addcontentsline

Puis ces commandes sont inhibées de manière qu'elles positionnent l'indicateur et donnent un avertissement (ce qui est utile pour avoir le numéro de ligne) :

```
\newcommand{\mtcaddchapter}[1][]{\mtcoffwarn@true
   \PackageWarning{mtcoff}%
   {\protect\mtcaddchapter{...} should be replaced\MessageBreak
    by \protect\addcontentsline{toc}{chapter}{...}\MessageBreak}
\def\mtc@ck{#1}
\ifx\mtc@ck\empty
\else
   \addcontentsline{toc}{chapter}{#1}%
\fi}
\newcommand{\mtcaddsection}[1][]{\mtcoffwarn@true
   \PackageWarning{mtcoff}%
      {\protect\mtcaddsection{...} should be replaced\MessageBreak
       by \protect\addcontentsline{toc}{section}{...}\MessageBreak}
\def\mtc@ck{#1}
\ifx\mtc@ck\empty
\else
     \addcontentsline{toc}{part}{#1}%
\fi}
\newcommand{\mtcaddpart}[1][]{\mtcoffwarn@true
   \PackageWarning{mtcoff}%
      {\protect\mtcaddpart{...} should be replaced\MessageBreak
       by \protect\addcontentsline{toc}{part}{...}\MessageBreak}
\def\mtc@ck{#1}
\ifx\mtc@ck\empty
\else
   \addcontentsline{toc}{part}{#1}%
\fi}
```

6.18 Inhibition des commandes pour les « pauses-café »

\addcoffeeline
\coffeeline
\@gobble
\@Undottedtocline
\@Undottedtoclinep

Nous inhibons les commandes concernant les lignes de contenu pour les « pauses-café », ainsi que la version spécifique de ligne de contenu sans points de conduite :

```
\def\addcoffeeline#1#2#3{\relax}
\let\coffeeline\@gobble
\let\l@coffee\relax
\def\@Undottedtocline#1#2#3#4#5{\relax}
\def\@Undottedtoclinep#1#2#3#4#5{\relax}
```

6.19 Inhibition des environnements mtchideinmain...

mtchideinmaintoc
mtchideinmainlof
mtchideinmainlot

Ces environnements acceptent un argument optionnel :

```
\newenvironment{mtchideinmaintoc}[1][-1]%
   {\empty}{\empty}
\newenvironment{mtchideinmainlof}[1][-1]%
   {\empty}{\empty}
\newenvironment{mtchideinmainlot}[1][-1]%
   {\empty}{\empty}
```

6.20 Inhibition de la commande \mtcfixindex

\mtcficindex

Cette commande accepte un argument optionnel :

```
\newcommand{\mtcfixindex}[1][]{\relax}
```

6.21 Inhibition de la commande \mtcfixglossary

\mtcficindex

Cette commande accepte un argument optionnel :

```
\newcommand{\mtcfixglossary}[1][]{\relax}
```

6.22 Inhibition des commandes \addstarred...

\ifmtcoffwarn@
\addstarredpart
\addstarredchapter
\addstarredsection
\addcontentsline

Ces commandes devraient être remplacées par des commandes standard, mais mtcoff les simule et donne un avertissement, qui sera rappelé en fin de document :

```
\def\addstarredpart#1{\mtcoffwarn@true
   \PackageWarning{mtcoff}%
```

```
  {\protect\addstarredpart{...} should be replaced by\MessageBreak
   \protect\addcontentsline{toc}{part}{...}\MessageBreak}
  \addcontentsline{toc}{part}{#1}}
\def\addstarredchapter#1{\mtcoffwarn@true
  \PackageWarning{mtcoff}%
  {\protect\addstarredchapter{...} should be replaced by\MessageBreak
   \protect\addcontentsline{toc}{chapter}{...}\MessageBreak}
  \addcontentsline{toc}{chapter}{#1}}
\def\addstarredsection#1{\mtcoffwarn@true
  \PackageWarning{mtcoff}%
  {\protect\addstarredsection{...} should be replaced by\MessageBreak
   \protect\addcontentsline{toc}{section}{...}\MessageBreak}
  \addcontentsline{toc}{section}{#1}}
```

Et le paquetage `mtcoff` est terminé.

```
⟨/mtcoff⟩
```

Chapitre 7

Correction pour la classe memoir

Ce code doit être chargé pour corriger une incompatibilité du paquetage minitoc avec les versions récentes de la classe memoir.cls.

```
⟨*mtcpatchmem⟩
\NeedsTeXFormat{LaTeX2e}[1996/06/01]%
\ProvidesPackage{mtcpatchmem}%
   [2005/09/16 v43 Package mtcpatchmem]
\PackageInfo{mtcpatchmem}%
   {*** mtcpatchmem package to patch the memoir class ***\@gobble}
\renewcommand{\@m@mchapter}[1][]{%
  \def\ch@pt@c{#1}% capture first optional arg
   \@ifnextchar[{\@chapter}{\@chapter[]}%
}
\def\@chapter[#1]#2{%
% if |\ch@pt@c| is empty, no [ was found at all. Use |#2| as
% entry for all fields.
   \ifx\ch@pt@c\@empty
     \def\f@rtoc{#2}%
     \def\f@rhdr{#2}%
   \else
% otherwise at least one [ was found. If |#1| is empty then only
% one was found.
     \let\f@rtoc\ch@pt@c
     \ifx\@empty#1\@empty
       \let\f@rhdr\ch@pt@c
     \else
       \def\f@rhdr{#1}%
     \fi
   \fi
   \ifnum \c@secnumdepth >\m@ne
     \if@mainmatter
       \refstepcounter{chapter}%
     \fi
   \fi
```

```
  \chaptermark{\f@rhdr}%
  \ifartopt
    \@makechapterhead{#2}%
    \@afterheading
  \else
    \insertchapterspace
    \if@twocolumn
      \@topnewpage[\@makechapterhead{#2}]%
    \else
      \@makechapterhead{#2}%
    \fi
    \@afterheading
  \fi
  \ifnum \c@secnumdepth >\m@ne
    \if@mainmatter
      \ifanappendix
        \addcontentsline{toc}{appendix}{%
          \protect\chapternumberline{\thechapter}\f@rtoc}%
      \else
        \addcontentsline{toc}{chapter}{%
          \protect\chapternumberline{\thechapter}\f@rtoc}%
      \fi
    \else
      \addcontentsline{toc}{chapter}{\f@rtoc}%
    \fi
  \else
    \addcontentsline{toc}{chapter}{\f@rtoc}%
  \fi
  \ifheadnameref\M@gettitle{\f@rhdr}\else\M@gettitle{\f@rtoc}\fi
}
</mtcpatchmem>
```

Chapitre 8

Les fichiers minitoc de définition (.mld) et objets (.mlo) de langue

Sommaire

Ce chapitre montre le code de chaque fichier `.mld`. Un fichier `.mld` est un fichier de définition de langue pour minitoc (*minitoc language definition file*), qui définit les titres des mini-tables pour une langue donnée. Il contient souvent quelques informations sur son origine, si vous avez besoin de plus de détails. Il est chargé soit via une option du paquetage[1] dans la commande `\usepackage` pour le paquetage `minitoc`, soit via la commande :

`\mtcselectlanguage`   `\mtcselectlanguage{`⟨*language*⟩`}`

[1] Ce peut aussi être une option globale pour le document.

\ptctitle \plftitle \plttitle \mtctitle \mlftitle \mlttitle \stctitle \slftitle \slttitle

Chaque fichier .mld doit définir les neuf commandes suivantes (pour les mini-tables des matières, les mini-listes des figures et les mini-listes des tableaux, aux niveaux partie, chapitre et section) :

- \ptctitle
- \plftitle
- \plttitle
- \mtctitle
- \mlftitle
- \mlttitle
- \stctitle
- \slftitle
- \slttitle

De nombreux fichiers .mld ont besoin de fontes spéciales adéquates pour la langue correspondante ; comme ceci est une question concernant la langue, l'utilisateur doit mettre en place un contexte linguistique et typographique correct pour la langue correspondante, en utilisant par exemple le paquetage babel [9, 10, 11], le système CJK [29], le système ANTOMEGA [24], les paquetages ArabTeX [25, 26], BangTEX [36], ethiop [6], FarsiTEX (htp://www.farsitex.org), guarani [7], Malayalam [1], MonTEX [14, 15] ou ArmTEX [16]. Notez que c'est souvent le nom en *anglais* de la langue qui est utilisé pour nommer le fichier .mld correspondant.

Mais pour certaines langues orientales [2], le code source des titres utilise quelques *codages exotiques*, difficiles à manipuler dans un fichier .dtx, le fichier .mld est alors une simple enveloppe qui charge un fichier .mlo [3], qui n'est pas engendré par les fichiers .dtx dans la version actuelle du paquetage minitoc. Le codage d'entrée adéquat doit être mis en place par l'utilisateur *avant* le chargement du fichier .mld via \mtcselectlanguage.

filecontents

Pour contourner cette limitation, le fichier minitoc.ins utilise des environnements filecontents pour engendrer les fichiers .mlo.

8.1 Langue « acadian » : acadian.mld

La langue acadian (acadien) [4] est simplement du français, donc nous chargeons le fichier textttfrench.mld (voir la section 8.43 page 247) :

```
3991 ⟨*acadian⟩
3992 \ProvidesFile{acadian.mld}[2004/12/14]
3993 \mtcselectlanguage{french}%
3994 ⟨/acadian⟩
```

[2] Principalement pour les variantes du chinois, du farsi (iranien), du coréen (hangul et hanja), du japonais et du thaï.

[3] Le suffixe .mlo signifie *minitoc language object*, objet minitoc de langage.

[4] Parlée dans certaines parties du sud des États-Unis, comme la Louisiane.

8.2 Langue « acadien » : acadien.mld

La langue « acadien »[5] est simplement du français (« acadien » est le terme français pour « acadian », en anglais), donc nous chargons le fichier french.mld (voir la section 8.43 page 247) :

```
⟨*acadien⟩
\ProvidesFile{acadien.mld}[2004/12/14]
\mtcselectlanguage{french}%
⟨/acadien⟩
```

8.3 Langue « afrikaan » : afrikaan.mld

Les titres pour la langue « afrikaan »[6] sont tirés du paquetage babel [10] :

```
⟨*afrikaan⟩
\ProvidesFile{afrikaan.mld}[1999/03/16]
%% Afrikaan(s) titles for minitoc.sty
\def\ptctitle{Inhoudsopgawe}%
\def\plftitle{Lys van figure}%
\def\plttitle{Lys van tabelle}%
%%
\def\mtctitle{Inhoudsopgawe}%
\def\mlftitle{Lys van figure}%
\def\mlttitle{Lys van tabelle}%
%%
\def\stctitle{Inhoudsopgawe}%
\def\slftitle{Lys van figure}%
\def\slttitle{Lys van tabelle}%
⟨/afrikaan⟩
```

8.4 Langue « afrikaans » : afrikaans.mld

« Afrikaans » est un synonyme de « afrikaan », donc nous chargeons simplement afrikaan.mld (voir la section 8.3) :

```
⟨*afrikaans⟩
\ProvidesFile{afrikaans.mld}[2004/12/14]
\mtcselectlanguage{afrikaan}%
⟨/afrikaans⟩
```

[5] Parlée dans certaines parties du sud des États-Unis, comme la Louisiane.

[6] Parlée en Afrique du Sud, elle a des origines néerlandaises ; comparez avec la section 8.31 page 242.

8.5 Langue « americain » : american.mld

La langue « american » (américain) est comme « english »[7], donc nous chargeons simplement `english.mld` (voir la section 8.32 page 242) :

```
⟨*american⟩
\ProvidesFile{american.mld}[2004/12/14]
\mtcselectlanguage{english}%
⟨/american⟩
```

8.6 Langue « arab » : arab.mld

Les titres pour la langue « arab » (arabe) sont tirés du paquetage ArabTEX [25, 26], qui devrait être utilisé, avec les fontes associées.

```
⟨*arab⟩
\ProvidesFile{arab.mld}[1999/03/16]
%% Arabic titles for minitoc.sty
%% Needs arabic fonts (cf. documentation of arabtex)
%% (strings taken from arabtex; to be used with arabtex)
\def\ptctitle{al-mu.htawayAtu}
\def\plftitle{qA'imaTu a.s-.suwari}
\def\plttitle{qA'imaTu al-^gadAwili}
%%
\def\mtctitle{al-mu.htawayAtu}
\def\mlftitle{qA'imaTu a.s-.suwari}
\def\mlttitle{qA'imaTu al-^gadAwili}
%%
\def\stctitle{al-mu.htawayAtu}
\def\slftitle{qA'imaTu a.s-.suwari}
\def\slttitle{qA'imaTu al-^gadAwili}
⟨/arab⟩
```

8.7 Langue « arabic » : arabic.mld

« Arabic » est un synonyme pour « arab », donc nous chargeons simplement `arab.mld` (voir la section 8.6) :

```
⟨*arabic⟩
\ProvidesFile{arabic.mld}[2005/02/09]
\mtcselectlanguage{arab}%
⟨/arabic⟩
```

[7] Ceci devrait être vrai pour les titres des mini-tables ; les langues elles-mêmes ont quelques différences.

8.8 Langue « armenian » : armenian.mld

Les titres pour la langue « armenian » (arménien) sont tirés du paquetage ArmTEX package [16], qui devrait être utilisé, avec les fontes associées.

```
⟨*armenian⟩
\ProvidesFile{armenian.mld}[1999/06/28]
%% Armenian titles for minitoc.sty
\def\ptctitle{Bovandakuthyun}
\def\plftitle{Patkerneri cank}
\def\plttitle{Aghyusakneri cank}
%%
\def\mtctitle{Bovandakuthyun}
\def\mlftitle{Patkerneri cank}
\def\mlttitle{Aghyusakneri cank}
%%
\def\stctitle{Bovandakuthyun}
\def\slftitle{Patkerneri cank}
\def\slttitle{Aghyusakneri cank}
⟨/armenian⟩
```

8.9 Langue « austrian » : austrian.mld

En ce qui concerne les titres des mini-tables, la langue « austrian » (autrichien) est comme la langue « german » (allemand), donc nous chargeons german.mld (voir la section 8.48 page 249) :

```
⟨*austrian⟩
\ProvidesFile{austrian.mld}[2004/12/14]
\mtcselectlanguage{german}%
⟨/austrian⟩
```

8.10 Langue « bahasa » : bahasa.mld

Les titres des mini-tables pour la langue « bahasa » [8] sont tirés du paquetage babel [10]. Des fontes spécifiques sont nécessaires.

```
⟨*bahasa⟩
\ProvidesFile{bahasa.mld}[1999/03/16]
%% Bahasa titles for minitoc.sty
\def\ptctitle{Daftar Isi}%
\def\plftitle{Daftar Gambar}%
\def\plttitle{Daftar Tabel}%
```

[8] Parlée en Indonésie et en Malaisie, avec des prononciations différentes mais avec la même écriture.

```
%%
\def\mtctitle{Daftar Isi}%
\def\mlftitle{Daftar Gambar}%
\def\mlttitle{Daftar Tabel}%
%%
\def\stctitle{Daftar Isi}%
\def\slftitle{Daftar Gambar}%
\def\slttitle{Daftar Tabel}%
⟨/bahasa⟩
```

8.11 Langue « bangla » : bangla.mld

Les titres pour la langue « bangla » [9] sont tirés du paquetage bangtex [36] ; ils ont besoin de fontes spécifiques.

```
⟨*bangla⟩
\ProvidesFile{bangla.mld}[2002/03/14]
%% Bangla titles for minitoc.sty
%% Needs specific fonts
\def\ptctitle{suu\*c*i potRo}
\def\plftitle{cho\*b*ir ta\*l*ika}
\def\plttitle{cho\*k*er ta\*l*ika}
%%
\def\mtctitle{suu\*c*i}
\def\mlftitle{cho\*b*ir ta\*l*ika}
\def\mlttitle{cho\*k*er ta\*l*ika}
%%
\def\stctitle{suu\*c*i}
\def\slftitle{cho\*b*ir ta\*l*ika}
\def\slttitle{cho\*k*er ta\*l*ika}
⟨/bangla⟩
```

8.12 Langue « basque » : basque.mld

Les titres pour la langue « basque » [10] sont tirés du paquetage babel [10]. Il semble que des fontes spéciales soient nécessaires.

```
⟨*basque⟩
\ProvidesFile{basque.mld}[1999/12/06]
%% Basque titles for minitoc.sty
%% Needs special fonts
\def\ptctitle{Gaien Aurkibidea}
\def\plftitle{Irudien Zerrenda}
\def\plttitle{Taulen Zerrenda}
```

[9] Parlée au Bangladesh et dans certaines parties de l'Inde.

[10] Parlée au Pays Basque, qui est situé dans le nord de l'Espagne et le sud-ouest de la France.

```
%%
\def\mtctitle{Gaien Aurkibidea}
\def\mlftitle{Irudien Zerrenda}
\def\mlttitle{Taulen Zerrenda}
%%
\def\stctitle{Gaien Aurkibidea}
\def\slftitle{Irudien Zerrenda}
\def\slttitle{Taulen Zerrenda}
⟨/basque⟩
```

8.13 Langue « bicig » : bicig.mld

Les titres pour la langue « bicig » [11] sont tirés du paquetage MonTEX [14, 15]. Cette langue requiert des fontes spécifiques. Voir aussi la section 8.83 page 268.

```
⟨*bicig⟩
\ProvidesFile{bicig.mld}[1999/03/16]
%% needs mongol fonts
%% Mongol (Bicig) titles for minitoc.sty
\def\ptctitle{\bcg{GarciG}}
\def\plftitle{\bcg{zuraG-un zigsaalt}}
\def\plttitle{\bcg{k"usn"agti"in jagsaalt}}
%%
\def\mtctitle{\bcg{GarciG}}
\def\mlftitle{\bcg{zuraG-un zigsaalt}}
\def\mlttitle{\bcg{k"usn"agti"in jagsaalt}}
%%
\def\stctitle{\bcg{GarciG}}
\def\slftitle{\bcg{zuraG-un zigsaalt}}
\def\slttitle{\bcg{k"usn"agti"in jagsaalt}}
⟨/bicig⟩
```

8.14 Langue « brazil » : brazil.mld

Les titres pour la langue « brazil » (brésilien) [12] sont tirés du paquetage babel [10] :

```
⟨*brazil⟩
\ProvidesFile{brazil.mld}[1999/03/16]
%% Portugues (brazil) titles for minitoc.sty
\def\ptctitle{Sum\'ario}%
\def\plftitle{Lista de Figuras}%
\def\plttitle{Lista de Tabelas}%
%%
```

[11] Le « bicig » est une forme écrite de la langue mongole.

