

The **etoc** package

JEAN-FRANÇOIS BURNOL

jfbu (at) free (dot) fr

Package version: v1.05 (2012/12/01)

Documentation generated from the source file
with timestamp "01-12-2012 at 13:48:11 CET"

Abstract

The **etoc** package gives to the user complete control on how the entries of the table of contents should be constituted from the *name*, *number*, and *page number* of each sectioning unit. This goes via the definition of *line styles* for each sectioning level used in the document. The package provides its own custom line styles as an example. The simplest usage though will be to delegate the details of the layout to packages dealing with list environments.

The `\tableofcontents` command can be used arbitrarily many times and has a variant `\localtableofcontents` which prints (surprise!) ‘local’ tables of contents. The formatting inherited (possibly customized by other packages) from the document class can also be used. Regarding the *global toc display*, **etoc** provides styles based on a possibly multi-column format, with either a heading or a title in-between rules, optionally with a frame around the table of contents. Using labels, one can reproduce at another location (and with another layout) a (local) TOC defined somewhere in the document.

Foreword

Popular packages dealing with TOCs include `tocloft`, `titletoc` and `minitoc`. Why another one? well, I started **etoc** for my own use, and only later found out about the above mentioned packages . . .

As is well explained in the `tocloft` package documentation, the standard L^AT_EX layout for the Table of Contents is buried in the class definitions. In particular, most of the lengths therein are hardcoded, and the only way for the user to change them is to recopy the class definitions into the document and then change them to obtain what is desired (within suitable `\makeatletter` and `\makeatother`). The more reasonable alternative is to use a dedicated package such as `tocloft` or to use another flexible document class.

However, although now things are hopefully not hard-coded, one still has to go through the package or class interface. This means one has to memorize a (possibly large) number of macros which will serve only to this task, and one will always be constrained to customizing one initially given layout.

The spirit of **etoc** is something else. The user will deal with the *name*, the *number* and the *page number* corresponding to each document sectional division (and found in a line of the `.toc` file) in a completely arbitrary manner: they are made available via the `\etocname`, `\etocnumber`, and `\etocpage` commands.

It is hoped that **etoc** will turn out to be essentially compatible with the `article`, `book`, `report`, `scrartcl`, `scrbook`, `scrreprt` and `memoir` classes. Preliminary testing is positive, but the author not being familiar with the latter classes,¹ he does not have a stock of documents which could have been used to test all their bells and whistles in the presence of **etoc**.

¹latter=all but article...

CONTENTS

Part I. Overview

Initial motivation: nested lists	1, p. 3
Limitations intrinsic to the use of environments	1.1, p. 3
Line styles and global style	2, p. 4
<code>\etocsettocstyle</code> for the toc display	2.1, p. 4
Multiple columns, ruled title, framed contents. <code>\etocsetstyle</code> for the line styles	2.2, p. 4
Instructing <code>etoc</code> about the document sectional division names.	2.3, p. 5
Compatibility mode	2.3, p. 5
Arbitrarily many TOCs, and local ones too	3, p. 5
On manually adding layout commands to the <code>.toc</code> file	3.1, p. 5
Labeling and reusing elsewhere	3.2, p. 6
Local table of contents for this part	3.3, p. 7

Part II. Package commands for line styles

The <code>\etocsetstyle</code> command	4, p. 8
An example	4.1, p. 8
<code>\etocname</code> and <code>\etocpage</code> —The <code>\etocskipfirstprefix</code> command— <code>secnumdepth</code> and <code>tocdepth</code>	4.2, p. 10
The <code>\etocsetlevel</code> command	4.2, p. 10
Further informations and let's make this a very long title to see how it will look like	4.3, p. 10
The <code>\etocnumber</code> command—The <code>\etocifnumbered</code> switch.	4.4, p. 11
The <code>\etocfontxxx...</code> commands	4.4, p. 11
This is a (pale) red subsubsection for illustrative purposes.	