[12] C'est le dialecte portugais parlé au Brésil. Ces titres sont *différents* au Brésil et au Portugal. Voir la section 8.92 page 272.

```
\def\mtctitle{Sum\'ario}%
\def\mlftitle{Lista de Figuras}%
\def\mlttitle{Lista de Tabelas}%
%%
\def\stctitle{Sum\'ario}%
\def\slftitle{Lista de Figuras}%
\def\slttitle{Lista de Tabelas}%
〈/brazil〉
```

8.15 Langue « brazilian » : `brazilian.mld`

La langue « brazilian » est comme « brazil », donc nous chargeons simplement `brazil.mld` (voir la section 8.14 page précédente) :

```
〈*brazilian〉
\ProvidesFile{brazilian.mld}[2005/07/11]
\mtcselectlanguage{brazil}%
〈/brazilian〉
```

8.16 Langue « breton » : `breton.mld`

Les titres pour la langue « breton »[13] sont tirés du paquetage `babel` [10] :

```
〈*breton〉
\ProvidesFile{breton.mld}[1999/03/16]
%% Breton titles for minitoc.sty
\def\ptctitle{Taolenn}
\def\plftitle{Listenn ar Figurenno\`u}
\def\plttitle{Listenn an taolenno\`u}
%%
\def\mtctitle{Taolenn}
\def\mlftitle{Listenn ar Figurenno\`u}
\def\mlttitle{Listenn an taolenno\`u}
%%
\def\stctitle{Taolenn}
\def\slftitle{Listenn ar Figurenno\`u}
\def\slttitle{Listenn an taolenno\`u}
〈/breton〉
```

[13] Parlées comme dialecte celtique local en Bretagne (France).

8.17 Langue « british » : british.mld

La langue « british » est comme « english », donc nous chargeons simplement english.mld (voir la section 8.32 page 242) :

```
⟨*british⟩
\ProvidesFile{british.mld}[2005/07/11]
\mtcselectlanguage{english}%
⟨/british⟩
```

8.18 Langue « bulgarian » : bulgarian.mld

Les titres pour la langue « bulgarian » (bulgare) sont tirés du paquetage babel [10] ; ils ont besoin de fontes cyrilliques spécifiques. Voir aussi la section 8.19 page suivante.

```
⟨*bulgarian⟩
\ProvidesFile{bulgarian.mld}[2001/02/28]
%% Bulgarian titles for minitoc.sty
%% Needs special fonts
\def\ptctitle{%
   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
\def\plftitle{%
   {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ %
    \cyrn\cyra\ \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
\def\plttitle{%
   {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ %
    \cyrn\cyra\ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
%%
\def\mtctitle{%
   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
\def\mlftitle{%
   {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ %
    \cyrn\cyra\ \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
\def\mlttitle{%
   {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ %
    \cyrn\cyra\ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
%%
\def\stctitle{%
   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
\def\slftitle{%
   {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ %
    \cyrn\cyra\ \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
\def\slttitle{%
   {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ %
    \cyrn\cyra\ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
⟨/bulgarian⟩
```

8.19 Langue « bulgarianb » : bulgarianb.mld

Les titres pour la langue « bulgarianb » (haut bulgare) sont tirés du paquetage babel [10] ; Ils requièrent des fontes cyrilliques spécifiques. Voir aussi la section 8.18 page précédente.

```
⟨*bulgarianb⟩
\ProvidesFile{bulgariand.mld}[2005/03/24]
%% Upper bulgarian titles for minitoc.sty
%% Needs cyrillic fonts for upper bulgarian
\def\ptctitle{%
 {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
\def\plftitle{% Figuri
  {\cyr \CYRF\cyri\cyrg\cyru\cyrr\cyri}}%
\def\plttitle{% Tablici
  {\cyr \CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyri}}%
%%
\def\mtctitle{% Sydyrzhanie
 {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
\def\mlftitle{% Figurite
  {\cyr \CYRF\cyri\cyrg\cyru\cyrr\cyri}}%
\def\mlttitle{% Tablici
  {\cyr \CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyri}}%
%%
\def\stctitle{% Sydyrzhanie
 {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
\def\slftitle{% Figuri
  {\cyr \CYRF\cyri\cyrg\cyru\cyrr\cyri}}%
\def\slttitle{% Tablici
  {\cyr \CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyri}}%
⟨/bulgarianb⟩
```

8.20 Langue « buryat » : buryat.mld

Les titres pour la langue « buryat » [14] sont tirés du paquetage MonTEX package [14, 15]. Cette langue requiert des fontes spécifiques. Voir aussi la section 8.83 page 268.

```
⟨*buryat⟩
\ProvidesFile{buryat.mld}[1999/03/16]
%% Buryat titles for minitoc.sty
%% Needs special fonts
\def\ptctitle{{\mnr Gar{\sh}ag}}%
\def\plftitle{{\mnr Zuraga"i jagsaalt}}%
\def\plttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
%%
\def\mtctitle{{\mnr Gar{\sh}ag}}%
\def\mlftitle{{\mnr Zuraga"i jagsaalt}}%
```

[14] Parlée dans certaines régions de la Mongolie. dans la république Buryat, près du lac Baïkal.

```
\def\mlttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
%%
\def\stctitle{{\mnr Gar{\sh}ag}}%
\def\slftitle{{\mnr Zuraga"i jagsaalt}}%
\def\slttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
⟨/buryat⟩
```

8.21 Langue « canadian » : `canadian.mld`

La langue « canadian » (notez le « i*a*n » final) est simplement la langue *anglaise* parlée au Canada. Nous chargeons simplement le fichier `english.mld` (voir la section 8.32 page 242) :

```
⟨*canadian⟩
\ProvidesFile{canadian.mld}[2004/12/14]
\mtcselectlanguage{english}%
⟨/canadian⟩
```

8.22 Langue « canadien » : `canadien.mld`

La langue « canadien » (notez le « i*e*n » final) est simplement la langue *française* parlée au Canada. Nous chargeons simplement le fichier `french.mld` (voir la section 8.43 page 247) :

```
⟨*canadien⟩
\ProvidesFile{canadien.mld}[2004/12/14]
\mtcselectlanguage{french}%
⟨/canadien⟩
```

8.23 Langue « castillan » : `castillan.mld`

La langue « castillan » est plus connue sous le nom de « spanish » (espagnol), mais est parlée principalement en Castille, une partie du centre de l'Espagne. Nous chargeons simplement le fichier `spanish.mld` (voir la section 8.105 page 280) :

```
⟨*castillan⟩
%% Castillan (spanish) titles for minitoc.sty
\ProvidesFile{castillan.mld}[2004/12/14]
\mtcselectlanguage{spanish}%
⟨/castillan⟩
```

8.24 Langue « castillian » : castillian.mld

La langue « castillian » est plus connue sous le nom de « spanish » (espagnol), mais est parlée principalement en Castille, une partie du centre de l'Espagne. « Castillian » est le nom anglais du castillan. Nous chargeons simplement le fichier `spanish.mld` (voir la section 8.105 page 280) :

```
⟨*castillian⟩
\ProvidesFile{castillian.mld}[2005/07/01]
%% Castillian (spanish) titles for minitoc.sty
\mtcselectlanguage{spanish}%
⟨/castillian⟩
```

8.25 Langue « catalan » : catalan.mld

Les titres pour la langue « catalan » [15] sont tirés du paquetage `babel` [10] :

```
⟨*catalan⟩
\ProvidesFile{catalan.mld}[1999/03/16]
%% Catalan titles for minitoc.sty
\def\ptctitle{\'Index}%
\def\plftitle{\'Index de figures}%
\def\plttitle{\'Index de taules}%
%%
\def\mtctitle{\'Index}%
\def\mlftitle{Figures}%
\def\mlttitle{Taules}%
%%
\def\stctitle{\'Index}%
\def\slftitle{Figures}%
\def\slttitle{Taules}%
⟨/catalan⟩
```

8.26 Langue « chinese1 » : chinese1.mld et chinese1.mlo

Il y a plusieurs variantes pour la langue chinoise. La langue « chinese1 » (chinois première variante) utilise des titres pris dans le fichier `Bg5.cap` du système CJK [29]. Des fontes spéciales sont nécessaires, évidemment. Voir aussi la section 8.27 page suivante.

Les titres pour la langue « chinese1 » contiennent des caractères qui ne peuvent pas être engendrés facilement, donc nous chargeons le fichier `chinese1.mlo`.

[15] Parlée en Catalogne, la partie orientale de l'Espagne, autour de Barcelone.

```
⟨*chinese1⟩
\ProvidesFile{chinese1.mld}[2005/01/28]
%% From the file file Bg5.cap of the CJK package
%%   for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%%   created by Werner Lemberg <wl@gnu.org>
%% Version 4.5.2 (28-Mar-2003)
%%
%% Chinese captions
%%
%% character set: Big 5
%% encoding: Big 5
%%
\mtcloadmlo{chinese1}
⟨/chinese1⟩
```

8.27 Langue « chinese2 » : `chinese2.mld` et `chinese2.mlo`

La langue « chinese2 » (chinois seconde variante) utilise des titres pris dans le fichier `Bg5.cpx` dans le système CJK [29]. Des fontes spéciales sont nécessaires, évidemment. Voir aussi la section 8.26 page précédente.

Les titres pour la langue « chinese2 » contiennent des caractères qui ne peuvent pas être engendrés facilement, donc nous chargeons le fichier `chinese2.mlo`.

```
⟨*chinese2⟩
\ProvidesFile{chinese2.mld}[2005/01/28]
%% From the file Bg5.cpx of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Chinese captions
%%
%% character set: Big 5
%% encoding: Big 5
%%
%% preprocessed
%%
\mtcloadmlo{chinese2}
⟨/chinese2⟩
```

8.28 Langue « croatian » : `croatian.mld`

Les titres pour la langue « croatian » (croate) sont tirés du paquetage `babel` [10] :

```
⟨*croatian⟩
\ProvidesFile{croatian.mld}[1999/03/16]
%% Croatian titles for minitoc.sty
\def\ptctitle{Sadr\v{z}aj}%
\def\plftitle{Slike}%
\def\plttitle{Tablice}%
%%
\def\mtctitle{Sadr\v{z}aj}%
\def\mlftitle{Slike}%
\def\mlttitle{Tablice}%
%%
\def\stctitle{Sadr\v{z}aj}%
\def\slftitle{Slike}%
\def\slttitle{Tablice}%
⟨/croatian⟩
```

8.29 Langue « czech » : czech.mld

Les titres pour la langue « czech » (tchèque) sont tirés du paquetage babel [10] :

```
⟨*czech⟩
\ProvidesFile{czech.mld}[1999/03/16]
%% Czech titles for minitoc.sty
%% Warning: defines \w as a ring accent
\def\w#1{\accent'27 #1}
\def\ptctitle{Obsah}%
\def\plftitle{Seznam obr\'azk\w{u}}%
\def\plttitle{Seznam tabulek}%
%%
\def\mtctitle{Obsah}%
\def\mlftitle{Seznam obr\'azk\w{u}}%
\def\mlttitle{Seznam tabulek}%
%%
\def\stctitle{Obsah}%
\def\slftitle{Seznam obr\'azk\w{u}}%
\def\slttitle{Seznam tabulek}%
⟨/czech⟩
```

8.30 Langue « danish » : danish.mld

Les titres pour la langue « danish » (danois) sont tirés du paquetage babel [10] :

```
⟨*danish⟩
\ProvidesFile{danish.mld}[1999/03/16]
%% Danish titles for minitoc.sty
\def\ptctitle{Indholdsfortegnelse}%
```

```
\def\plftitle{Figurer}%
\def\plttitle{Tabeller}%
%%
\def\mtctitle{Indholdsfortegnelse}%
\def\mlftitle{Figurer}%
\def\mlttitle{Tabeller}%
%%
\def\stctitle{Indholdsfortegnelse}%
\def\slftitle{Figurer}%
\def\slttitle{Tabeller}%
⟨/danish⟩
```

8.31 Langue « dutch » : dutch.mld

Les titres pour la langue « dutch » (néerlandais) sont tirés du paquetage babel [10] :

```
⟨*dutch⟩
\ProvidesFile{dutch.mld}[1999/03/16]
% Dutch titles for minitoc.sty
\def\ptctitle{Inhoudsopgave}%
\def\plftitle{Lijst van figuren}%
\def\plttitle{Lijst van tabellen}%
%%
\def\mtctitle{Inhoudsopgave}%
\def\mlftitle{Lijst van figuren}%
\def\mlttitle{Lijst van tabellen}%
%%
\def\stctitle{Inhoudsopgave}%
\def\slftitle{Lijst van figuren}%
\def\slttitle{Lijst van tabellen}%
⟨/dutch⟩
```

8.32 Langue « english » : english.mld

Les titres pour la langue « english » (anglais) sont tirés du paquetage babel [10]. Voir aussi les sections 8.5 page 231, 8.17 page 236, 8.21 page 238, 8.111 page 282 et 8.113 page 283. *C'est la langue par défaut.*

```
⟨*english⟩
\ProvidesFile{english.mld}[1999/03/16]
%% English titles for minitoc.sty
\def\ptctitle{Table of Contents}
\def\plftitle{List of Figures}
\def\plttitle{List of Tables}
%%
\def\mtctitle{Contents}
```

```
\def\mlftitle{Figures}
\def\mlttitle{Tables}
%%
\def\stctitle{Contents}
\def\slftitle{Figures}
\def\slttitle{Tables}
⟨/english⟩
```

8.33 Langue « esperant » : esperant.mld

Les titres pour la langue « esperant » (esperanto) sont tirés du paquetage `babel` [10].

```
⟨*esperant⟩
\ProvidesFile{esperant.mld}[1999/03/16]
%% Esperanto titles for minitoc.sty
\def\ptctitle{Enhavo}%
\def\plftitle{Listo de figuroj}%
\def\plttitle{Listo de tabeloj}%
%%
\def\mtctitle{Enhavo}%
\def\mlftitle{Listo de figuroj}%
\def\mlttitle{Listo de tabeloj}%
%%
\def\stctitle{Enhavo}%
\def\slftitle{Listo de figuroj}%
\def\slttitle{Listo de tabeloj}%
⟨/esperant⟩
```

8.34 Langue « esperanto » : esperanto.mld

Les langues « esperanto » et « esperant » sont synonymes, donc nous chargeons simplement le fichier `esperant.mld` (voir la section 8.33) :

```
⟨*esperanto⟩
\ProvidesFile{esperanto.mld}[2004/12/14]
\mtcselectlanguage{esperant}%
⟨/esperanto⟩
```

8.35 Langue « estonian » : estonian.mld

Les titres pour la langue « estonian » (estonien) sont tirés du paquetage `babel` [10] :

```
⟨*estonian⟩
\ProvidesFile{estonian.mld}[1999/03/16]
%% Estonian titles for minitoc.sty
\def\ptctitle{Sisukord}%
\def\plftitle{Joonised}%
\def\plttitle{Tabelid}%
%%
\def\mtctitle{Sisukord}%
\def\mlftitle{Joonised}%
\def\mlttitle{Tabelid}%
%%
\def\stctitle{Sisukord}%
\def\slftitle{Joonised}%
\def\slttitle{Tabelid}%
⟨/estonian⟩
```

8.36 Langue « ethiopia » : ethiopia.mld

Les titres pour la langue « ethiopia » (éthiopien) sont tirés du paquetage ethiop [6]. Des fontes spécifiques sont nécessaires.

```
⟨*ethiopia⟩
\ProvidesFile{ethiopia.mld}[1999/03/16]
%% Needs special fonts
\def\ptctitle{yezate}%
\def\plftitle{%
    ya\eth@doaltchar{85}‘elo\eth@doaltchar{109}
    mAwe\eth@doaltchar{187}}%
\def\plttitle{%
    yasane\eth@doaltchar{176}ra\eth@doaltchar{149}
    mAwe\eth@doaltchar{187}}%
%%
\def\mtctitle{yezate}%
\def\mlftitle{%
    ya\eth@doaltchar{85}‘elo\eth@doaltchar{109}
    mAwe\eth@doaltchar{187}}%
\def\mlttitle{%
    yasane\eth@doaltchar{176}ra\eth@doaltchar{149}
    mAwe\eth@doaltchar{187}}%
%%
\def\stctitle{yezate}%
\def\slftitle{%
    ya\eth@doaltchar{85}‘elo\eth@doaltchar{109}
    mAwe\eth@doaltchar{187}}%
\def\slttitle{%
    yasane\eth@doaltchar{176}ra\eth@doaltchar{149}
    mAwe\eth@doaltchar{187}}%
⟨/ethiopia⟩
```

8.37 Langue « ethiopian » : ethiopian.mld

La langue « ethiopian » (éthiopien) est un simple synonyme pour la langue « ethiopia », donc nous chargeons simplement le fichier `ethiopia.mld` (voir la section 8.36 page précédente) :

```
⟨*ethiopian⟩
\ProvidesFile{ethiopian.mld}[2004/12/14]
%% Needs ethopian special fonts
\mtcselectlanguage{ethiopia}%
⟨/ethiopian⟩
```

8.38 Langue « farsi1 » : farsi1.mld et farsi1.mlo

Il y a plusieurs variantes pour la langue farsi, parlée en Iran. La langue « farsi1 » (farsi première variante) utilise des titres pris dans le fichier `farsi.sty` du système FarsiTEX (`http://www.farsitex.org`), par Dr Mohammad Ghodsi, Roozbeh Pournader, Hassan Abolhassani et al. Des fontes spéciales sont nécessaires, évidemment. Voir aussi la section 8.39.

Les titres pour la langue « farsi1 » contiennent des caractères qui ne peuvent pas être engendrés facilement, donc nous chargeons le fichier `farsi1.mlo`.

```
⟨*farsi1⟩
\ProvidesFile{farsi1.mld}[2005/09/13]
% From farsi.sty of the FarsiTeX project
% by Dr Mohammad Ghodsi,
% Roozbeh Pournader (roozbeh@sharif.edu),
% Hassan Abolhassani, and others.
% http://www.farsitex.org
\mtcloadmlo{farsi1}
⟨/farsi1⟩
```

8.39 Langue « farsi2 » : farsi2.mld et farsi2.mlo

Il y a plusieurs variantes pour la langue farsi, parlée en Iran. La langue « farsi2 » (farsi seconde variante) utilise des titres pris dans le fichier `farsi.sty` du système FarsiTEX (`http://www.farsitex.org`), par Dr Mohammad Ghodsi, Roozbeh Pournader, Hassan Abolhassani et al. Des fontes spéciales sont nécessaires, évidemment. Voir aussi la section 8.38.