Am I also in red?	5, p. 12
-----------------------------	----------

An unnumbered section	p. 12
---------------------------------	-------

A comparison	6, p. 13
------------------------	----------

Part III. Package commands for global styles

Specifying the toc display style	7, p. 16
The command <code>\etocruledstyle</code>	7.1, p. 17
The command <code>\etocmulticolstyle</code>	7.2, p. 17
The command <code>\etocdisplaystyle</code>	
A point of some importance	7.3, p. 18
The command <code>\etocruled</code>	7.4, p. 18
The commands <code>\etocframedstyle</code> and <code>\etocframed</code>	7.5, p. 18
The command <code>\etocsettocstyle</code>	7.6, p. 18
The compatibility mode <code>\etocstandarddisplaystyle</code>	7.7, p. 19
The starred variants of the <code>\tableofcontents</code> etc... commands	8, p. 19

Table of contents for part III	9, p. 19
--	----------

Part IV. Using and customizing `etoc`

Summary of the main style commands	10, p. 22
Setting up local styles	10.1, p. 22
Setting up global styles	10.2, p. 22
Displaying Tables of Contents	10.3, p. 22
Labels and references	10.4, p. 22
An example design	10.5, p. 23

Customizing <code>etoc</code>	11, p. 25	All subsections of this document	12.3, p. 28
Customizing the <code>etoc</code> line styles	11.1, p. 25	Compatibility with other packages	12.4, p. 29
Customizing the global styles	11.2, p. 26	TEX nicl matters	12.5, p. 29
		Errors and catastrophes	12.6, p. 30
 Part V. Tips			
... and tricks	12, p. 27	Part VI. The code	
Hacking framed parboxes	12.1, p. 27		
Interverting the levels	12.2, p. 28	Implementation	13, p. 30

Part I.

Overview

1. Initial motivation: nested lists

The initial impetus was to feed nested list environments with the data consisting of the *name*, *number*, and *page number* as recorded in the `.toc` file.² For example, typesetting the line corresponding to the first sub-section in a given section would open a list environment which would be closed only when a section, chapter, or part line entry in the `.toc` file would be encountered. `etoc` allows to do this very easily and the opening and closing may be for example `\begin{enumerate}` and `\end{enumerate}` pairs, will all the customizing allowed by packages such as `enumitem`.

1.1. Limitations intrinsic to the use of environments

There is a first limitation to this method: the `.toc` file may contain other commands, such as language changing commands, which do not expect to see their scope limited in this way inside a group ($\text{L}\text{A}\text{T}\text{E}\text{X}$'s environments create groups). Therefore the package own line styles (illustrated by the main table of contents in this document) do not make use of environments to avoid that problem.

A second limitation is that one may nest at most 4 levels of `enumerate` environments, and 4 levels of `itemize` environment. I tried alternating them and did succeed to nest 6 levels (and not 8 alas . . . ³). With `\etocnumber` as the optional parameter to `\item`: `\item[\etocnumber]` one may transform the itemize into an enumerated list... anyway, 4 levels of sectional divisions in a TOC are generally sufficient, and again using `enumerate` environments is only a possibility provided by `etoc`, it is by no means mandatory to use

²I got started about this through reading this question on the TEX Stackexchange site: tex.stackexchange.com/questions/79682

³this is surely a well-known issue which I did not at all investigate any further.

2. Line styles and global style

them in the line styles specifications.

We will give in this manual a simple-minded example of nested use of `enumerate` environments. More sophisticated examples would use more sophisticated `enumitem` options. One may say then that again the user has to memorize some customizing! indeed, but the syntax and option names to memorize are in no way related only to matters of tables of contents, hence an economy of use of the poor brain.

Still a different sort of example, less elementary and not making use at all of list-making environments, is built-in in `etoc` own “line styles”. They are activated by the command `\etocsettoclines...` and do come with their own set of customizing macros!

2. Line styles and global style

I should have distinguished between the *line styles* (the way the name, number and page numbers are used at each level) which I was alluding to until now and the *toc display style* (I am using this for lack of a better name) which tells how the title should be set, whether an entry in the `.toc` file should be made, whether the contents should be typeset with multiple columns, etc...

2.1. `\etocsettocstyle` for the toc display

The low-level `\etocsettocstyle` command allows to decide what should be done before and after the line entries of the TOC are typeset, and in particular how the title should be printed.

2.1.1. Multiple columns, ruled title, framed contents

`etoc` provides four (customizable...) toc styles which are variants on the use of the `multicol` package: `\etocmulticolstyle`, `\etoctocstyle`, `\etocruledstyle`, and `\etocframedstyle`. All use the `multicol` package with a default of two columns (single-column mode is of course allowed).