Les titres pour la langue « farsi2 » contiennent des caractères qui ne peuvent pas être engendrés facilement, donc nous chargeons le fichier `farsi2.mlo`.

```
⟨*farsi2⟩
\ProvidesFile{farsi2.mld}[2005/09/13]
% From farsi.sty of the FarsiTeX project
% by Dr Mohammad Ghodsi,
% Roozbeh Pournader (roozbeh@sharif.edu),
% Hassan Abolhassani, and others.
% http://www.farsitex.org
\mtcloadmlo{farsi2}
⟨/farsi2⟩
```

8.40 Langue « finnish » : finnish.mld

Les titres pour la langue « finnish » (finnois) sont tirés du paquetage `babel` [10]. Voir aussi la section 8.41.

```
⟨*finnish⟩
\ProvidesFile{finnish.mld}[1999/03/16]
%% Finnish titles for minitoc.sty
\def\ptctitle{Sis\"alt\"o}%   /* Could be "Sis\"allys" as well */
\def\plftitle{Kuvat}%
\def\plttitle{Taulukot}%
%%
\def\mtctitle{Sis\"alt\"o}%   /* Could be "Sis\"allys" as well */
\def\mlftitle{Kuvat}%
\def\mlttitle{Taulukot}%
%%
\def\stctitle{Sis\"alt\"o}%   /* Could be "Sis\"allys" as well */
\def\slftitle{Kuvat}%
\def\slttitle{Taulukot}%
⟨/finnish⟩
```

8.41 Langue « finnish2 » : finnish2.mld

Les titres pour la langue « finnish2 » (finnois seconde variante) sont tirés d'une variante proposée dans le paquetage `babel` [10]. Voir aussi la section 8.40.

```
⟨*finnish2⟩
\ProvidesFile{finnish2.mld}[2005/04/06]
%% Finnish titles for minitoc.sty
\def\ptctitle{Sis\"allys}%
\def\plftitle{Kuvat}%
\def\plttitle{Taulukot}%
%%
\def\mtctitle{Sis\"allys}%
\def\mlftitle{Kuvat}%
\def\mlttitle{Taulukot}%
```

```
%%
\def\stctitle{Sis\"allys}%
\def\slftitle{Kuvat}%
\def\slttitle{Taulukot}%
</finnish2>
```

8.42 Langue « francais » : francais.mld

La langue « francais »[16] est synonyme de la langue « french », donc nous chargeons le fichier french.mld (voir la section 8.43) :

```
<*francais>
\ProvidesFile{francais.mld}[2004/12/14]
\mtcselectlanguage{french}%
</francais>
```

8.43 Langue « french » : french.mld

Les titres pour la langue « french » (français) sont tirés du paquetage babel [10]. Voir aussi les sections 8.1 page 229, 8.2 page 230, 8.22 page 238, 8.42, 8.44 page suivante, 8.45 page suivante et 8.46 page suivante.

```
<*french>
\ProvidesFile{french.mld}[2005/06/16]
%%
\def\ptctitle{Table des Mati\`eres}
\def\plftitle{Liste des Figures}
\def\plttitle{Liste des Tableaux}
%%
%% French titles for minitoc.sty
\def\mtctitle{Sommaire}
\def\mlftitle{Figures}
\def\mlttitle{Tableaux}
%%
\def\stctitle{Sommaire}
\def\slftitle{Figures}
\def\slttitle{Tableaux}
</french>
```

[16] L'orthographe correcte est « français », mais je n'ai pas osé utiliser une cédille dans un nom de fichier.

8.44 Langue « frenchb » : frenchb.mld

La langue « frenchb » est un synonyme pour la langue « french », donc nous chargeons le fichier french.mld. Voir la section 8.43 page précédente.

```
⟨*frenchb⟩
\ProvidesFile{frenchb.mld}[2003/02/11]
\mtcselectlanguage{french}%
⟨/frenchb⟩
```

8.45 Langue « frenchle » : frenchle.mld

La langue « frenchle » est un synonyme pour la langue « french », donc nous chargeons le fichier french.mld. Voir la section 8.43 page précédente.

```
⟨*frenchle⟩
\ProvidesFile{frenchle.mld}[2003/02/20]
\mtcselectlanguage{french}%
⟨/frenchle⟩
```

8.46 Langue « frenchpro » : frenchpro.mld

La langue « frenchpro » est un synonyme pour la langue « french », donc nous chargeons le fichier french.mld. Voir la section 8.43 page précédente.

```
⟨*frenchpro⟩
\ProvidesFile{frenchpro.mld}[2003/02/20]
\mtcselectlanguage{french}%
⟨/frenchpro⟩
```

8.47 Langue « galician » : galician.mld

Les titres pour la langue « galician » (galicien) [17] sont tirés du paquetage babel [10] :

```
⟨*galician⟩
\ProvidesFile{galician.mld}[1999/03/16]
%% Galician titles for minitoc.sty
%%
\def\ptctitle{\'Indice Xeral}%
```

[17] Parlée en Galice, dans la partie nord-ouest de l'Espagne, autour de Saint-Jacques de Compostelle.

```
\def\plftitle{\'Indice de Figuras}%
\def\plttitle{\'Indice de T\'aboas}%
%%
\def\mtctitle{\'Indice Xeral}%
\def\mlftitle{\'Indice de Figuras}%
\def\mlttitle{\'Indice de T\'aboas}%
%%
\def\stctitle{\'Indice Xeral}%
\def\slftitle{\'Indice de Figuras}%
\def\slttitle{\'Indice de T\'aboas}%
⟨/galician⟩
```

8.48 Langue « german » : german.mld

Les titres pour la langue « german » (allemand) sont tirés du paquetage babel [10]. Voir aussi la section 8.9 page 232.

```
⟨*german⟩
\ProvidesFile{german.mld}[1999/03/16]
%% German titles for minitoc.sty
\def\ptctitle{Inhaltsangabe}
\def\plftitle{Figuren}
\def\plttitle{Tabellen}
%%
\def\mtctitle{Inhaltsangabe}
\def\mlftitle{Figuren}
\def\mlttitle{Tabellen}
%%
\def\stctitle{Inhaltsangabe}
\def\slftitle{Figuren}
\def\slttitle{Tabellen}
⟨/german⟩
```

8.49 Langue « germanb » : germanb.mld

La langue « germanb » (allemand, variante) est une variante de la langue « german ». Les titres sont tirés du paquetage babel [10] :

```
⟨*germanb⟩
\ProvidesFile{germanb.mld}[1999/03/16]
%% German titles (variant) for minitoc.sty
\def\ptctitle{Inhaltsverzeichnis}%    % oder nur: Inhalt
\def\plftitle{Abbildungsverzeichnis}
\def\plttitle{Tabellenverzeichnis}
%%
\def\mtctitle{Inhaltsverzeichnis}%    % oder nur: Inhalt
```

```
\def\mlftitle{Abbildungsverzeichnis}
\def\mlttitle{Tabellenverzeichnis}
%%
\def\stctitle{Inhalt}%    % oder nur: Inhalt
\def\slftitle{Abbildungen}
\def\slttitle{Tabellen}
⟨/germanb⟩
```

8.50 Langue « greek » : greek.mld

Les titres pour la langue « greek » (grec moderne) sont tirés du paquetage babel [10]. Des fontes grecques sont requises.

```
⟨*greek⟩
\ProvidesFile{greek.mld}[1999/03/16]
%% greek.mld
%% Needs greek fonts.
\def\ptctitle{Perieq'omena}
\def\plftitle{Kat'alogoc Sqhm'atwn}
\def\plttitle{Kat'alogoc Pin'akwn}
%%
\def\mtctitle{Perieq'omena}
\def\mlftitle{Kat'alogoc Sqhm'atwn}
\def\mlttitle{Kat'alogoc Pin'akwn}
%%
\def\stctitle{Perieq'omena}
\def\slftitle{Kat'alogoc Sqhm'atwn}
\def\slttitle{Kat'alogoc Pin'akwn}
⟨/greek⟩
```

8.51 Langue « greek-mono » : greek-mono.mld

Les titres pour la langue « greek-mono » (grec dit monotonique[18]) sont tirés du projet ANTOMEGA [24] :

```
⟨*greek – mono⟩
\ProvidesFile{greek-mono.mld}[2005/02/08]
%% from omega-greek.ldf (Antomega project)
%% Needs Omega
%% Alexej M. Kryokov
%% Dmitry Ivanov
%%
\def\ptctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^03cc^^^^03bc%
```

[18]Le grec monotonique, selon une récente (1982) mais fortement constestée – et constestable – réforme de la langue grecque.

```
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\plftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^03ac^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^03ac^^^^03c4^^^^03c9%
 ^^^^03bd}}%
\def\plftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^03ac^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^03ac^^^^03ba^^^^03c9^^^^03bd}}%
%%
\def\mtctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^03cc^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\mlftitle%{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^03ac^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^03ac^^^^03c4^^^^03c9%
 ^^^^03bd}}%
\def\mlftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^03ac^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^03ac^^^^03ba^^^^03c9^^^^03bd}}%
%%
\def\stctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^03cc^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\slftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^03ac^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^03ac^^^^03c4^^^^03c9%
 ^^^^03bd}}%
\def\slftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^03ac^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^03ac^^^^03ba^^^^03c9^^^^03bd}}%
⟨/greek – mono⟩
```

8.52 Langue « greek-polydemo » : `greek-polydemo.mld`

Les titres pour la langue « greek-polydemo » [19] sont tirés du projet ANTOMEGA [24] :

```
⟨*greek – polydemo⟩
\ProvidesFile{greek-polydemo.mld}[2005/02/08]
%% from omega-greek.ldf (Antomega project)
%% Needs Omega
%% Alexej M. Kryokov
%% Dmitry Ivanov
%%
\def\ptctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^1f79^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\plftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^1f71^^^^03c4^^^^03c9%
```

[19] Grec polytonique « démotique » (populaire), pour le grec classique.

```
 ^^^^03bd}}%
\def\plttitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^1f71^^^^03ba^^^^03c9^^^^03bd}}%
%%
\def\mtctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^1f79^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\mlftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^1f71^^^^03c4^^^^03c9%
 ^^^^03bd}}%
\def\mlttitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^1f71^^^^03ba^^^^03c9^^^^03bd}}%
%%
\def\stctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^1f79^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\slftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^1f71^^^^03c4^^^^03c9%
 ^^^^03bd}}%
\def\slttitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^1f71^^^^03ba^^^^03c9^^^^03bd}}%
⟨/greek – polydemo⟩
```

8.53 Langue « greek-polykatha » : `greek-polykatha.mld`

Les titres pour la langue « greek-polykatha » [20] sont tirés du projet ANTOMEGA [24] :

```
⟨*greek – polykatha⟩
\ProvidesFile{greek-polykatha.mld}[2005/02/08]
%% from omega-greek.ldf (Antomega project)
%% Needs Omega
%% Alexej M. Kryokov
%% Dmitry Ivanov
%%
\def\ptctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^1f79^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\plftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^1f71^^^^03c4^^^^03c9%
 ^^^^03bd}}%
```

[20] Grec polytonique, en style « kathaverousa » (purifié), une forme de la langue grecque créée au début du dix-neuvième siècle par Adamantios Korais, pour purifier la langue du vocabulaire byzantin ou non grec. Il a maintenant été démodé par le grec démotique (populaire), mais a laissé d'importantes traces dans la langue grecque moderne.

```
\def\plttitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^1f71^^^^03ba^^^^03c9^^^^03bd}}%
%%
\def\mtctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^1f79^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\mlftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^1f71^^^^03c4^^^^03c9%
 ^^^^03bd}}%
\def\mlttitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^1f71^^^^03ba^^^^03c9^^^^03bd}}%
%%
\def\stctitle{\localgreek%
{^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^1f79^^^^03bc%
 ^^^^03b5^^^^03bd^^^^03b1}}%
\def\slftitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^1f71^^^^03c4^^^^03c9%
 ^^^^03bd}}%
\def\slttitle{\localgreek%
{^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
 ^^^^03c2 ^^^^03c0^^^^03b9^^^^03bd^^^^1f71^^^^03ba^^^^03c9^^^^03bd}}%
⟨/greek – polykatha⟩
```

8.54 Langue « guarani » : guarani.mld

La langue « guarani » est la langue principale parlée au Paraguay. Très souvent, un mélange de guarani et d'espagnol, appelé « jopara », est utilisé. Les titres sont tirés du fichier guarani.ldf de Javier Bezos [7]. Un codage d'entrée spécial (win-gn.def) est nécessaire. Ces fichiers sont disponibles sur les archives CTAN.

```
⟨*guarani⟩
\ProvidesFile{guarani.mld}[2005/08/26]
%% Guarani titles for minitoc.sty
%% from guarani.ldf by Javier Bezos.
%% Input encoding win-gn.def is needed.
%%
\def\ptctitle{\'Indice general}%
\def\plftitle{\'Indice de figuras}%
\def\plttitle{\'Indice de cuadros}%
%%
\def\mtctitle{\'Indice general}%
\def\mlftitle{\'Indice de figuras}%
\def\mlttitle{\'Indice de cuadros}%
%%
\def\stctitle{\'Indice general}%
\def\slftitle{\'Indice de figuras}%
```

```
\def\slttitle{\'Indice de cuadros}%
⟨/guarani⟩
```

8.55 Langue « hangul1 » : hangul1.mld et hangul1.mlo

Les titres pour la langue « hangul1 » (coréen, écriture hangul première variante) sont tirés du fichier `hangul.cap` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.56, 8.57 page suivante, 8.58 page suivante, 8.59 page 256 et 8.60 page 257.

Les titres pour la langue « hangul1 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `hangul1.mlo`.

```
⟨*hangul1⟩
\ProvidesFile{hangul1.mld}[2005/01/28]
%% From the file hangul.cap of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Hangul captions
%%
%% character set: KS X 1001:1992 (=KS C 5601-1992)
%% encoding: EUC (=Wansung)
%%
\mtcloadmlo{hangul1}
⟨/hangul1⟩
```

8.56 Langue « hangul2 » : hangul2.mld et hangul2.mlo

Les titres pour la langue « hangul2 » (coréen en écriture hangul, deuxième variante) sont tirés du fichier `hangul.cpx` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.55, 8.57 page suivante, 8.58 page suivante, 8.59 page 256 et 8.60 page 257.

Les titres pour la langue « hangul2 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `hangul2.mlo`.

```
⟨*hangul2⟩
\ProvidesFile{hangul2.mld}[2005/01/28]
%% From the file hangul.cpx of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
```

```
%% Version 4.5.2 (28-Mar-2003)
%%
%% Hangul captions
%%
%% character set: KS X 1001:1992 (=KS C 5601-1992)
%% encoding: EUC (=Wansung)
%%
%% preprocessed
%%
\mtcloadmlo{hangul2}
⟨/hangul2⟩
```

8.57 Langue « hangul3 » : hangul3.mld et hangul3.mlo

Les titres pour la langue « hangul3 » (coréen en écriture hangul, troisième variante) sont tirés du fichier hangul2.cap du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.55 page précédente, 8.56 page précédente, 8.58, 8.59 page suivante et 8.60 page 257.

Les titres pour la langue « hangul3 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons hangul3.mlo.

```
⟨*hangul3⟩
\ProvidesFile{hangul3.mld}[2005/01/28]
%% From the file hangul2.cap of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Hangul captions set 2
%%
%% character set: KS X 1001:1992 (=KS C 5601-1992)
%% encoding: EUC (=Wansung)
%%
\mtcloadmlo{hangul3}
⟨/hangul3⟩
```

8.58 Langue « hangul4 » : hangul4.mld et hangul4.mlo

Les titres pour la langue « hangul4 » (coréen en écriture hangul, quatrième variante) sont tirés du fichier hangul2.cpx du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.55 page précédente, 8.56 page précédente, 8.57, 8.59 page suivante et 8.60 page 257.

Les titres pour la langue « hangul4 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `hangul4.mlo`.

```
⟨*hangul4⟩
\ProvidesFile{hangul4.mld}[2005/01/28]
%% From the file hangul2.cpx of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Hangul captions set 2
%%
%% character set: KS X 1001:1992 (=KS C 5601-1992)
%% encoding: EUC (=Wansung)
%%
%% preprocessed
%%
\mtcloadmlo{hangul4}
⟨/hangul4⟩
```

8.59 Langue « hanja1 » : `hanja1.mld` et `hanja1.mlo`

Les titres pour la langue « hanja1 » (coréen en écriture ancienne hanja, première variante) sont tirés du fichier `hanja.cpx` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir ausi les sections 8.55 page 254, 8.56 page 254, 8.57 page précédente, 8.58 page précédente et 8.60 page suivante.

Les titres pour la langue « hanja1 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `hanja1.mlo`.

```
⟨*hanja1⟩
\ProvidesFile{hanja1.mld}[2005/01/28]
%% From the file hanja.cpx of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Hanja captions
%%
%% character set: KS X 1001:1992 (=KS C 5601-1992)
%% encoding: EUC (=Wansung)
%%
%% preprocessed
%%
\mtcloadmlo{hanja1}
⟨/hanja1⟩
```

8.60 Langue « hanja2 » : `hanja2.mld` et `hanja2.mlo`

Les titres pour la langue « hanja2 » (coréen en écriture ancienne hanja, seconde variante) sont tirés du fichier `hanja.cpx` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir ausi les sections 8.55 page 254, 8.56 page 254, 8.57 page 255, 8.58 page 255 et 8.59 page précédente.

Les titres pour la langue « hanja2 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `hanja2.mlo`.

```
⟨*hanja2⟩
\ProvidesFile{hanja2.mld}[2005/01/28]
%% From the file hanja.cap of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <a7971428@unet.univie.ac.at>
%%
%% Version 4.1.3 (20-Jun-1997)
%%
%% Hanja captions
%%
%% character set: KS X 1001:1992 (=KS C 5601-1992)
%% encoding: EUC (=Wansung)
%%
\mtcloadmlo{hanja2}
⟨/hanja2⟩
```

8.61 Langue « hebrew » : `hebrew.mld`

Les titres pour la langue « hebrew » (hébreu) sont tirés du paquetage ArabTeX [25, 26], qui devrait être utilisé, avec les fontes associées.

```
⟨*hebrew⟩
\ProvidesFile{hebrew.mld}[2001/02/28]
%% Hebrew titles for minitoc.sty
%% Need hebrew fonts (see arabtex documentation)
\def\ptctitle{\tav\vav\kaf\finalnun\
              \ayin\nun\yod\nun\yod\finalmem}%
\def\plftitle{\resh\shin\yod\mem\tav\
              \alef\yod\vav\resh\yod\finalmem}%
\def\plttitle{\resh\shin\yod\mem\tav\
              \tet\bet\lamed\alef\vav\tav}%
%%
\def\mtctitle{\tav\vav\kaf\finalnun\
              \ayin\nun\yod\nun\yod\finalmem}%
\def\mlftitle{\resh\shin\yod\mem\tav\
              \alef\yod\vav\resh\yod\finalmem}%
\def\mlttitle{\resh\shin\yod\mem\tav\
```

```
              \tet\bet\lamed\alef\vav\tav}%
%%
\def\stctitle{\tav\vav\kaf\finalnun\
              \ayin\nun\yod\nun\yod\finalmem}%
\def\slftitle{\resh\shin\yod\mem\tav\
              \alef\yod\vav\resh\yod\finalmem}%
\def\slttitle{\resh\shin\yod\mem\tav\
              \tet\bet\lamed\alef\vav\tav}%
⟨/hebrew⟩
```

8.62 Langue « hungarian » : `hungarian.mld`

La langue « hungarian » (hongrois) est synonyme de la langue « magyar », donc nous chargeons `magyar.mld`. Voir la section 8.78 page 265.

```
⟨*hungarian⟩
\ProvidesFile{hungarian.mld}[2004/12/14]
\mtcselectlanguage{magyar}%
⟨/hungarian⟩
```

8.63 Langue « icelandic » : `icelandic.mld`

Les titres pour la langue « icelandic » (islandais) sont tirés du paquetage `babel` [10] :

```
⟨*icelandic⟩
\ProvidesFile{icelandic.mld}[2001/02/28]
%% Icelandic titles for minitoc.sty
%% need inputenc with 8-bits encoding
\def\ptctitle{Efnisyfirlit}%
\def\plftitle{Myndaskrá}%
\def\plttitle{Töfluskrá}%
%%
\def\mtctitle{Efnisyfirlit}%
\def\mlftitle{Myndaskrá}%
\def\mlttitle{Töfluskrá}%
%%
\def\stctitle{Efnisyfirlit}%
\def\slftitle{Myndaskrá}%
\def\slttitle{Töfluskrá}%
⟨/icelandic⟩
```

8.64 Langue « interlingua » : interlingua.mld

Les titres pour la langue « interlingua » [21] sont tirés du paquetage babel [10] :

```
⟨*interlingua⟩
\ProvidesFile{interlingua.mld}[2001/07/04]
\def\ptctitle{Contento}
\def\plftitle{Lista de Figuras}
\def\plttitle{Lista de Tabellas}
%%
\def\mtctitle{Contento}
\def\mlftitle{Figuras}
\def\mlttitle{Tabellas}
%%
\def\stctitle{Contento}
\def\slftitle{Figuras}
\def\slttitle{Tabellas}
⟨/interlingua⟩
```

8.65 Langue « irish » : irish.mld

Les titres pour la langue « irish » (irlandais) sont tirés du paquetage babel [10] :

```
⟨*irish⟩
\ProvidesFile{irish.mld}[1999/03/16]
%% Irish titles for minitoc.sty
\def\ptctitle{Cl\'ar\'Abhair}
\def\plftitle{L\'ear\'aid\'{\i}}
\def\plttitle{T\'abla\'{\i}}
%%
\def\mtctitle{Cl\'ar\'Abhair}
\def\mlftitle{L\'ear\'aid\'{\i}}
\def\mlttitle{T\'abla\'{\i}}
%%
\def\stctitle{Cl\'ar\'Abhair}
\def\slftitle{L\'ear\'aid\'{\i}}
\def\slttitle{T\'abla\'{\i}}
⟨/irish⟩
```

8.66 Langue « italian » : italian.mld

Les titres pour la langue « italian » (italien) sont tirés du paquetage babel [10] :

[21] Une tentative pour une langue universelle.

```
⟨*italian⟩
\ProvidesFile{italian.mld}[1999/03/16]
%% Italian titles for minitoc.sty
\def\ptctitle{Contenuto}%
\def\plftitle{Elenco delle figure}%
\def\plttitle{Elenco delle tabelle}%
%%
\def\mtctitle{Contenuto}%
\def\mlftitle{Elenco delle figure}%
\def\mlttitle{Elenco delle tabelle}%
%%
\def\stctitle{Contenuto}%
\def\slftitle{Elenco delle figure}%
\def\slttitle{Elenco delle tabelle}%
⟨/italian⟩
```

8.67 Langue « japanese » : `japanese.mld` et `japanese.mlo`

Il y a plusieurs variantes pour les titres en japonais. Les titres pour une première variante de la langue « japanese » (japonais) ont été trouvés (par une recherche avec Google) sur le site Web du Professeur Toshiki KUMAZAWA [22]. Voir aussi les sections 8.68, 8.69 page suivante, 8.70 page 262 et 8.71 page 262.

Les titres pour la langue « japanese » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `japanese.mlo`.

```
⟨*japanese⟩
\ProvidesFile{japanese.mld}[1999/03/16]
%% Japanese titles for minitoc.sty
%% Needs japanese fonts (CJK) and special input encoding.
%% From Kumazawa Toshiki
%% kumazawa@biwako.shiga-u.ac.jp
%% http://www.biwako.shiga-u.ac.jp/sensei/kumazawa/tex/minitoc.html
\mtcloadmlo{japanese}
⟨/japanese⟩
```

8.68 Langue « japanese2 » : `japanese2.mld` et `japanese2.mlo`

Les titres pour la langue « japanese2 » (japonais, deuxième variante) sont tirés du fichier `JIS.cap` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.67, 8.69 page suivante, 8.70 page 262 et 8.71 page 262.

[22] `http:www.biwako.shiga-u.ac.jp/sensei/kumazawa/tex/minitoc.html`

Les titres pour la langue « japanese2 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `japanese2.mlo`.

```
⟨*japanese2⟩
\ProvidesFile{japanese2.mld}[2005/01/28]
%% From the file JIS.cap of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Japanese captions
%%
%% character set: JIS X 0208:1997 (or JIS X 0208-1990)
%% encoding: EUC
%%
\mtcloadmlo{japanese2}
⟨/japanese2⟩
```

8.69 Langue « japanese3 » : `japanese3.mld` et `japanese3.mlo`

Les titres pour la langue « japanese3 » (japonais, troisième variante) sont tirés du fichier `JIS.cpx` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.67 page précédente, 8.68 page précédente, 8.70 page suivante et 8.71 page suivante.

Les titres pour la langue « japanese3 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `japanese3.mlo`.

```
⟨*japanese3⟩
\ProvidesFile{japanese3.mld}[2005/01/28]
%% From the file JIS.cpx of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Japanese captions
%%
%% character set: JIS X 0208:1997 (or JIS X 0208-1990)
%% encoding: EUC
%%
%% preprocessed
%%
\mtcloadmlo{japanese3}
⟨/japanese3⟩
```

8.70 Langue « japanese4 » : `japanese4.mld` et `japanese4.mlo`

Les titres pour la langue « japanese4 » (japonais, quatrième variante) sont tirés du fichier `SJIS.cap` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.67 page 260, 8.68 page 260, 8.69 page précédente et 8.71.

Les titres pour la langue « japanese4 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `japanese4.mlo`.

```
⟨*japanese4⟩
\ProvidesFile{japanese4.mld}[2005/01/28]
% From the file SJIS.cap of the CJK package
% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
% created by Werner Lemberg <wl@gnu.org>
%
% Version 4.5.2 (28-Mar-2003)
%
% Japanese captions
%
% character set: JIS X 0208:1997 (or JIS X 0208-1990)
% encoding: SJIS
%
\mtcloadmlo{japanese4}
⟨/japanese4⟩
```

8.71 Langue « japanese5 » : `japanese5.mld` et `japanese.mlo`

Les titres pour la langue « japanese5 » (japonais, cinquième variante) sont tirés du fichier `SJIS.cpx` du système CJK [29]. Des fontes spéciales sont nécessaires, bien sûr. Voir aussi les sections 8.67 page 260, 8.68 page 260, 8.69 page précédente et 8.70.

Les titres pour la langue « japanese5 » contiennent des caractères qui ne peuvent pas être générés facilement, donc nous chargeons `japanese5.mlo`.

```
⟨*japanese5⟩
\ProvidesFile{japanese5.mld}[[2005/01/28]
%% From the file SJIS.cpx of the CJK package
%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
%% created by Werner Lemberg <wl@gnu.org>
%%
%% Version 4.5.2 (28-Mar-2003)
%%
%% Japanese captions
%%
%% character set: JIS X 0208:1997 (or JIS X 0208-1990)
%% encoding: SJIS
%%
%% preprocessed
```

```
%%
\mtcloadmlo{japanese5}
⟨/japanese5⟩
```

8.72 Langue « latin » : `latin.mld`

Les titres pour la langue « latin » sont tirés du paquetage `babel` [10]. Voir aussi la section 8.73.

```
⟨*latin⟩
\ProvidesFile{latin.mld}[2001/02/28]
%% Latin titles for minitoc.sty
\def\ptctitle{Index}%
\def\plftitle{Conspectus descriptionum}%
\def\plttitle{Conspectus tabularum}%
%%
\def\mtctitle{Index}%
\def\mlftitle{Conspectus descriptionum}%
\def\mlttitle{Conspectus tabularum}%
%%
\def\stctitle{Index}%
\def\slftitle{Conspectus descriptionum}%
\def\slttitle{Conspectus tabularum}%
⟨/latin⟩
```

8.73 Langue « latin2 » : `latin2.mld`

Les titres pour la langue « latin2 » (latin, variante abrégée) sont tirés du paquetage `babel` [10], mais ont été abrégés. Voir aussi la section 8.72.

```
⟨*latin2⟩
\ProvidesFile{latin2.mld}[2005/04/06]
%% Latin titles (short) for minitoc.sty
\def\ptctitle{Index}%
\def\plftitle{Conspectus descriptionum}%
\def\plttitle{Conspectus tabularum}%
%%
\def\mtctitle{Index}%
\def\mlftitle{Descriptiones}%
\def\mlttitle{Tabulae}%
%%
\def\stctitle{Index}%
\def\slftitle{Descriptiones}%
\def\slttitle{Tabulae}%
⟨/latin2⟩
```

8.74 Langue « latvian » : latvian.mld

Les titres pour la langue « latvian » [23] sont tirés du projet ANTOMEGA [24]. Voir aussi la section 8.75 :

```
⟨*latvian⟩
\ProvidesFile{latvian.mld}[2005/02/08]
%% from latvian.ldf (Antomega project)
%% Needs Omega
%% Alexej M. Kryokov
%% Dmitry Ivanov
%%
\def\ptctitle{\locallatvian{Saturs}}%
\def\plftitle{\locallatvian{Att^^^^0113lu saraksts}}%
\def\plttitle{\locallatvian{Tabulu saraksts}}%
%%
\def\mtctitle{\locallatvian{Saturs}}%
\def\mlftitle{\locallatvian{Att^^^^0113lu saraksts}}%
\def\mlttitle{\locallatvian{Tabulu saraksts}}%
%%
\def\stctitle{\locallatvian{Saturs}}%
\def\slftitle{\locallatvian{Att^^^^0113lu saraksts}}%
\def\slttitle{\locallatvian{Tabulu saraksts}}%
⟨/latvian⟩
```

8.75 Langue « letton » : letton.mld

La langue « letton » est un synonyme pour la langue « latvian », donc nous chargeons simplement latvian.mld. Voir la section 8.74.

```
⟨*letton⟩
\ProvidesFile{letton.mld}[2005/02/08]
\mtcselectlanguage{latvian}%
⟨/letton⟩
```

8.76 Langue « lithuanian » : lithuanian.mld

Les titres pour la langue « lithuanian » (lithuanien) sont tirés du paquetage babel [10] :

```
⟨*lithuanian⟩
\ProvidesFile{lithuanian.mld}[1999/07/29]
%% Lithuanian titles for minitoc.sty
\def\ptctitle{Turinys}
```

[23] « Latvien » est l'autre nom du « letton ».

```
\def\plftitle{Paveiksl\protect\c u s\protect\c ara\protect\v sas}
\def\plttitle{Lentel\protect\.es}
%%
\def\mtctitle{Turinys}
\def\mlftitle{Paveiksl\protect\c u s\protect\c ara\protect\v sas}
\def\mlttitle{Lentel\protect\.es}
%%
\def\stctitle{Turinys}
\def\slftitle{Paveiksl\protect\c u s\protect\c ara\protect\v sas}
\def\slttitle{Lentel\protect\.es}
⟨/lithuanian⟩
```

8.77 Langue « lsorbian » : lsorbian.mld

Les titres pour la langue « lsorbian » (bas sorabe) [24] sont tirés du paquetage babel [10]. Voir aussi la section 8.114 page 284.

```
⟨*lsorbian⟩
\ProvidesFile{lsorbian.mld}[1999/03/16]
%% Lower sorbian titles for minitoc.sty
\def\ptctitle{Wop\'simje\'se}
\def\plftitle{Zapiswobrazow}
\def\plttitle{Zapistabulkow}
%%
\def\mtctitle{Wop\'simje\'se}
\def\mlftitle{Zapiswobrazow}
\def\mlttitle{Zapistabulkow}
%%
\def\stctitle{Wop\'simje\'se}
\def\slftitle{Zapiswobrazow}
\def\slttitle{Zapistabulkow}
⟨/lsorbian⟩
```

8.78 Langue « magyar » : magyar.mld

Les titres pour la langue « magyar » sont tirés du paquetage babel [10]. Un synonyme de « magyar » est « hungarian » (hongrois) (voir la section 8.62 page 258). Voir aussi la section 8.79 page suivante.

```
⟨*magyar⟩
\ProvidesFile{magyar.mld}[1999/03/16]
% Magyar titles for minitoc.sty
```

[24] Le bas sorabe, le sorabe ou wende, est un membre du sous-groupe slave occidental des langues indo-européennes parlé en Basse-Lusace dans les *länder* allemands de Saxe et de Brandebourg. Les Sorabes sont des descendants des Wendes, nom germain des tribus slaves qui occupaient la zone entre l'Elbe et la Saale à l'ouest et l'Oder à l'est pendant la période médiévale (VIème siècle).

```
\def\ptctitle{Tratalom}%
\def\plftitle{\'Abr\'ak}%
\def\plttitle{T\'abl\'azatok}%
%%
\def\mtctitle{Tratalom}%
\def\mlftitle{\'Abr\'ak}%
\def\mlttitle{T\'abl\'azatok}%
%%
\def\stctitle{Tratalom}%
\def\slftitle{\'Abr\'ak}%
\def\slttitle{T\'abl\'azatok}%
⟨/magyar⟩
```

8.79 Langue « magyar2 » : magyar2.mld

Les titres pour la langue « magyar2 » sont tirés d'une variante proposée dans le paquetage babel [10]. Voir aussi la section 8.78 page précédente.

```
⟨*magyar2⟩
\ProvidesFile{magyar.mld}[2005/04/06]
%% Magyar titles for minitoc.sty (variant)
\def\ptctitle{Tratalom}%
\def\plftitle{\'Abr\'ak list\'aja}%
\def\plttitle{T\'abl\'azatok list\'aja}%
%%
\def\mtctitle{Tratalom}%
\def\mlftitle{\'Abr\'ak list\'aja}%
\def\mlttitle{T\'abl\'azatok list\'aja}%
%%
\def\stctitle{Tratalom}%
\def\slftitle{\'Abr\'ak list\'aja}%
\def\slttitle{T\'abl\'azatok list\'aja}%
⟨/magyar2⟩
```

8.80 Langue « malayalam-keli » : malayalam-keli.mld

Les titres pour la langue « malayalam » (malayalam, écriture *keli*) [25], avec les fontes « Keli », sont tirés du paquetage malayalam [1]. Cette langue requert des fontes spécifiques. Voir aussi les sections 8.81 page suivante et 8.82 page suivante.

```
⟨*malayalam – keli⟩
\ProvidesFile{malayalam-keli.mld}[2005/06/07]
%
```

[25] La langue Malayalam est parlée de la côte occidentale de Malabar jusqu'à l'extrême sud de l'Inde, principalement dans l'état de Kerala. C'est l'une des langues dravidiennes fortement liée au Tamoul. L'alphabet et l'écriture datent du huitième ou neuvième siècle.

```
% Malayalam: Keli fonts
%
\def\ptctitle{\mm \X{\<68>}\X{\<197>}\X{\<83>}\X{\<161>}\<119>}%
\def\plftitle{\mm \X{\<78>\<110>}\X{\<123>\<88>}\X{\<167>}\X{\<196>}}%
\def\plttitle{\mm \X{\<116>\<83>}\X{\<95>\<110>}\X{\<102>\<112>}\X{\<73>}\X{\<196>}}%
%
\def\mtctitle{\mm \X{\<68>}\X{\<197>}\X{\<83>}\X{\<161>}\<119>}%
\def\mlftitle{\mm \X{\<78>\<110>}\X{\<123>\<88>}\X{\<167>}\X{\<196>}}%
\def\mlttitle{\mm \X{\<116>\<83>}\X{\<95>\<110>}\X{\<102>\<112>}\X{\<73>}\X{\<196>}}%
%
\def\stctitle{\mm \X{\<68>}\X{\<197>}\X{\<83>}\X{\<161>}\<119>}%
\def\slftitle{\mm \X{\<78>\<110>}\X{\<123>\<88>}\X{\<167>}\X{\<196>}}%
\def\slttitle{\mm \X{\<116>\<83>}\X{\<95>\<110>}\X{\<102>\<112>}\X{\<73>}\X{\<196>}}%
⟨/malayalam – keli⟩
```