These commands must be followed either by `\tableofcontents` or `\localtableofcontents`, and dropping the ‘style’ in the command name gives shortcuts (except for `\etoc` which does not exist) for doing the whole thing in one go (and inside a group).

2.2. `\etocsetstyle` for the line styles

The command to inform `etoc` of what to do with `\etocname`, `\etocnumber`, and `\etocpage` is called `\etocsetstyle`. It has five mandatory arguments. The first one is the name of the sectional unit: a priori known names are `book`, `part`, `chapter`, `section`, `subsection`, `subsubsection`, `paragraph`, and `subparagraph`. The four other arguments say: 1) what to do when this level is first encountered, down from a more general one, then 2) & 3) (two arguments, a ‘prefix’ and a ‘contents’) what to do when a new entry of that type is found, and 4) the last argument is the code to execute when a division unit of higher importance is again hit upon.

3. Arbitrarily many TOCs, and local ones too

2.2.1. Instructing **etoc** about the document sectional division names

Other names (*atom*, *particle* . . .) for sectional divisions may be made known to **etoc** with the command `\etocsetlevel`.⁴

2.3. Compatibility mode

Both for the “line styles” and the “global style”, it is possible to switch into a compatibility mode which uses the document class default style.⁵ This is activated by:

```
\etocstandardlines      % ‘line entries’ as without \usepackage{etoc}
\etocstandarddisplaystyle % ‘toc display’ as without \usepackage{etoc}
```

Actually, starting with version 1.05, this is the default at the start of the document, and just adding `\usepackage{etoc}` should hopefully not change anything to the look of a document, at least for the `article`, `book`, `report`, `scrartcl`, `scrbook`, `scrreprt` and `memoir` classes.

After declaring a document division name to **etoc** with `\etocsetlevel`, and if the document class does things in a manner analogous as the standard classes, then the `\etocstandardlines` compatibility switch should be functional. But at any rate the **etoc** default line styles and the ones defined via `\etocsetstyle` will work. Except that maybe **etoc** will fail to separate the number from the name.

3. Arbitrarily many TOCs, and local ones too

etoc allows arbitrarily many `\tableofcontents` commands in your document. The line styles and the global style may of course be changed in-between. Furthermore `\localtableofcontents` will print local tables of contents⁶: *i.e.* all sections and sub-units inside a given chapter, or all subsections and lower inside a given subsection, etc . . .

3.1. On manually adding layout commands to the `.toc` file

When using in that way many tables of contents in the same document one should of course beware of adding manually things to the `.toc` file. For example the technique of inserting

```
\addtocontents{toc}{\string\clearpage}
```

just before a `\part` to fix the problem when some part entry (in the table of contents) is isolated at the bottom of one page will cause problems with multiple TOCs: this `\clearpage` will be executed by **etoc** each time a `\tableofcontents` or `\localtableofcontents` command is encountered! The more prudent thing is to have issued rather:

```
\addtocontents{toc}{\string\myclearpage}
```

and to use where needed something like:

```
\let\myclearpage\clearpage
\tableofcontents
\let\myclearpage\relax
```

⁴however, no more than the eight distinct levels from -2 to +5 may be used.

⁵for the “global style” **etoc** checks if it knows the class, and if not defaults to the `article` class layout.

⁶Up to version 1.04 we called this `\tableofcontents*`, but for reasons of compatibility with the `memoir` class, we have decided to drop this usage.

3. Arbitrarily many TOCs, and local ones too

The memoir class has the command `\settocdepth` which writes a `\changetocdepth` command inside the `.toc` file. This will impact the typesetting by `etoc` of all tables of contents, with possibly unexpected results: imagine the document has `\settocdepth{chapter}` at some point to avoid having the sections from subsequent chapters be listed in the main table of contents. Then a local table of contents in one of these chapters will print a title but will be without any entry. A solution is to do `\begin{group} \renewcommand*{\changetocdepth}[1]{} \localtableofcontents \end{group}`, and to set the desired level for the local table of contents with the other memoir command `\maxtocdepth`.⁷

3.2. Labeling and reusing elsewhere

`etoc` allows the labeling of a TOC with `\label{toc:A}` and will redisplay it elsewhere when told `\tableofcontents\ref{toc:A}`. Again, the actual layout (title inclusive) is decided locally. The line styles and global style (including the title) will be the current ones and the current value of the `tocdepth` counter is obeyed. As an example here is the table of contents of Part IV:

I am from far away	
10. Summary of the main style commands	22
10.1. Setting up local styles	22
10.2. Setting up global styles	22
10.3. Displaying Tables of Contents	22
10.4. Labels and references	22
10.5. An example design	23
11. Customizing <code>etoc</code>	25
11.1. Customizing the <code>etoc</code> line styles	25
11.2. Customizing the global styles	26

We actually did something like:

```
\tableofcontents \label{toc:d}\ref{toc:c}
```

Hence the present location can itself now be referred to via `\ref{toc:d}`: it gives the id of this TOC⁸ in the sequence of document TOCs, and will be a link if package `hyperref` is used.

However one should not use elsewhere `\tableofcontents \ref{toc:d}`. The reason is that the doubly cloned TOC has no way of knowing that it is a clone of a clone, and what ‘local’ means will erroneously be construed as being decided by the location of the first clone, not the original. So to clone again, one should use the original: `\tableofcontents \ref{toc:c}`.

⁷The memoir class allows multiple calls to the `\tableofcontents` command, so these issues already arise there, independently of `etoc`, see page 170 of the memoir manual.

⁸i.e 3, there was an invisible TOC with id 2 at the beginning of this part.

3. Arbitrarily many TOCs, and local ones too

The `\label` command should follow `\tableofcontents` before the reference via `\ref` to the other TOC. Indeed a `\label` following it is detected by `\tableofcontents` (skipping spaces and an optional `*`). A `\label` after the `\ref` will not be treated by `etoc` itself and will refer to the latest reference point, possibly to some counter updated in the title of the TOC or, although this would require the user to have done a suitable set-up in the line styles, even some counter modified inside the TOC.

As another example let us display a table of contents for this first part of the document. Indeed, immediately after the line in the source file starting with the first `\part` we wrote:

```
\setcounter{tocdepth}{-2}
\localtableofcontents \label{toc:partone}
\setcounter{tocdepth}{2}
```

The `tocdepth` being set to `-2`, nothing at all was inserted in the page (for the same effect one must use `-3` in the `memoir` class; and `-3` works in all classes) but now that it has been reset to `2`, we can display the TOC with the simple command:

```
\tableofcontents \ref{toc:partone}
```

3.3. Local table of contents for this part

This sub-section heading was manually added to the document source, it is not part of the TOC style.

Contents of Part One		
1	Initial motivation: nested lists	3
1.1	Limitations intrinsic to the use of environments	3
2	Line styles and global style	4
2.1	<code>\etocsettocstyle</code> for the toc display	4
2.1.1	Multiple columns, ruled title, framed contents	4
2.2	<code>\etocsetstyle</code> for the line styles	4
2.2.1	Instructing <code>etoc</code> about the document sectional division names	5
2.3	Compatibility mode	5
3	Arbitrarily many TOCs, and local ones too	5
3.1	On manually adding layout commands to the <code>.toc</code> file	5
3.2	Labeling and reusing elsewhere	6
3.3	Local table of contents for this part	7

The actual design is not pre-built in `etoc`; it uses the `etoc` ‘framed’ style with a background color and a dummy empty title, the actual title having been put inside the frame

as part of the `<start>` code of the section ‘line style’. The frame borders have been set to have the same color as the one which serves as background for the entire thing. This design (with other colors) is in use also for [this other toc](#), and the coding used has been included in its subsection.

Part II.

Package commands for line styles

Contents of this second part

4 The `\etocsetstyle` command (page 8)

4.1 An example (p. 8)

`\etocname` and `\etocpage` – The `\etocskipfirstprefix` command – `secnumdepth` and `tocdepth`.

4.2 The `\etocsetlevel` command (p. 10)

4.3 Further informations and let’s make this a very long title to see how it will look like (p. 10)

The `\etocnumber` command – The `\etocifnumbered` switch.

4.4 The `\etocfontxxx...` commands (p. 11)

This is a (pale) red subsubsection for illustrative purposes.

5 Am I also in red? (page 12)

An unnumbered section (page 12)

6 A comparison (page 13)

4. The `\etocsetstyle` command

4.1. An example

Let us explain how `etoc` was used to produce the table of contents displayed at the beginning of this second part. This is a local table of contents, and we used the command `\localtableofcontents`.