8.81 Langue « malayalam-rachana » : malayalam-rachana.mld

Les titres pour la langue « malayalam », avec les fontes traditionnelles « Rachana » (ancien *lipi*), sont tirés du paquetage malayalam [1]. Cette langue requiert des fontes spécifiques. Voir aussi les sections 8.80 page précédente et 8.82.

```
⟨*malayalam – rachana⟩
\ProvidesFile{malayalam-rachana.mld}[2005/06/07]
%
% Malayalam: Rachana fonts, traditionnal.
%
\def\ptctitle{\mm \X{\<68>}\X{\<201>}\X{\<83>}\X{\<183>}\<119>}%
\def\plftitle{\mm \X{\<78>\<111>}\X{{\C\<94>}}\X{\<186>}\X{\<179>}}%
\def\plttitle{\mm \X{\<117>\<83>}\X{\<95>\<111>}\X{{\F\<59>}}\X{\<73>}\X{\<179>}}%
%
\def\mtctitle{\mm \X{\<68>}\X{\<201>}\X{\<83>}\X{\<183>}\<119>}%
\def\mlftitle{\mm \X{\<78>\<111>}\X{{\C\<94>}}\X{\<186>}\X{\<179>}}%
\def\mlttitle{\mm \X{\<117>\<83>}\X{\<95>\<111>}\X{{\F\<59>}}\X{\<73>}\X{\<179>}}%
%
\def\stctitle{\mm \X{\<68>}\X{\<201>}\X{\<83>}\X{\<183>}\<119>}%
\def\slftitle{\mm \X{\<78>\<111>}\X{{\C\<94>}}\X{\<186>}\X{\<179>}}%
\def\slttitle{\mm \X{\<117>\<83>}\X{\<95>\<111>}\X{{\F\<59>}}\X{\<73>}\X{\<179>}}%
⟨/malayalam – rachana⟩
```

8.82 Langue « malayalam-rachana2 » : malayalam-rachana2.mld

Les titres pout la langue « malayalam », avec les fontes « Rachana » réformées (nouveau *lipi*), sont tirés du paquetage malayalam [1]. Cette langue requert des fontes spécifiques. Voir aussi les sections 8.80 page précédente et 8.81.

```
⟨*malayalam – rachana2⟩
```

```
\ProvidesFile{malayalam-rachana2.mld}[2005/06/07]
%
% Malayalam: Rachana fonts, reformed.
%
\def\ptctitle{\mm \X{\<68>}\X{\<201>}\X{\<83>}\X{\<183>}\<119>}
\def\plftitle{\mm \X{\<78>\<111>}\X{\<125>\<88>}\X{\<186>}\X{\<179>}}
\def\plttitle{\mm \X{\<117>\<83>}\X{\<95>\<111>}\X{\<106>\<113>}\X{\<73>}\X{\<179>}}
%
\def\mtctitle{\mm \X{\<68>}\X{\<201>}\X{\<83>}\X{\<183>}\<119>}
\def\mlftitle{\mm \X{\<78>\<111>}\X{\<125>\<88>}\X{\<186>}\X{\<179>}}
\def\mlttitle{\mm \X{\<117>\<83>}\X{\<95>\<111>}\X{\<106>\<113>}\X{\<73>}\X{\<179>}}
%
\def\stctitle{\mm \X{\<68>}\X{\<201>}\X{\<83>}\X{\<183>}\<119>}
\def\slftitle{\mm \X{\<78>\<111>}\X{\<125>\<88>}\X{\<186>}\X{\<179>}}
\def\slttitle{\mm \X{\<117>\<83>}\X{\<95>\<111>}\X{\<106>\<113>}\X{\<73>}\X{\<179>}}
⟨/malayalam – rachana2⟩
```

8.83 Langue « mongol » : mongol.mld

Les titres pour la langue « mongol » sont tirés du paquetage MonTEX package [14, 15]. Cette langue requiert des fontes spécifiques. Voir aussi les sections 8.13 page 234 et 8.20 page 237.

```
⟨*mongol⟩
\ProvidesFile{mongol.mld}[1999/03/16]
%% Mongol (xalx) titles for minitoc.sty
%% Needs mongol fonts
\def\ptctitle{{\mnr Garqig}}
\def\plftitle{{\mnr Zurgi"in jagsaalt}}
\def\plttitle{{\mnr X"usn"agti"in jagsaalt}}
%%
\def\mtctitle{{\mnr Garqig}}
\def\mlftitle{{\mnr Zurgi"in jagsaalt}}
\def\mlttitle{{\mnr X"usn"agti"in jagsaalt}}
%%
\def\stctitle{{\mnr Garqig}}
\def\slftitle{{\mnr Zurgi"in jagsaalt}}
\def\slttitle{{\mnr X"usn"agti"in jagsaalt}}
⟨/mongol⟩
```

8.84 Langue « naustrian » : naustrian.mld

La langue « naustrian » (néo-autrichien) est un synonyme de la langue « ngermanb » (néo-allemand) (une version réformée de la variante germanb de la langue allemande), donc nous chargeons ngermanb.mld. Voir aussi la section 8.86 page suivante.

```
⟨*naustrian⟩
\ProvidesFile{naustrian.mld}[2004/12/14]
\mtcselectlanguage{ngermanb}%
⟨/naustrian⟩
```

8.85 Langue « ngerman » : ngerman.mld

La langue « ngerman » (néo-allemand) est un synonyme de la langue « ngermanb »[26], donc nous chargeons `ngermanb.mld`. Voir aussi la section 8.86.

```
⟨*ngerman⟩
\ProvidesFile{ngerman.mld}[2004/12/14]
\mtcselectlanguage{ngermanb}%
⟨/ngerman⟩
```

8.86 Langue « ngermanb » : ngermanb.mld

Les titres pour la langue « ngermanb »[27] sont tirés du paquetage `babel` [10]. Voir aussi les sections 8.84 page précédente et 8.85.

```
⟨*ngermanb⟩
\ProvidesFile{ngermanb.mld}[1999/12/06]
%% New german (B) titles for minitoc.sty
\def\ptctitle{Inhaltsverzeichnis}%    % oder nur: Inhalt
\def\plftitle{Abbildungsverzeichnis}%
\def\plttitle{Tabellenverzeichnis}%
%%
\def\mtctitle{Inhalt}%
\def\mlftitle{Abbildungsverzeichnis}%
\def\mlttitle{Tabellenverzeichnis}%
%%
\def\stctitle{Inhalt}%
\def\slftitle{Abbildungsverzeichnis}%
\def\slttitle{Tabellenverzeichnis}%
⟨/ngermanb⟩
```

[26] Une version réformée de la variante `germanb` de la langue allemande.
[27] Une variante pour la langue allemande, avec une orthographe réformée.

8.87 Langue « norsk » : norsk.mld

Les titres pour la langue « norsk » (norvégien, ou « bokmål », langue du royaume) sont tirés du paquetage babel [10], avec l'aide de Dag Langmyhr. Voir aussi la section 8.88.

```
⟨*norsk⟩
\ProvidesFile{norsk.mld}[1999/03/16]
%% Norsk titles for minitoc.sty
%% Thanks to Dag Langmyhr (dag@ifi.uio.no)
\def\ptctitle{Innhold}%
\def\plftitle{Figurer}% or Figurliste
\def\plttitle{Tabeller}% or Tabelliste
%%
\def\mtctitle{Innhold}%
\def\mlftitle{Figurer}% or Figurliste
\def\mlttitle{Tabeller}% or Tabelliste
%%
\def\stctitle{Innhold}%
\def\slftitle{Figurer}% or Figurliste
\def\slttitle{Tabeller}% or Tabelliste
⟨/norsk⟩
```

8.88 Langue « nynorsk » : nynorsk.mld

Les titres pour la langue « nynorsk » (néo-norvégien)[28] sont tirés du paquetage babel [10], avec l'aide de Dag Langmyhr. Voir aussi la section 8.87.

```
⟨*nynorsk⟩
\ProvidesFile{nynorsk.mld}[1999/03/16]
%% Nynorsk titles for minitoc.sty
%% Thanks to Dag Langmyhr (dag@ifi.uio.no)
\def\mtctitle{Innhald}%
\def\mlftitle{Figurar}% or Figurliste
\def\mlttitle{Tabellar}% or Tabelliste
%%
\def\ptctitle{Innhald}%
\def\plftitle{Figurar}% or Figurliste
\def\plttitle{Tabellar}% or Tabelliste
%%
\def\stctitle{Innhald}%
\def\slftitle{Figurar}% or Figurliste
\def\slttitle{Tabellar}% or Tabelliste
⟨/nynorsk⟩
```

[28] Créée vers 1800 par Ivar Åssen pour construire une vraie langue norvégienne indépendante et nationale, en réaction au danois, à partir des divers dialectes parlés dans le pays.

8.89 Langue « polish » : polish.mld

Les titres pour la langue « polish » (polonais) sont tirés du paquetage babel [10]. Voir aussi la section 8.90.

```
⟨*polish⟩
\ProvidesFile{polish.mld}[1999/03/16]
%% Polish titles for minitoc.sty
\def\ptctitle{Spis rzeczy}%
\def\plftitle{Spis rysunk\'ow}%
\def\plttitle{Spis tablic}%
%%
\def\mtctitle{Spis rzeczy}%
\def\mlftitle{Spis rysunk\'ow}%
\def\mlttitle{Spis tablic}%
%%
\def\stctitle{Spis rzeczy}%
\def\slftitle{Spis rysunk\'ow}%
\def\slttitle{Spis tablic}%
⟨/polish⟩
```

8.90 Langue « polish2 » : polish2.mld

Les titres pour la langue « polish2 » (polonais, seconde variante) sont tirés du projet ANTOMEGA [24]. Voir aussi la section 8.89.

```
⟨*polish2⟩
\ProvidesFile{polish2.mld}[2005/02/08]
%% from omega-polish.ldf (Antomega project)
%% Needs Omega
%% Alexej M. Kryokov
%% Dmitry Ivanov
%%
\def\ptctitle{\localpolish{Spis tre^^^^00b1ci}}%
\def\plftitle{\localpolish{Spis rysunk^^^^00adw}}%
\def\plttitle{\localpolish{Spis tablic}}%
%%
\def\mtctitle{\localpolish{Spis tre^^^^00b1ci}}%
\def\mlftitle{\localpolish{Spis rysunk^^^^00adw}}%
\def\mlttitle{\localpolish{Spis tablic}}%
%%
\def\stctitle{\localpolish{Spis tre^^^^00b1ci}}%
\def\slftitle{\localpolish{Spis rysunk^^^^00adw}}%
\def\slttitle{\localpolish{Spis tablic}}%
⟨/polish2⟩
```

8.91 Langue « portuges » language : portuges.mld

C'est une autre orthographe pour « portuguese » (voir la section 8.92), donc nous chargeons simplement `portuguese.mld` :

```
⟨*portuges⟩
\ProvidesFile{portuges.mld}[2005/06/06]
\mtcselectlanguage{portuguese}
⟨/portuges⟩
```

8.92 Langue « portuguese » : portuguese.mld

Les titres pour la langue « portuguese » (portugais) sont tirés du paquetage `babel` [10]. Voir aussi la section 8.14 page 234, parce que les titres sont différents au Brésil, même si l'on y parle aussi le portugais.

```
⟨*portuguese⟩
\ProvidesFile{portuguese.mld}[2005/07/08]
%% Portuguese titles for minitoc.sty
\def\ptctitle{Conte\'udo}%
\def\plftitle{Lista de Figuras}%
\def\plttitle{Lista de Tabelas}%
%%
\def\mtctitle{Conte\'udo}%
\def\mlftitle{Lista de Figuras}%
\def\mlttitle{Lista de Tabelas}%
%%
\def\stctitle{Conte\'udo}%
\def\slftitle{Lista de Figuras}%
\def\slttitle{Lista de Tabelas}%
⟨/portuguese⟩
```

8.93 Langue « romanian » : romanian.mld

Les titres pour la langue « romanian » (roumain) sont tirés du paquetage `babel` [10].

```
⟨*romanian⟩
\ProvidesFile{romanian.mld}[1999/03/16]
%% Romanian titles for minitoc.sty
\def\ptctitle{Cuprins}%
\def\plftitle{List\u{a} de figuri}%
\def\plttitle{List\u{a} de tabele}%
%%
\def\mtctitle{Cuprins}%
```

```
\def\mlftitle{List\u{a} de figuri}%
\def\mlttitle{List\u{a} de tabele}%
%%
\def\stctitle{Cuprins}%
\def\slftitle{List\u{a} de figuri}%
\def\slttitle{List\u{a} de tabele}%
⟨/romanian⟩
```

8.94 Langue « russian » : russian.mld

Les titres pour la langue « russian » (russe) sont tirés du paquetage babel [10]. Des fontes cyrilliques spécifiques sont requises.

```
⟨*russian⟩
\ProvidesFile{russian.mld}[1999/03/16]
%% Russian titles for minitoc.sty
\def\ptctitle{Oglavlenie}%
\def\plftitle{Pere{\cz}en{\mz} risunkov}%
\def\plttitle{Pere{\cz}en{\mz} tablic}%
%%
\def\mtctitle{Oglavlenie}%
\def\mlftitle{Pere{\cz}en{\mz} risunkov}%
\def\mlttitle{Pere{\cz}en{\mz} tablic}%
%%
\def\stctitle{Oglavlenie}%
\def\slftitle{Pere{\cz}en{\mz} risunkov}%
\def\slttitle{Pere{\cz}en{\mz} tablic}%
⟨/russian⟩
```

8.95 Langue « russian2m » : russian2m.mld

Les titres pour la langue « russian2m » (une variante moderne de « russian ») sont tirés du projet ANTOMEGA [24]. Des fontes cyrilliques spécifiques sont requises. Voir aussi la section 8.94.

```
⟨*russian2m⟩
\ProvidesFile{russian2m.mld}[2005/02/08]
%% from russian2m.ldf (Antomega project, russian modern)
%% Needs Omega
%% Alexej M. Kryokov
%% Dmitry Ivanov
%%
%% Needs cyrillic fonts
%%
\def\ptctitle{\localrussian%
{^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
```

```
 ^^^^0438^^^^0435}}%
\def\plftitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a ^^^^0438^^^^043b%
 ^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446^^^^0438%
 ^^^^0439}}%
\def\plttitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a ^^^^0442^^^^0430%
 ^^^^0431^^^^043b^^^^0438^^^^0446}}%
%%
\def\mtctitle{\localrussian%
{^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
 ^^^^0438^^^^0435}}%
\def\mlftitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a ^^^^0438^^^^043b%
 ^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446^^^^0438%
 ^^^^0439}}%
\def\mlttitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a ^^^^0442^^^^0430%
 ^^^^0431^^^^043b^^^^0438^^^^0446}}%
%%
\def\stctitle{\localrussian%
{^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
 ^^^^0438^^^^0435}}%
\def\slftitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a ^^^^0438^^^^043b%
 ^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446^^^^0438%
 ^^^^0439}}%
\def\slttitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a ^^^^0442^^^^0430%
 ^^^^0431^^^^043b^^^^0438^^^^0446}}%
⟨/russian2m⟩
```

8.96 Langue « russian2o » : `russian2o.mld`

Les titres pour la langue « russian2o » (« russian2o » est une ancienne variante de « russian ») sont tirés du projet ANTOMEGA [24]. Des fontes cyrilliques spécifiques sont requises. Voir aussi la section 8.94 page précédente.

```
⟨*russian2o⟩
\ProvidesFile{russian2o.mld}[2005/02/08]
%% from russian2o.mld (Antomega project - russian old)
%% Needs Omega
%% Alexej M. Kryokov
%% Dmitry Ivanov
%%
\def\ptctitle{\localrussian%
{^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
 ^^^^0456^^^^0435}}%
\def\plftitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a ^^^^0438%
```

```
 ^^^^043b^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446%
 ^^^^0456^^^^0439}}%
\def\plttitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a ^^^^0442%
 ^^^^0430^^^^0431^^^^043b^^^^0438^^^^0446^^^^044a}}%
%%
\def\mtctitle{\localrussian%
{^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
 ^^^^0456^^^^0435}}%
\def\mlftitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a ^^^^0438%
 ^^^^043b^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446%
 ^^^^0456^^^^0439}}%
\def\mlttitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a ^^^^0442%
 ^^^^0430^^^^0431^^^^043b^^^^0438^^^^0446^^^^044a}}%
%
\def\stctitle{\localrussian%
{^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
 ^^^^0456^^^^0435}}%
\def\slftitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a ^^^^0438%
 ^^^^043b^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446%
 ^^^^0456^^^^0439}}%
\def\slttitle{\localrussian%
{^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a ^^^^0442%
 ^^^^0430^^^^0431^^^^043b^^^^0438^^^^0446^^^^044a}}%
⟨/russian2o⟩
```

8.97 Langue « russianb » : `russianb.mld`

Les titres pour la langue « russianb » (« russianb » est une variante de « russian ») sont tirés du paquetage `babel` [10]. Des fontes cyrilliques spécifiques sont requises. Voir aussi la section 8.94 page 273.

```
⟨*russianb⟩
\ProvidesFile{russianb.mld}[1999/03/16]
%% russianb.mld
\def\ptctitle{%
  {\cyr \CYRO\CYRg\CYRl\CYRa\CYRv\CYRl\CYRe\CYRn\CYRi\CYRe}}%
\def\plftitle{%
  {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
    \CYRi\CYRl\CYRl\CYRyu\CYRs\CYRt\CYRr\CYRa\CYRc\CYRi\CYRishrt}}%
\def\plttitle{%
  \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
  \CYRt\CYRa\CYRb\CYRl\CYRi\CYRc}%
}
%%
\def\mtctitle{%
  {\cyr \CYRO\CYRg\CYRl\CYRa\CYRv\CYRl\CYRe\CYRn\CYRi\CYRe}}%
```

```
\def\mlftitle{%
  {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
    \CYRi\CYRl\CYRl\CYRyu\CYRs\CYRt\CYRr\CYRa\CYRc\CYRi\CYRishrt}}%
\def\mlttitle{%
  \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
  \CYRt\CYRa\CYRb\CYRl\CYRi\CYRc}%
}
%%
\def\stctitle{%
  {\cyr \CYRO\CYRg\CYRl\CYRa\CYRv\CYRl\CYRe\CYRn\CYRi\CYRe}}%
\def\slftitle{%
  {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
    \CYRi\CYRl\CYRl\CYRyu\CYRs\CYRt\CYRr\CYRa\CYRc\CYRi\CYRishrt}}%
\def\slttitle{%
  \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
  \CYRt\CYRa\CYRb\CYRl\CYRi\CYRc}%
}
⟨/russianb⟩
```