We shall distinguish between the *line styles* and the *toc display style*. The line styles were (essentially) obtained in the following manner:⁹

⁹the present document has `\renewcommand{\familydefault}{\sfdefault}` in its preamble, hence `\normalfont` switches to the sans typeface; so in the section line-style, I wrote `\rmfamily` instead.

4. The `\etocsetstyle` command

```
\setcounter{tocdepth}{3}

\etocsetstyle{section}
{\begin{enumerate}}
{\normalsize\bfseries\rmfamily\item}
{\etocname (page \etocpage)}
{\end{enumerate}}

\etocsetstyle{subsection}
{\begin{enumerate}}
{\normalfont\item}
{\etocname (p.\~\etocpage)}
{\end{enumerate}}

\etocsetstyle{subsubsection}
{\par\nobreak\begingroup\normalfont
  \footnotesize\itshape\etocskipfirstprefix}
{\allowbreak\,--\,}
{\etocname\unskip}
{.\hfil\par\endgroup\pagebreak[3]}
```

Each `\etocsetstyle` command has five mandatory arguments:

`\etocsetstyle{<levelname>}{<start>}{<prefix>}{<contents>}{<finish>}`

The initially recognized `<levelname>`'s are the sectioning levels of the standard document classes: from *part* (or *book* which is used by the `memoir` class) down to *subparagraph*.

The `<start>` code is executed when a toc entry of that level is encountered and the previous one was at a higher level. The `<finish>` code is executed when one again encounters a higher level toc entry. In the mean-time all entries for that level are typeset by executing first the `<prefix>` code and then the `<contents>` code.

4.1.1. `\etocname` and `\etocpage`

The line styles specifications use the commands `\etocname` and `\etocpage` as synonyms for the name, respectively the page number of the current entry.¹⁰ Temporarily, we relied on the automatic numbering of the `enumerate` environments but there is also a command `\etocnumber` to get the real thing from the `.toc` file.

4.1.2. The `\etocskipfirstprefix` command

The chosen `subsubsection` style specifies to list the names of the subsubsections from a given subsection in an in-line manner, with – as a separator. The command `\etocskipfirstprefix`, which, if present, *must* be the very last one in the `start` code, instructs to not use that separator of the first item.

With this style, one would have to be imaginative to design something then for paragraph and subparagraph entries! perhaps as superscripts? Well, usually one does not need paragraphs and subparagraphs numbered and listed in the TOC, so our putative user here chose a design where no provision is made for them and added the definitive:

¹⁰ `etoc` uses the `xspace` package, and there is no need to write `\etocname{}` text rather than just `\etocname` text for example. However, this forced us to add a mysterious `\unskip` in the subsubsection style.

4. The `\etocsetstyle` command

```
\etocsetstyle{paragraph}{}{}{}{}{}
\etocsetstyle{subparagraph}{}{}{}{}{}

```

This is also the situation with the default package line styles!

4.1.3. `secnumdepth` and `tocdepth`

The \LaTeX counter `secnumdepth` decides which levels are numbered and automatically written to the `.toc` file. Its functioning is completely independent from the `etoc` package. The counter `tocdepth` decides the finest level displayed in the table of contents: its local value is obeyed by `etoc`, and as `\tableofcontents` can now be used arbitrarily times, this is an important customization lever.

4.2. The `\etocsetlevel` command

If your document uses other sectioning names, such as, for example *molecule*, or *atom*, it should inform `etoc` of this in the following way:

```
\etocsetlevel{cell}{0}
\etocsetlevel{molecule}{1}
\etocsetlevel{atom}{2}
\etocsetlevel{nucleus}{3}

```

The accepted numbers range from -2 to 5 inclusive. The package code does:

```
\etocsetlevel{book}{-2}
\etocsetlevel{part}{-1}
\etocsetlevel{chapter}{0}
\etocsetlevel{section}{1}
\etocsetlevel{subsection}{2}
\etocsetlevel{subsubsection}{3}
\etocsetlevel{paragraph}{4}
\etocsetlevel{subparagraph}{5}

```

4.3. Further informations and let's make this a very long title to see how it will look like

4.3.1. The `\etocnumber` command

So far, our specifications would use the numbering generated by the `enumerate` environments, but it would be nicer to use the actual numbers as found in the `.toc` file. This is available via the `\etocnumber` command. To get the labels in the `enumerate` list to use it we can proceed with the syntax `label=` from the package `enumitem`:

```
\etocsetstyle{section}
{\begin{enumerate}[label=\etocnumber]}
{\normalsize\bfseries\rmfamily\item}
{\etocname (page \etocpage)}
{\end{enumerate}}

```

Rather than just `\etocnumber` we then used something like `\fbox{\etocnumber}`.

4. The `\etocsetstyle` command

4.3.2. The `\etocifnumbered` switch

The `\fbox` would give an unaesthetic result in the case of an unnumbered section (which ended up in the table of contents via an `\addcontentsline` command).

The `\etocifnumbered{<A>}{}` command executes `<A>` if the number exists, and `` if not. So our final code could be:

```
\etocsetstyle{section}
{\begin{enumerate}[label=\etocifnumbered{\etocnumber}{}]}
{\normalsize\bfseries\rmfamily\item}
{\etocname (page \etocpage)}
{\end{enumerate}}
```

We used, actually:

```
\etocsetstyle{section}
{\begin{enumerate}[leftmargin=0pt,label=]}
{\normalsize\bfseries\rmfamily\item}
{\etocifnumbered{\makebox[1cm][c]{\color{green}\fboxrule1pt
\protect\normalcolor\etocnumber}}{\hspace{1cm}}}
\etocname (page \etocpage)}
{\end{enumerate}}
```

```
\etocsetstyle{subsection}
{\begin{enumerate}[label=\etocnumber, leftmargin=1cm]}
{\normalfont \item}
{\etocname (p.\~\etocpage)}
{\end{enumerate}}
```

If we had changed only the section level, and not the subsection level, an error on compilation would have occurred because the package style for subsections expects to start ‘in vertical mode’. An additional `\par` token in the `<contents>` part of the section level would have fixed this: `{... (page \etocpage)\par}`.

4.4. The `\etocfontxxx...` commands

The package predefines for use in its default line styles:

```
\newcommand\etocfontminustwo{\normalfont \LARGE \bfseries}
\newcommand\etocfontminusone{\normalfont \large \bfseries}
\newcommand\etocfontzero{\normalfont \normalsize \bfseries}
\newcommand\etocfontone{\normalfont \normalsize \bfseries}
\newcommand\etocfonttwo{\normalfont \normalsize}
\newcommand\etocfontthree{\normalfont \footnotesize}
\newcommand\etocfontfour{\normalfont \footnotesize}
\newcommand\etocfontfive{\normalfont \footnotesize}
```

The same font is used for the name, number, and page number, at each level (up to subsubsection).

5. Am I also in red?

In document classes where there is a `\chapter` command, the default font for the level 1 (‘section’) does not use the bold series.

These font commands only impact the package default line styles and do nothing under the `\etocstandardlines` regime. And, let us recall, levels four (paragraph) and five (subparagraphs) are actually not displayed by the package default line styles.

4.4.1. This is a (pale) red subsubsection for illustrative purposes

Just before this subsubsection we inserted in the `.tex` file:

```
\color{red!50}  
\addtocontents{toc}{\string\color{red!50}}
```

This entire subsubsection and its title is printed in red, as well as the title of the next section (we cancel the color change only after it). But what about its entry in the table of contents? well it is red in the main table of contents at the beginning of this document, which uses the package line styles, and it is not red in the table of contents at the beginning of this second document part. The reason is that the `\finish` code for the subsubsection level closed a group, whereas the `\start` and `\finish` code of the package line styles do not contain group opening and closing instructions. So here the group was closed while reading the `.toc` file and this explains why the next entry in the local TOC was not typeset in red.

In the present case the group at the subsubsection level is there to confine the font changes, but this could be made superfluous via adding a `\normalsize` to the subsection specification. I take this opportunity to mention that **etoc** always typesets the entire table of contents inside a group, hence no font change therein can propagate to the outside.

5. Am I also in red?

Well, actually no in this `scrartcl` class but I assure you I was in the `article` class!

```
\normalcolor  
\addtocontents{toc}{\string\normalcolor}
```

Back to black. Note that this scope problem arises in real life in a multi-lingual document, as the `babel` package writes to the `.toc` file the language changes occurring in the document. There is no easy general solution¹¹ to this, but it will probably concern only a minority of cases. Again, the line styles defined by **etoc** do not use groups.