8.98 Langue « russianc » : russianc.mld

Les titres pour la langue « russianc » (« russianc » est une variante du russe « russian », utilisée dans la partie de la Mongolie sous influence russe) sont tirés du fichier `russian.def` dans le paquetage MonTEX [14, 15]. Des fontes cyrilliques spécifiques sont requises. Voir aussi la section 8.94 page 273.

```
⟨*russianc⟩
\ProvidesFile{russianc.mld}[1999/03/16]
%% Needs cyrillic fonts
%% Russian titles for minitoc.sty
%% Needs cyrillic fonts
\def\ptctitle{\xalx{Oglawlenie}}
\def\plftitle{\xalx{Spisok risunkow}}
\def\plttitle{\xalx{Spisok tablic}}
%%
\def\mtctitle{\xalx{Soderjanie}}
\def\mlftitle{\xalx{Spisok risunkow}}
\def\mlttitle{\xalx{Spisok tablic}}
%%
\def\stctitle{\xalx{Soderjanie}}
\def\slftitle{\xalx{Spisok risunkow}}
\def\slttitle{\xalx{Spisok tablic}}
⟨/russianc⟩
```

8.99 Langue « samin » : samin.mld

Les titres pour la langue « samin » (Sami du Nord) [29] sont tirés du fichier `samin.dtx` du paquetage `babel` [10]. Des fontes spécifiques sont requises.

```
⟨*samin⟩
\ProvidesFile{samin.mld}[2001/02/28]
%% Samin titles for minitoc.sty
\def\ptctitle{Sisdoallu}%
\def\plftitle{Govvosat}%
\def\plttitle{Tabeallat}%
%%
\def\mtctitle{Sisdoallu}%
\def\mlftitle{Govvosat}%
\def\mlttitle{Tabeallat}%
%%
\def\stctitle{Sisdoallu}%
\def\slftitle{Govvosat}%
\def\slttitle{Tabeallat}%
⟨/samin⟩
```

8.100 Langue « scottish » : scottish.mld

Les titres pour la langue « scottish » (écossais) sont tirés du paquetage `babel` [10] :

```
⟨*scottish⟩
\ProvidesFile{scottish.mld}[1999/03/16]
%%Scottish titles for minitoc.sty
\def\ptctitle{Cl\‘ar-obrach}
\def\plftitle{LiostaDhealbh}
\def\plttitle{LiostaChl\‘ar}
%%
\def\mtctitle{Cl\‘ar-obrach}
\def\mlftitle{LiostaDhealbh}
\def\mlttitle{LiostaChl\‘ar}
%%
\def\stctitle{Cl\‘ar-obrach}
\def\slftitle{LiostaDhealbh}
\def\slttitle{LiostaChl\‘ar}
⟨/scottish⟩
```

[29] Plusieurs dialectes ou langues Sami sont parlés en Finlande, Norvège, Suède et dans la péninsule de Kola (Russie). Les alphabets sont différents, donc il y aurait éventuellement besoin de plusieurs fichiers `.dtx` pour, par exemple, le Lule et le Sami du Sud. D'où le nom `samin.dtx` (et non pas `sami.dtx` ou analogue dans le cas du Sami du Nord. Cette note est tirée du fichier `samin.dtx`.

8.101 Langue « serbian » : serbian.mld

Les titres pour la langue « serbian » (serbe) sont tirés du paquetage babel [10]. Voir aussi la section 8.102.

```
⟨*serbian⟩
\ProvidesFile{serbian.mld}[1999/12/06]
%%Serbian titles for minitoc.sty
\def\ptctitle{Sadr\v{z}aj}%
\def\plftitle{Slike}%
\def\plttitle{Tabele}%
%%
\def\mtctitle{Sadr\v{z}aj}%
\def\mlftitle{Slike}%
\def\mlttitle{Tabele}%
%%
\def\stctitle{Sadr\v{z}aj}%
\def\slftitle{Slike}%
\def\slttitle{Tabele}%
⟨/serbian⟩
```

8.102 Langue « serbianc » : serbianc.mld

Les titres pour la langue « serbianc » (serbe cyrillique) [30] ont été donnés par Marko ÈEHAJA et Frank KÜSTER. Des fontes cyrilliques sont requises. Voir aussi la section 8.101.

```
⟨*serbianc⟩
\ProvidesFile{serbianc.mld}[2001/11/25]
%% Provides titles for minitoc.sty in Serbian Cyrillic
%%
%%Marko Èehaja Internut@Thetaworld.Org
%%Frank Küster, Biozentrum der Univ. Basel, frank@kuesterei.ch
%%Abt. Biophysikalische Chemie
\def\ptctitle{%
  {\cyr\CYRS\cyra\cyrd\cyrr\cyrzh\cyra\cyrje}}%
\def\plftitle{%
  {\cyr\CYRS\cyrl\cyri\cyrk\cyre}}%
\def\plttitle{\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyre}%
%%
\def\mtctitle{%
  {\cyr\CYRS\cyra\cyrd\cyrr\cyrzh\cyra\cyrje}}%
\def\mlftitle{%
  {\cyr\CYRS\cyrl\cyri\cyrk\cyre}}%
\def\mlttitle{\cyr\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyre}%
%%
\def\stctitle{%
  {\cyr\CYRS\cyra\cyrd\cyrr\cyrzh\cyra\cyrje}}%
```

[30] La langue « serbianc » est écrite en caractères cyrilliques.

```
\def\slftitle{%
  {\cyr\CYRS\cyrl\cyri\cyrk\cyre}}%
\def\sltitle{\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyre}%
⟨/serbianc⟩
```

8.103 Langue « slovak » : `slovak.mld`

Les titres pour la langue « slovak » (slovaque) sont tirés du paquetage `babel` [10] :

```
⟨*slovak⟩
\ProvidesFile{slovak.mld}[1999/03/16]
%% Slovak titles for minitoc.sty
\def\ptctitle{Obsah}%
\def\plftitle{Zoznam obr\'azkov}%
\def\plttitle{Zoznam tabuliek}%
%%
\def\mtctitle{Obsah}%
\def\mlftitle{Zoznam obr\'azkov}%
\def\mlttitle{Zoznam tabuliek}%
%%
\def\stctitle{Obsah}%
\def\slftitle{Zoznam obr\'azkov}%
\def\slttitle{Zoznam tabuliek}%
⟨/slovak⟩
```

8.104 Langue « slovene » : `slovene.mld`

Les titres pour la langue « slovene » (slovène) sont tirés du paquetage `babel` [10] :

```
⟨*slovene⟩
\ProvidesFile{slovene.mld}[1999/03/16]
%% Slovene titles for minitoc.sty
\def\ptctitle{Kazalo}%
\def\plftitle{Slike}%
\def\plttitle{Tabele}%
%%
\def\mtctitle{Kazalo}%
\def\mlftitle{Slike}%
\def\mlttitle{Tabele}%
%%
\def\stctitle{Kazalo}%
\def\slftitle{Slike}%
\def\slttitle{Tabele}%
⟨/slovene⟩
```

8.105 Langue « spanish » : spanish.mld

Les titres pour la langue « spanish » (espagnol) sont tirés du paquetage babel [10]. Notez que le « spanish » (espagnol) est en fait le « castillan » (voir section 8.23 page 238). D'autres langues sont parlées en Espagne : « basque » (section 8.12 page 233), « catalan » (section 8.25 page 239), et « galician » (galicien) (section 8.47 page 248). La langue « spanish2 » est une variante de « spanish » avec des titres plus courts (voir la section 8.106). La langue « spanish3 » (voir la section 8.107 page suivante) est une variante pour le projet ANTOMEGA [24] ; certains titres sont différents.

```
⟨*spanish⟩
\ProvidesFile{spanish.mld}[1999/03/16]
%% Spanish titles for minitoc.sty
\def\ptctitle{\'Indice General}%
\def\plftitle{\'Indice de Figuras}%
\def\plttitle{\'Indice de Tablas}%
%%
\def\mtctitle{\'Indice General}%
\def\mlftitle{\'Indice de Figuras}%
\def\mlttitle{\'Indice de Tablas}%
%%
\def\stctitle{\'Indice General}%
\def\slftitle{\'Indice de Figuras}%
\def\slttitle{\'Indice de Tablas}%
⟨/spanish⟩
```

8.106 Langue « spanish2 » : spanish2.mld

Les titres pour la langue « spanish2 » (espagnol deuxième variante) sont tirés du paquetage babel [10], mais abrégés pour les niveaux chapitre et section. Voir la section 8.105.

```
⟨*spanish2⟩
\ProvidesFile{spanish2.mld}[2005/03/31]
%% Spanish titles for minitoc.sty
\def\ptctitle{\'Indice General}%
\def\plftitle{\'Indice de Figuras}%
\def\plttitle{\'Indice de Tablas}%
%%
\def\mtctitle{Contenido}%
\def\mlftitle{Figuras}%
\def\mlttitle{Tablas}%
%%
\def\stctitle{Contenido}%
\def\slftitle{Figuras}%
\def\slttitle{Tablas}%
⟨/spanish2⟩
```

8.107 Langue « spanish3 » : spanish3.mld

Les titres pour la langue « spanish3 » (espagnol troisième variante) sont tirés du fichier omega-spanish.ldf du projet ANTOMEGA [24]. Voir la section 8.105 page précédente.

```
⟨*spanish3⟩
\ProvidesFile{spanish3.mld}[2005/09/06]
%% Spanish titles for minitoc.sty
%% from omega-spanish.ldf of the ANTOMEGA project.
\def\ptctitle{\localspanish{^^^^00cdndice general}}%
\def\plftitle{\localspanish{^^^^00cdndice de figuras}}%
\def\plttitle{\localspanish{^^^^00cdndice de cuadros}}%
%%
\def\mtctitle{\localspanish{^^^^00cdndice general}}%
\def\mlftitle{\localspanish{^^^^00cdndice de figuras}}%
\def\mlttitle{\localspanish{^^^^00cdndice de cuadros}}%
%%
\def\stctitle{\localspanish{^^^^00cdndice general}}%
\def\slftitle{\localspanish{^^^^00cdndice de figuras}}%
\def\slttitle{\localspanish{^^^^00cdndice de cuadros}}%
⟨/spanish3⟩
```

8.108 Langue « swedish » : swedish.mld

Les titres pour la langue « swedish » (suédois) sont tirés du paquetage babel [10] :

```
⟨*swedish⟩
\ProvidesFile{swedish.mld}[1999/03/16]
%% Swedish titles for minitoc.sty
\def\ptctitle{Inneh\csname aa\endcsname ll}%
\def\plftitle{Figurer}%
\def\plttitle{Tabeller}%
%%
\def\mtctitle{Inneh\csname aa\endcsname ll}%
\def\mlftitle{Figurer}%
\def\mlttitle{Tabeller}%
%%
\def\stctitle{Inneh\csname aa\endcsname ll}%
\def\slftitle{Figurer}%
\def\slttitle{Tabeller}%
⟨/swedish⟩
```

8.109 Langue « thai » : thai.mld et thai.mlo

Les titres pour la langue « thai » (thaï) sont tirés du paquetage babel [10], en utilisant des fontes du système CJK [29].

Les titres pour la langue « thai » contiennent des caractères qui ne peuvent pas être engendrés facilement, donc nous chargeons thai.mlo.

```
⟨*thai⟩
\ProvidesFile{thai.mld}[2005/01/28]
%% from thaicjk.ldf CJK 4.5.2 Thai support for the babel system
%% by Werner Lemberg <wl@gnu.org>
%%
\mtcloadmlo{thai}
⟨/thai⟩
```

8.110 Langue « turkish » : turkish.mld

Les titres pour la langue « turkish » (turc) sont tirés du paquetage babel [10] :

```
⟨*turkish⟩
\ProvidesFile{turkish.mld}[1999/03/16]
%% Turkish titles for minitoc.sty
\def\ptctitle{\.I\c cindekiler}%
\def\plftitle{\c Sekiller Listesi}%
\def\plttitle{Tablolar\in Listesi}%
%%
\def\mtctitle{\.I\c cindekiler}%
\def\mlftitle{\c Sekiller Listesi}%
\def\mlttitle{Tablolar\in Listesi}%
%%
\def\stctitle{\.I\c cindekiler}%
\def\slftitle{\c Sekiller Listesi}%
\def\slttitle{Tablolar\in Listesi}%
⟨/turkish⟩
```

8.111 Langue « UKenglish » : UKenglish.mld

La langue « UKenglish » est comme « english », donc nous chargeons simplement english.mld (voir la section 8.32 page 242) :

```
⟨*UKenglish⟩
\ProvidesFile{UKenglish.mld}[2005/07/11]
\mtcselectlanguage{english}%
⟨/UKenglish⟩
```

8.112 Langue « ukraineb » : ukraineb.mld

Les titres pour la langue « ukraineb » (ukrainien) sont tirés du paquetage babel [10]. Des fontes cyrilliques sont requises.

```
⟨*ukraineb⟩
\ProvidesFile{ukraineb.mld}[1999/12/06]
%% Ukraine (B) titles for minitoc.sty
%% Needs cyrillic fonts
\def\mtctitle{{\cyr\CYRZ\cyrm\cyrii\cyrs\cyrt}}%
\def\mlftitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
       \ \cyrii\cyrl\cyryu\cyrs\cyrt\cyrr\cyra\cyrc\cyrii\cyrishrt}}%
\def\mlttitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
       \ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyrsftsn}}%
%%
\def\ptctitle{{\cyr\CYRZ\cyrm\cyrii\cyrs\cyrt}}%
\def\plftitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
       \ \cyrii\cyrl\cyryu\cyrs\cyrt\cyrr\cyra\cyrc\cyrii\cyrishrt}}%
\def\plttitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
       \ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyrsftsn}}%
%%
\def\stctitle{{\cyr\CYRZ\cyrm\cyrii\cyrs\cyrt}}%
\def\slftitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
       \ \cyrii\cyrl\cyryu\cyrs\cyrt\cyrr\cyra\cyrc\cyrii\cyrishrt}}%
\def\slttitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
       \ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyrsftsn}}%
⟨/ukraineb⟩
```

8.113 Langue « USenglish » : USenglish.mld

La langue « USenglish » (américain) est comme « english »[31], donc nous chargeons simplement english.mld (voir la section 8.32 page 242) :

```
⟨*USenglish⟩
\ProvidesFile{USenglish.mld}[2005/07/11]
\mtcselectlanguage{english}%
⟨/USenglish⟩
```

[31] Ceci devrait être vrai pour les titres des mini-tables ; les langues elles-mêmes ont quelques différences.

8.114 Langue « usorbian » : `usorbian.mld`

Les titres pour la langue « usorbian » (haut sorabe) [32] sont tirés du paquetage `babel` [10]. Voir aussi la section 8.77 page 265.

```
⟨*usorbian⟩
\ProvidesFile{usorbian.mld}[1999/03/16]
%% Upper sorbian titles for minitoc.sty
%% Needs cyrillic fonts
\def\ptctitle{Wobsah}
\def\plftitle{Zapiswobrazow}
\def\plttitle{Zapistabulkow}
%%
\def\mtctitle{Wobsah}
\def\mlftitle{Zapiswobrazow}
\def\mlttitle{Zapistabulkow}
%%
\def\stctitle{Wobsah}
\def\slftitle{Zapiswobrazow}
\def\slttitle{Zapistabulkow}
⟨/usorbian⟩
```

8.115 Langue « vietnam » : `vietnam.mld`

Les titres pout la langue « vietnam » (vietnamien) sont tirés du paquetage `vietnam.sty`. Des fontes vietnamiennes sont requises. Voir aussi la section 8.116 page suivante.

```
⟨*vietnam⟩
\ProvidesFile{vietnam.mld}[1999/03/16]
%% vietnamese titles for minitoc.sty
%%
\def\ptctitle{M\d{u}c l\d{u}c}
\def\plftitle{Danh s\'ach h\`inh v\~e}
\def\plttitle{Danh s\'ach b\h{a}ng}
%%
\def\mtctitle{M\d{u}c l\d{u}c}
\def\mlftitle{Danh s\'ach h\`inh v\~e}
\def\mlttitle{Danh s\'ach b\h{a}ng}
%%
\def\stctitle{M\d{u}c l\d{u}c}
\def\slftitle{Danh s\'ach h\`inh v\~e}
\def\slttitle{Danh s\'ach b\h{a}ng}
⟨/vietnam⟩
```

[32] Le haut sorabe, le sorabe ou wende, est un membre du sous-groupe slave occidental des langues indo-européennes parlé en Haute-Lusace dans les *länder* allemands de Saxe et de Brandebourg. Les Sorabes sont des descendants des Wendes, nom germain des tribus slaves qui occupaient la zone entre l'Elbe et la Saale à l'ouest et l'Oder à l'est pendant la période médiévale (VIème siècle).

8.116 Langue « vietnamese » : vietnamese.mld

La langue « vietnamese » est simplement un synonyme pour la langue « vietnam » (vietnamien). Dons nous chargeons simplement `vietnam.mld`. Des fontes vietnamiennes sont requises, Voir aussi la section 8.115 page précédente.

```
⟨*vietnamese⟩
\ProvidesFile{vietnamese.mld}[2004/12/14]
\mtcselectlanguage{vietnam}%
⟨/vietnamese⟩
```

8.117 Langue « welsh » : welsh.mld

Les titres pour la langue « welsh » (gallois) sont tirés du paquetage `babel` [10] :

```
⟨*welsh⟩
\ProvidesFile{welsh.mld}[1999/12/06]
%% Welsh titles for minitoc.sty
\def\ptctitle{Cynnwys}%
\def\plftitle{Rhestr Ddarluniau}%
\def\plttitle{Rhestr Dablau}%
%%
\def\mtctitle{Cynnwys}%
\def\mlftitle{Rhestr Ddarluniau}%
\def\mlttitle{Rhestr Dablau}%
%%
\def\stctitle{Cynnwys}%
\def\slftitle{Rhestr Ddarluniau}%
\def\slttitle{Rhestr Dablau}%
⟨/welsh⟩
```

Compléments

Bibliographie

[1] Alex AJ. « Typesetting Malayalam Using LaTeX 2ε ». http://www.tug.org/tex-archive/language/malayalam/doc/mm-usr.pdf, avril 2003. alexaj@myrealbox.com.

[2] Donald ARSENEAU. « The notoccite package ». http://www.tug.org/tex-archive/macros/latex/contrib/misc/notoccite.sty, juillet 2000. asnd@triumf.ca.

[3] Donald ARSENEAU. « The placeins package ». http://www.tug.org/tex-archive/macros/latex/contrib/misc/placeins.sty et http://www.tug.org/tex-archive/macros/latex/contrib/misc/placeins.sty, juin 2002. asnd@triumf.ca.

[4] Benjamin BAYART. *Joli manuel pour LaTeX 2ε*. Guide local pour l'ESIEE, décembre 1995.

[5] Emmanuel BEFFARA. « Rubber Manual 1.0 ». http://www.pps.jussieu.fr/~beffara/soft/rubber/doc.html, juillet 2005.

[6] Berhanu BEYENE, Manfred KUDLEK, Olaf KUMMER et Jochen METZINGER. « Ethiopian Language Support for the Babel Package ». http://www.tug.org/tex-archive/language/ethiopia/ethiop/doc/ethiodoc.ps, décembre 1998. Universität Hamburg.

[7] Javier BEZOS. « Typesetting Guaraní with TeX ». http://www.tug.org/tex-archive/language/guarani/guarani.pdf, juillet 2004.

[8] Javier BEZOS. « The titlesec and titletoc packages ». http://www.tug.org/tex-archive/macros/latex/contrib/titlesec/titlesec.pdf, janvier 2005. http://www.texytipografia.com/contact.php.

[9] Johannes BRAAMS. « Babel, a multilingual style-option system for use with LaTeX's standard document styles ». *TUGboat*, 12(2):291–301, juin 1991.

[10] Johannes BRAAMS. « Babel, a multilingual package for use with LaTeX's standard document classes ». http://www.tug.org/tex-archive/macros/latex/required/babel/babel.pdf et http://www.tug.org/tex-archive/macros/latex/required/babel/user.pdf, juin 2004. babel@bramms.cistron.nl.

[11] Johannes BRAAMS, Victor EIJKHOUT et Nico POPPELIER. « The development of national LaTeX styles ». *TUGboat*, 10(3):401–406, novembre 1989.

[12] David Carlisle. « The xr package ». http:/www.tug.org/tex-archive/macros/required/tools/xr.dtx, mai 1994. carlisle@cs.man.ac.uk.

[13] Steven Douglas Cochran. « The subfig Package ». http://www.tug.org/tex-archive/macros/latex/contrib/subfig/subfig.pdf, janvier 2004. sds@cs.cmu.edu, cochran@ieee.org.

[14] Oliver Corff. « MonTEX – A Quick Guide *(draft)* ». http://www.tug.org/tex-archive/language/mongolian/montex/doc/mlsquick.pdf, novembre 2001. corff@zedat.fu-berlin.de.

[15] Oliver Corff. « MonTEX – Mongolian for LaTeX 2_ε, Implementation Level System Documentation ». http://www.tug.org/tex-archive/language/mongolian/montex/doc/montex.pdf, juillet 2002. corff@zedat.fu-berlin.de.

[16] Serguei Dachian, Arnak Dalalyan et Vardan Vakopian. « ArmTEX: a System for Writing in Armenian with TEX and LaTEX ». http://www.tug.org/tex-archive/language/armtex/v2.0/manual.ps, écrit en arménien, juin 1999. Serguei.Dachian@univ-lemans.fr.

[17] Bernard Desgraupes. *LaTEX, Apprentissage, guide et référence*. Vuibert, Paris, seconde édition, 2003.

[18] Jean-Pierre F. Drucbert. « The shorttoc package ». http://www.tug.org/tex-archive/macros/latex/contrib/shorttoc/shorttoc.dtx, août 2002. jean-pierre.drucbert@onera.fr.

[19] Jean-Pierre F. Drucbert. « Le paquetage minitoc ». http://www.tug.org/tex-archive/macros/latex/contrib/minitoc/fminitoc.pdf, juillet 2005. jean-pierre.drucbert@onera.fr.

[20] Jean-Pierre F. Drucbert. « The minitoc package ». http://www.tug.org/tex-archive/macros/latex/contrib/minitoc/minitoc.pdf, juillet 2005. jean-pierre.drucbert@onera.fr.

[21] Victor Eijkhout. « The comment package ». http://www.tug.org/tex-archive/macros/latex/contrib/comment/, octobre 1999. victor@eijkhout.net.

[22] Donald E. Knuth. *The TEXbook*, volume A de *Computers and Typesetting*. Addison-Wesley, Reading, Massachusetts, seizième édition, 1989. révisé pour couvrir TEX3, 1991.

[23] Donald E. Knuth. *Le TEXbook, Composition informatique*. Vuibert, Paris, 2003. Traduction en français de [22], par Jean-Côme Charpentier.

[24] Alexej M. Kryukov. « Typesetting multilingual documents with ANTOMEGA ». http://www.tug.org/tex-archive/systems/omega/contrib/antomega/doc/omega/antomega/antomega.pdf, septembre 2003.

[25] Klaus Lagally. « ArabTEX—Typetting Arabic with vowels and ligatures ». Dans *Proceedings of the 7th European TEX Conference*, pages 153–172, Prague, 1992. CsTUG. ISBN 80-210-0480-0, lagallyk@acm.org.

[26] Klaus Lagally. « ArabTeX Version 3 ». http://www.tug.org/tex-archive/language/arabtex/doc/html/arabtex.htm, octobre 2003. lagallyk@acm.org.

[27] Leslie Michel LAMPORT. *LaTeX: A Document Preparation System: User's Guide and Reference Manual*. Addison-Wesley, Reading, Massachusetts, seconde édition, 1994.

[28] LaTeX3 PROJECT TEAM. « LaTeX 2ε font selection ». `http://www.tug.org/tex-archive/macros/latex/doc/fontguide.pdf`, février 2004.

[29] Werner LEMBERG. « The CJK package for LaTeX 2ε — Multilingual support beyond babel ». *TUGboat*, 18(3):214–224, 1997.

[30] Knut LICHERT. « LaTeX 2ε for people in associations: `minutes.sty` ». `http://www.tug.org/tex-archive/macros/latex/contrib/minutes/`, janvier 2001.

[31] Nicolas MARKEY. Tame the BeaST (Tthe B to X of BibTeX). `http://www.tug.org/tex-archive/info/bibtex/tamethebeast/ttb_en.pdf`, `markey@lsv.ens-cachan.fr`, juin 2005.

[32] Rowland McDONNELL. « The `sectsty` package v2.0.2 ». `http://www.tug.org/tex-archive/macros/latex/contrib/sectsty/`, avril 2002. `rowland.mcdonnell@physics.org`.

[33] Frank MITTELBACH, Michel GOOSSENS, Johannes BRAAMS, David CARLISLE et Chris ROWLEY. *The LaTeX Companion 2*. Tools and Techniques for Computer Typesetting. Addison-Wesley Professional, Reading, Massachusetts, 2004.

[34] Tobias OETIKER, Hubert PARTL, Helene HYNA et Elisabeth SCHLEGL. « Une courte (?) introduction à LaTeX 2ε ». *Cahiers GUTenberg*, Hors-série numéro 2, février 2003. Traduction de [35] en français par Matthieu HERRB, adaptation et compléments par Daniel FLIPO, voir aussi : `http://www.tug.org/tex-archive/info/lshort/french/flshort-3.20.pdf`.

[35] Tobias OETIKER, Hubert PARTL, Helene HYNA et Elisabeth SCHLEGL. « The Not So Short Introduction to LaTeX 2ε ». `http://www.tug.org/tex-archive/info/lshort/english/lshort.pdf`, avril 2004.

[36] Palash Baran PAL. « Bangtex: a package for typesetting documents in Bangla using the TeX/LaTeX systems ». `http://tnp.saha.ernet.in/~pbpal/bangtex/bangtex.html`, janvier 2001.

[37] Oren PATASHNIK. « BibTeXing ». Documentation pour les utilisateurs normaux de BibTeX, `http://www.tug.org/tex-archive/biblio/bibtex/contrib/doc/btxdoc.pdf`, février 1988.

[38] Oren PATASHNIK. « Designing BibTeX Styles ». La partie de la documentation de BibTeX qui n'intéresse pas tous les utilisateurs, `http://www.tug.org/tex-archive/biblio/bibtex/contrib/doc/btxhak.pdf`, février 1988.

[39] Sebastian RAHTZ et Heiko OBERDIEK. « Hypertext marks in LaTeX: a manual for `hyperref` ». `http://www.tug.org/tex-archive/macros/latex/contrib/hyperref/doc/manual.pdf`, février 2004.

[40] Young RYU. « The TX Fonts ». `http://www.tug.org/tex-archive/fonts/txfonts/doc/txfontsdocA4.pdf`, décembre 2000.

[41] Stephan P. von Bechtolsheim. « TEX in practice: Comments on a 4-volume, 1400-page series on TEX ». *TUGboat*, 11(3):409–412, septembre 1990.

[42] Stephan P. von Bechtolsheim. *TEX in Practice*. Springer Verlag, Berlin, Germany / Heidelberg, Germany / London, UK / etc., 1992.

1. Basics.
2. Paragraphs, Maths and Fonts.
3. Tokens, Macros.
4. Output Routines.

(Une approche progressive des secrets de TEX).

[43] Peter R. Wilson. « The `needspace` package ». `http://www.tug.org/tex-archive/macros/latex/contrib/misc/needspace.sty`, janvier 2003. `pandgwilson@earthlink.net`.

[44] Peter R. Wilson. « The `tocloft` package ». `http://www.tug.org/tex-archive/macros/latex/contrib/tocloft/tocloft.pdf`, septembre 2003. `pandgwilson@earthlink.net`.

[45] Peter R. Wilson. « The `abstract` package ». `http://www.tug.org/tex-archive/macros/latex/contrib/abstract/abstract.pdf`, avril 2004. `pandgwilson@earthlink.net`.

[46] Peter R. Wilson. « The `appendix` package ». `http://www.tug.org/tex-archive/macros/latex/contrib/appendix/appendix.pdf`, avril 2004. `pandgwilson@earthlink.net`.

[47] Peter R. Wilson. « The `tocbibend` package ». `http://www.tug.org/tex-archive/macros/latex/contrib/tocbibind/tocbibind.pdf`, mai 2004. `pandgwilson@earthlink.net`.

[48] Peter R. Wilson. « The Memoir Class ». `http://www.tug.org/tex-archive/macros/latex/contrib/memoir/memman.pdf`, février 2005. `pandgwilson@earthlink.net`.

[49] Peter R. Wilson et Donald Arseneau. « The `iftmarg` package ». `http://www.tug.org/tex-archive/macros/latex/contrib/misc/ifmtarg.sty`, mars 2000. `pandgwilson@earthlink.net`, `asnd@triumf.ca`.

Historique des modifications

00 1990/10/01

Version originale, par Nigel WARD.

00 1991/11/01

Révision pour réutiliser de manière transparente les commandes `\chapter`, `\section`, `\subsection`, et engendrer automatiquement le nom du fichier minitoc, avec en plus d'autres nettoyages (Dan JURAFSKY).

01 1993/06/01

Ajout de `\chapterend` pour terminer la portée d'une minitoc. (*Si vous oubliez de mettre* `\chapterend` à la fin de *chaque* chapitre, une entrée pour le chapitre suivant apparaîtra dans chaque minitoc.) (Merci à Yufan HU).

Ajout de quelques modifications pour fonctionner avec `xr.sty` (références externes). `xr.sty` version 5 est bien plus tolérant.

Au moins trois passes (3) de LaTeX sont nécessaires pour obtenir des minitocs correctes (la première passe crée les fichiers `.mtc`⟨*X*⟩, la deuxième les utilise (mais ils peuvent contenir des numéros de pages faux) et les recrée, la troisième devrait être bonne).

Fonctionne avec `\chapter[xxx]{yyy}` et les éléments flottants.Fonctionne avec deux colonnes (mais la minitoc est composée en une colonne ; comment faire pout qu'elle s'étende sur les deux colonnes ?).

La profondeur de la minitoc est ajustable par l'utilisateur avec le compteur `minitocdepth` (similaire à `tocdepth` pour la table des matières).

Nouvelle conception, pour éviter d'allouer un newwrite, ou descripteur de fichier, pour chaque chapitre (une faute mortelle) (Jean-Pierre F. DRUCBERT).

Problèmes : vous *devez* avoir mis `\chapterend` pour terminer chaque chapitre ayant une minitoc. Comment éviter cette contrainte ?

Remplacement de l'environnement `minipage` par un environnement `verse`, pour qu'une minitoc puisse se répartir sur plusieurs pages.

Toute la mise en page de la minitoc est dans la commande, donc si quelqu'un souhaite redéfinir cette mise en page, il a juste à la réécrire (et seulement elle).

Vous pouvez inhiber la minitoc pour le chapitre suivant en le faisant précéder par `\minitocno`. (`\minitocyes` est sans objet pour l'utilisateur : elle est implicite *après* les pseudo-chapitres `\chapter*`).

02 1993/07/05

Ajout de la compatibilité avec `hangcaption.sty` (le paquetage `hangcaption` (s'il est utilisé) doit être chargé *avant* le paquetage `minitoc`). *Attention* aux options qui modifient `\@caption`.

03 1993/07/09

Version 3 non distribuée (erronée).

04 1993/07/09

Ajout de `\if@realch` pour éviter que des lignes de contenu provenant de pseudo-chapitres aillent dans la table des matières. Le fichier paquetage `mtcoff.sty` vous permet d'utiliser un document LaTeX contenant des commandes minitoc et de les rendre transparentes : il suffit de remplacer l'option minitoc par mtcoff.

05 1993/07/13

Ajout d'un mécanisme de sélection pour ne pas écrire des choses superflues dans les minitocs.

06 1993/07/15

Correction de problèmes sur les chapitres dans la table des matières, retrait de code obsolète pour `\caption` (les filtres sont meilleurs), ajout de la compatibilité avec `toch.sty` (`toch.sty` crée une table des chapitres ; si utilisé, doit être chargé *avant* `minitoc.sty`).

07 1993/07/22

(*différences majeures*) Réécriture complète, en utilisant des astuces de `xr.sty` (la version 5, de David Carlisle). Les informations pour les minitocs sont directement prélevées dans le fichier `.toc`. Suppression de `\chapterend` et `\minitocno`, ajout de `\minitoc`, `\dominitoc` et `\faketableofcontents`.

08 1993/07/29

Ajustements d'espacement.

09 1993/08/04

Ajout de modifications pour MS-DOS (rechercher MS-DOS, et décommenter ; rechercher UNIX, mettre en commentaire). MS-DOS permet seulement 3 caractères pour les suffixes dans les noms de fichiers (pitoyable. . .).

10 1993/08/05

Fonctionne avec les appendices. Détecte maintenant les versions obsolètes de `latex.tex` : (`\@inputcheck` ou `\reset@font` non définies).

11 1993/08/18

Ajout de `\mtcSfont`, fonte pour les entrées de section, `\mtcSSfont` pour les entrées de sous-section, `\mtcSSSfont` pour les entrées de sous-sous-section, `\mtcPfont` pour les entrées de paragraphe, `\mtcSPfont` pour les entrées de sous-paragraphe.

12 1993/12/16

Utilisation de `\kern` au lieu de `\vspace*` et ajout de pénalités (`\nopagebreak`) pour éviter une coupure de page just avant la dernière `\mtc@rule`. Ajout aussi d'un environnement `samepage`. Retrait de vieilles lignes mises en commentaire dans des versions précédentes.

13 1993/12/17

Ajout de code pour `minilof` et `minilot`. Pour MS-DOS, décommenter la définition de `\SHORTEXT`.

14 1994/01/03

Correction de l'espacement en dessous d'une `minitoc/lof/lot` et ajout de `\raggedright` pour éviter des avertissements « underfull ».

Correction de quelques problèmes d'espacement (en évitant des ~'s). `\mtifont` est changée de `\normalsize\bf` en `\large\bf`.

Quelques modifications suggérées par Donald Arseneau (merci) : `\@newread` devient `\newread`, une version *not outer* de `\newread` ; `\empty` remplacée par `\relax` dans la définition par défaut de `\reset@font`.

Retrait du forçage de `\clubpenalty` et `\widowpenalty` à `10000` (fait par `\samepage`) et `\noindent`.

Traitement simplifié de l'argument optionnel dans les commandes `\minitoc`, `\minilof` et `\minilot`.

15 1994/01/27

Ajout de `\parttoc`, `\partlof` et `\partlot` pour la classe `book`, de `\secttoc`, `\sectlof` et `\sectlot` pour les articles, avec des commandes et paramètres en parallèle avec ceux pour les mini-tables.

16 1994/02/02

Corrections d'erreurs (typos).

17 1994/06/23

« `n` » (nul) synonyme de « `e` » (*empty*, vide) dans l'argument optionnel de argument `\minitoc`, `\dominitoc`, et analogues.

Compatibilité avec « LaTeX 2_ε ». Merci à Denis Roegel (qui a trouvé le problème) et Frank Mittelbach (qui a donné les indications pour le résoudre).

18 1994/06/26

Introduction des fichiers de langage comme options. Grand merci à Michel Goossens (via Frank Mittelbach) qui a été inspiré par le code du paquetage `babel` (de Johannes Braams).

Rendu `minitoc` vraiment compatible avec LaTeX 2_ε.

19 1994/08/16

Ajout de code pour numéroter les chapitres (parties, sections) sans commencer à 1. Ajout des commandes `\firstchapteris`, etc.

Ajout des commandes `\mtcrule`, `\nomtcrule`, etc.

Correction d'une erreur dans `\c@mti`.

Correction dans `mtcswedish.sty` (Jan Michel RYNNING.)

Correction de l'appendice dans les articles.

20 1994/08/25

Ajout des commandes `\mtcpagenumbers` et `\nomtcpagenumbers` (et analogues) pour faire des minitocs avec ou sans numéros de pages. Défaut : avec numéros de pages.

Correction d'un problème avec des chapitres numérotés avec des chiffres romains majuscules.

Correction de l'espacement avant et après les minitocs et analogues.

Correction de l'espacement vertical.

Correction du problème (difficile) de l'appendice dans les articles.

21 1994/09/07

Correction de typos dans `minitoc.sty` et `minitoc.tex`.

22 1994/10/10

Correction de typos dans `minitoc.sty`.

23 1994/11/08

Ajout d'une ligne manquante dans `\sectlof`.

Fonctionne avec les classes de document réinitialisant le numéro de chapitre (ou de section) à chaque partie (merci à Denis ROEGEL).

Retrait du code pour `\firstchapteris` et compagnie. Ces commandes sont obsolètes.

Retrait du code pour l'appendice.

24 1994/12/21

Les commandes `\protect` ont été retirées des fichiers `.toc`, `.lot` et `.lot`, donc certaines commandes internes ont été corrigées pour être compatibles avec la version de LaTeX 2_ε de décembre 1994. Merci à Denis ROEGEL qui a fait le travail.

25 1996/09/13

Mise à jour de `mtcnorsk.sty` et ajout de `mtcnynorsk.sty` sur une suggestion de Dag LANGMYHR.

26 1996/11/14

Ajout des langues breton, estonian, germanb, greek, irish, russianb, scottish, lower- et upper-sorbian ; renommé « esperanto » en « esperant » comme dans le paquetage `babel`.

Les fichiers spécifiques des langues sont maintenant nommés ⟨*langue*⟩`.mld` (remplace `mtc`⟨*langue*⟩`.sty`) parce que ne sont pas des paquetages et que cela fait des noms plus courts.

27 1996/12/20

`english.mld` chargé comme langue par défaut.

Ajout de `vietnam.mld` et `arab.mld`.

Corrections pour les commandes de sectionnement étoilées.

Renommé `minitocoff.sty` en `mtcoff.sty` pour que le nom reste court.

28 1997/10/29

Ajout de l'autoconfiguration des suffixes.

Ajout de l'option de paquetage `shortext`.

Ajout des langues afrikaan(s), brazil et ethiopia(n).

Ajout du code *coffee* (pauses café).

Ajout du code de `\addstarred` (pour ce qui concerne un chapitre étoilé).

Correction d'une erreur dans les parttocs.

28 1998/06/15

Ajout de la langue bahasa.

Un typo corrigé par Donald ARSENEAU :
`{\let@dottedtocline\@undottedtocline}{}`
devrait probablement être
`{\let\@dottedtocline\@undottedtocline}{}`
(une contre-oblique manquait après `\let`). Merci à lui.

28 1998/12/03
Ajout des options de paquetage `tight` et `loose`.

29 1999/03/16
Ajout des langues bicig, buryat, mongol et russianc.

29 1999/06/28
Ajout de la langue armenian (d'après ArmTEX).

29 1999/07/23
Ajout des options de paquetage `dotted/undotted` (défaut : `dotted`).

29 1999/07/29
Ajout de la langue lithuanian.

30 1999/12/06
Ajout des langues basque, ngermanb, serbian, ukraineb et welsh.
Correction d'une erreur dans la définition de `\sltname` (`mlt` devrait être `slt`).

31 2000/04/04
Ajout de la compatibilité avec le paquetage `hyperref`, merci à Heiko OBERDIEK, qui a aussi simplifié certains morceaux de code et corrigé l'infâme erreur sur `\chapter*`.

32 2000/08/08
Ajout de très (trop) nombreuses nouvelles commandes pour les dispositifs associés aux mini-tables : `\beforeparttoc`, `\beforepartlof`, `\beforepartlot`, `\afterparttoc`, `\afterpartlof`, `\afterpartlot`, `\thispageparttocstyle`, `\thispagepartlofstyle` et `\thispagepartlotstyle`.
Correction de `\nomtcrule`.
Documentation améliorée par Stefan ULRICH.

33 2000/12/07
Ajout de nouvelles commandes d'ajustement : `\mtcaddchapter`, `\mtcaddsection` et `\mtcaddpart`. Ces commandes ajoutent des choses dans les fichiers `.toc`, `.lof` et `.lot` pour le problème de `\chapter*` (`\section*` et `\part*`). D'après une suggestion de Karl F. EVERITT.

33 2000/12/08
Correction d'un problème dans `\mtcaddchapter` et co. avec un argument optionnel blanc.

34 2000/12/13
Ajout dans la documentation d'un section sur l'utilisation avec le paquetage `tocbibend`.
Ajout de fichiers `.mld` pour les noms alternatifs de langues : ainsi, `american.mld` ne fait que charger `english.mld`, qui contient les vraies définitions.

35 2001/01/09
Ajout de macros pour tester si un fichier est « vide » (c'est-à-dire vide, blanc ou inexistant) ou « non vide » (c'est-à-dire utile). J'ai utilisé du code de Stephan P. von BECHTOLSHEIM.

Ajout des options de paquetage `checkfiles/nocheckfiles`.
Remplacement de `\The@chapter` par `\The@mtc`.

35 2001/02/26
Ajout de `bulgarian.mld`, `hebrew.mld`, `icelandic.mld`, `latin.mld` et `samin.mld`.

35 2001/03/09
Ajout de `\mtcselectlanguage`.

35 2001/06/01
Correction de l'option de paquetage `estionian` (manquante).

35 2001/07/04
Ajout de la langue interlingua.

36 2002/02/11
Correction d'une interaction avec `\tableofcontents` qui crée un `\chapter*` ou une `\section*`, en perturbant les compteurs `mtc/stc` (problème signallé par Frank Mittelbach).

36 2002/02/18
Contournement du problème `\parttoc`-`\chapter*`.
Correction d'un problème d'espacement avec des titres vides (problème signallé par Frank Mittelbach).
36 2002/02/19
Ajout de `\mtcskip` et `\mtcskipamount`.
36 2002/02/27
Correction du test sur les fichiers vides.
36 2002/03/13
Ajout de la langue bangla.
36 2002/03/15
Réduction de la profondeur de `\mtc@strutbox`.
37 2003/01/24
Version #37 sautée.
38 2003/01/24
`\hrule` et `\vrule` remplacées par `\rule` (LaTeX).
`pt` devient `\@pt` et `0pt` devient `\z@`.
Ajout de `mtc@zrule` pour les filets de dimensions nulles.
Ajout de la langue frenchb (synonyme de french).
38 2003/01/30
Ajout des options de paquetage `flsection` et `flsectionb`.
Changement du test pour les titres vides.
38 2003/01/31
Les options de paquetage `tight` et `loose` sont appliquées à `\parttoc` (Thomas Leonhardt).

38 2003/02/07
Options de paquetage `flsection` et `flsectionb` retirées et remplacées par l'option de paquetage `insection` (comme `flsectionb`).
38 2003/02/11
Correction de la numérotation des SLF, SLT.
38 2003/02/20
Ajout des langues frenchle et frenchpro (synonymes de french).
Correction des secttocs, enfin.
38 2003/03/18
Correction de certains espacements verticaux et d'étais (j'ai ajouté quelques modifications de Frank Mittelbach, grand merci à lui.). Il reste pas mal de nettoyage à faire, mais il semble nécessaire de publier la distribution maintenant.
39 2003/04/09
`\nomtcpagenumbers` et la classe `memoir`.
Commandes de fontes modernes pour compatibilité avec la classe `memoir`.
39 2003/06/08
Ajout de `\@fileswfalse` et `\mtc@hook@beforeinputfile` pour le paquetage `notoccite` (demandé par Donald Arseneau) ; ajout de l'option de paquetage `notoccite` (charge le paquetage `notoccite`).
39 2004/09/08
Ajout de commentaires dans les fichiers `.mld` utilisant des fontes spéciales.
Ajout des options de langue et des fichiers `.mld` pour les dialectes : canadian (english), acadian, acadien, canadien (french), naustrian, ngerman (ngermanb).

Corrections dans la documentation : ajout d'un paragraphe sur comment faire une table des matières pour les appendices, éventuellement non listée dans la table des matières principale.

39 2004/09/17

Corrections dans la documentation ; corrections sur les filets.

40 2004/12/09

Ajout d'un paragraphe sur un problème avec le paquetage `appendix`.

Ajout d'une figure dans `minitoc.tex` sur la nécessité de trois compilations.

Ajout de quelques informations dans `minitoc.bug`.

Ajout des langues japanese et castillan.

Retrait du test sur la présence du paquetage `multicol` dans `minitoc.tex`, puisque `multicol` est un paquetage obligatoire (*required*).

40 2004/12/13

Mise à jour de `fminitoc.bib` et `minitoc.bib`.

40 2004/12/14

Ajout de l'option de paquetage `hints`. Cette option est encore expérimentale ; votre avis est le bienvenu.

40 2004/12/20

Ajout de `fminitoc.pdf` (documentation en français en format PDF).

41 2005/01/05

Corrections dans la documentation.

Message ajouté si certaines commandes de sectionnement ne sont pas disponibles.

Remplacement des commandes `\typeout` dans `minitoc.sty` par les commandes `\PackageInfo` ou `\PackageWarning` ; avec le numéro de ligne lorsqu'il est utile (`\@gobble` si pas de numéro de ligne). En conséquence, le paquetage est moins bavard (`\PackageInfo` écrit seulement dans le fichier `.log`, pas sur le terminal).

41 2005/01/06

Ajout des commandes `\mtcsetfont` (Benjamin Bayart) et `\mtcsettitlefont`, avec une syntaxe beaucoup plus simple.

41 2005/01/10

Ajout de la bibliographie.

41 2005/01/11

Classes $\mathcal{AMS}$: `amsart` et `amsproc` sont incompatibles avec `minitoc`, `amsbook` demande des précautions.

41 2005/01/12

Ajout de `\mtcsetformat`.

41 2005/01/18

Ajout d'un *hint* pour recommander l'option de paquetage `insection`.

Ajout de `\mtcsettitle`.

41 2005/01/19

Ajout d'un *hint* sur la cohérence `\dominitoc`/`\minitoc` et co.

Ajout d'un *hint* sur la présence de `\dominitoc` et co.

Amélioration de la documentation sur les *hints*.

41 2005/01/20

Ajout d'un *hint* sur l'utilisation des suffixes courts avec plus de 99 parties ou 99 chapitres ou 99 sections.

41 2005/01/25

`\ptifont` : `\Huge\bfseries` devient `\LARGE\bfseries`.

41 2005/01/26

Ajout de `\mtcsetpagenumbers`.

41 2005/01/28

Ajout de nombreux fichiers de langues : `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`.

41 2005/02/02
Ajout de `\mtcsetrules`.
41 2005/02/03
Ajout de `\plfrule`, `\noplfrule`, `\mlfrule`, `\nomlfrule`, `\slfrule`, `\noslfrule`, `\pltrule`, `\nopltrule`, `\mltrule`, `\nomltrule`, `\sltrule`, `\nosltrule`.
41 2005/02/04
Ajout de l'environnement `mtchideinmaintoc`.
41 2005/02/08
Ajout de nouvelles langues : `latvian.mld`, `letton.mld`, `greek-mono.mld`, `greek-polydemo.mld`, `greek-polykatha.mld`, `polish2.mld`, `russian2m.mld`, `russian2o.mld`.
41 2005/02/09
Ajout des environnements `mtchideinmainlof` et `mtchideinmainlot`.
41 2005/02/10
Ajout de tests sur les environnements `mtchideinmain*`.
41 2005/02/14
Ajout de `\mtcfixindex`.
42 2005/02/14
Remplacement de « language » par « langue » dans la documentation française.
42 2005/02/15
Correction d'un typo mineur.
42 2005/02/16
Amélioration de `\mtcfixindex`.
42 2005/02/21
Ajout de `\mtcsettitle`, oubliée dans v41.
43 2005/02/21
Version 43 : consolidation de v40, v41 et v42.
43 2005/02/24
Correction d'une erreur dans `mtcoff.sty` à propos de `\mtcfixindex`.
Correction d'une grosse erreur dans `\mtcsetformat`.
43 2005/03/02
Ajout du fichier `INSTALL` et d'un chapitre sur l'installation.
Correction des macros `\mtcset...`
Déplacement de l'historique vers la fin du code du paquetage.
43 2005/03/07
Compléter le *hint* sur la cohérence `\dominitoc`/`\minitoc` et co.
Correction d'un typo (Benjamin Bayart).
43 2005/03/08
Ajout d'un *hint* sur la cohérence entre `\minitoc` et `\tableofcontents`.
43 2005/03/09
Ajout de commentaires sur les fontes.
43 2005/03/10
Corrections dans la documentation.
43 2005/03/11
Ajout de `\mtcsetfeature`.
43 2005/03/14
Ajout de `bulgarianb.mld` (haut bulgare).
43 2005/03/15
Ajout de `*[-\baselineskip]` après le `\\` après le filet supérieur de chaque mini-table au niveau partie.
43 2005/03/16
Corrections dans les arguments de `\mtcsetfeature`.
43 2005/03/18
Retrait de `\markboth` pour les minitocs (...) et secttocs (...).

43 2005/03/21
Ajout de spanish2.mld.
43 2005/03/22
Ajout d'un *hint* pour le paquetage abstract.
43 2005/04/06
Ajout de finnish2.mld, latin2.mld et magyar2.mld.
Correction de l'environnement stc@verse.
43 2005/04/08
Renommage de portuges.mld en portugues.mld.
43 2005/04/12
Correction dans \mtcskip.
Première version en format .dtx.
43 2005/04/14
Retrait de \ypart, \ychapter, \ysection, etc., inutilisées.
43 2005/05/11
Ajout de \mtcfixglossary.
Ajout de minitoc.ist pour mettre en page l'index correctement.
Correction d'un typo dans \@dosectlot.
Impression de la documentation avec « oneside » pour avoir toutes les notes marginales sur la gauche. Ajout du code (étendu) de morefloats.sty (Don Hosek) pour permettre davantage de notes marginales et d'éléments flottants.
43 2005/05/26
Correction des filets dans les parttocs, partlofs et partlots.
43 2005/05/30
Ajout d'un *hint* sur le paquetage sectsty (il doit être chargé *avant* minitoc).
Correction des entrées de niveau chapitre dans les parttocs, lorsque les numéros de pages doivent être omis.
43 2005/06/01
Ajout d'un *hint* sur l'utilisation de commandes obsolètes.
Ajout d'un *hint* sur les tentatives d'insertion de mini-tables vides.
Les mini-listes de figures ou de tableaux ne doivent pas être imprimées vides même si tocdepth < 1.
43 2005/06/02
Ajout de \mtcsetdepth.
Ajout de la notion de profondeur pour les mini-listes de figures/tableaux.
L'option hints est le choix par défaut et n'est plus considérée comme expérimentale.
43 2005/06/03
Ajout d'un message d'erreur dans \mtcsetdepth si le compteur n'est pas disponible.
43 2005/06/06
Ajout de portuges.mld, qui charge portugues.mld.
43 2005/06/07
Ajout de trois variantes pour la langue malayalam : malayalam-keli.mld, malayalam-rachana.mld et malayalam-rachana2.mld.
43 2005/06/14
Ajout d'une méthode pour documentation bilingue.
43 2005/06/15
Ajout de fminitoc.ist pour mettre en forme correctement l'index en français.
43 2005/06/16
Changement de « Liste des Tables » en « Liste des Tableaux » dans french.mld et dans la documentation française, pour suivre les choix du paquetage babel.
43 2005/06/17
Le fichier fminitoc.dtx est maintenant engendré par minitoc.ins.
43 2005/06/21
Ajout des mots-clés OUI, NON, oui, non, O et o pour vrai/faux.

Condensation du code de détection des suffixes courts ou longs.

43 2005/06/22

Ajout des mots-clés `VRAI`, `FAUX`, `vrai`, `faux`, `V` et `v` pour vrai/faux.

43 2005/06/23

Établir correctement l'indicateur `\ifFTR` pour avoir les mois dans la langue correcte dans la bibliographie.

43 2005/06/29

Établir l'indicateur `\mtcoffwarn@true` par `mtcoff.sty` si une commande `\mtcadd...` est trouvée.

43 2005/07/01

Ajout de `castillian.mld`.

Renommage de `portugues.mld` en `portuguese.mld`.

43 2005/07/11

Ajout de `brazilian.mld`, `british.mld`, `UKenglish.mld` et `USenglish.mld`.

43 2005/07/12

Suppression de « Général : » dans l'historique des modifications.

43 2005/07/13

Remplacement de quelques commandes `\PackageWarning` par `\PackageInfo`.

43 2005/07/18

Restauration de la correspondance de chaque option de langue avec un fichier `.mld`.

43 2005/07/20

Amélioration des environnements `mtchideinmainlof` et `mtchideinmainlot`.

43 2005/07/21

Ajout des commandes `\decrementptc`, `\decrementmtc` et `\decrementstc`.

Retrait de deux indicateurs inutiles : `\if@mtc@setpagenumbers@act@` et `\if@mtc@setrules@act@`.

43 2005/07/22

Ajout d'un test sur la version du paquetage `placeins`.

Amélioration de quelques messages dans `mtcoff.sty`.

Correction d'une bourde dans `mtcoff.sty`.

43 2005/08/23

Ajout d'une note sur `\FloatBarrier`.

43 2005/08/24

Ajout d'une note sur un problème d'alignement dans les minitocs. Mise à jour de `minitoc.bug`.

Fait deux versions des environnements `mtchideinmainlof` et `mtchideinmainlot`, selon la présence du compteur de profondeur correspondant.

La classe `memoir` est incompatible si elle est trop récente.

43 2005/08/25

Ajout d'un commentaire sur la position des commandes `\do...`de préparation.

Corrections dans les environnements `mtchideinmainlof` et `mtchideinmainlot`.

43 2005/08/26

Ajout de `guarani.mld`.

43 2005/08/29

Ajout de `\incrementptc`, `\incrementmtc` et `\incrementstc`. Ajout d'un argument optionnel à `\adjustptc`, `\adjustmtc` et `\adjuststc`.

Ajout des options `k-tight` et `k-loose`.

43 2005/09/02

Ajout d'un patch pour la version récente de la classe `memoir`.

43 2005/09/06

Ajout de `spanish3.mld`.

43 2005/09/08

Utiliser `\mtcselectlanguage` dans les options de langue et dans les fichiers `.mld` « secondaires ».

43 2005/09/09
Ajout de `\mtcloadmlo` à utiliser dans certains fichiers `.mld` pour charger un fichier `.mlo`.
43 2005/09/12
Ajout d'un test pour interdire les appels directs de `\mtcloadmlo` par l'utilisateur.
43 2005/09/13
Ajout d'une note sur le script `rubber`.
Ajout de `farsi1.mld`, `farsi1.mlo`, `farsi2.mld` et `farsi2.ml0`.
43 2005/09/15
Ajout de `mtcglo.ist` pour formatter le glossaire.
43 2005/09/16
Retrait des numéros de pages dans le glossaire. Fait par les scripts *mk.

Index

A

N

O

P

R

S

T