An unnumbered section

I don’t have much to say here. I just added an unnumbered section, and made it appear in the `.toc` file.

```
\section*{An unnumbered section}  
\addcontentsline{toc}{section}{An unnumbered section}
```

¹¹one can insert by hand the language changes in each concerned sectioning command.

6. A comparison

The main table of contents was done in the `etoc` style. Let us issue here again two `\tableofcontents` commands, one with the document class style (with some special centered title in place of the standard Contents heading) and the other one with the style we defined in the previous part on the basis of enumerate environments.

```
\setcounter{secnumdepth}{4}
\setcounter{tocdepth}{4}
```

Let us use the standard formatting of entries, as is provided by the document class, and a two-column layout:

```
\etocstandardlines
\etocruledstyle{% 2 is the default number of columns
  \renewcommand{\etocprule}{\hrule height0pt}%
  \normalfont\normalsize\rmfamily\itshape
  \parbox{.8\linewidth}{\centering
    This is the global table of contents with standard
    (to the {\ttfamily scrartcl} class) entries
    (and on two columns). Contrarily to
    \hyperref[toc:test]{\color{green!35!blue!75}the next TOC}
    it contains paragraph entries.}}
\tableofcontents
```

This is the global table of contents with standard (to the `scrartcl` class) entries (and on two columns). Contrarily to [the next TOC](#) it contains paragraph entries.

I. Overview	3	3. Arbitrarily many TOCs, and local ones too	5
1. Initial motivation: nested lists	3	3.1. On manually adding layout commands to the <code>.toc</code> file	5
1.1. Limitations intrinsic to the use of environments	3	3.2. Labeling and reusing elsewhere	6
2. Line styles and global style	4	3.3. Local table of contents for this part	7
2.1. <code>\etocsettocstyle</code> for the toc display	4	II. Package commands for line styles	8
2.1.1. Multiple columns, ruled title, framed contents	4	4. The <code>\etocsetstyle</code> command	8
2.2. <code>\etocsetstyle</code> for the line styles	4	4.1. An example	8
2.2.1. Instructing <code>etoc</code> about the document sectional division names	5	4.1.1. <code>\etocname</code> and <code>\etocpage</code>	9
2.3. Compatibility mode	5		

6. A comparison

4.1.2. The <code>\etocskip-firstprefix</code> command	9	7.3. A point of some importance 18	
4.1.3. <code>secnumdepth</code> and <code>tocdepth</code>	10	7.3.0.2. Do we really want paragraph entries in the TOC? .	18
4.2. The <code>\etocsetlevel</code> command	10	7.3.0.3. really? .	18
4.3. Further informations and let's make this a very long title to see how it will look like	10	7.4. The command <code>\etocruled</code>	18
4.3.1. The <code>\etocnumber</code> command . .	10	7.5. The commands <code>\etocframedstyle</code> and <code>\etocframed</code>	18
4.3.2. The <code>\etocifnumbered</code> switch . . .	11	7.6. The command <code>\etocsettocstyle</code>	18
4.4. The <code>\etocfontxxx...</code> commands	11	7.7. The compatibility mode <code>\etocstandarddisplaystyle</code>	19
4.4.1. This is a (pale) red subsubsection for illustrative purposes	12	8. The starred variants of the <code>\tableofcontents</code> etc... commands	19
5. Am I also in red?	12	9. Table of contents for part III	19
An unnumbered section	12	IV. Using and customizing <code>etoc</code>	21
6. A comparison	13	10. Summary of the main style commands	22
III. Package commands for global styles	16	10.1. Setting up local styles . . .	22
7. Specifying the toc display style	16	10.2. Setting up global styles . .	22
7.1. The command <code>\etocruledstyle</code>	17	10.3. Displaying Tables of Contents	22
7.2. The command <code>\etocmulticolstyle</code>	17	10.4. Labels and references . . .	22
7.2.1. The command <code>\etocsettocstyle</code>	17	10.5. An example design	23
7.2.1.1. <code>\etocsettocstyle-with-marks</code> .	18	11. Customizing <code>etoc</code>	25
		11.1. Customizing the <code>etoc</code> line styles	25
		11.2. Customizing the global styles	26
		V. Tips	27
		12. ... and tricks	27
		12.1. Hacking framed parboxes .	27
		12.2. Intervverting the levels . . .	28
		12.3. All subsections of this document	28

12.4. Compatibility with other packages	29	VI. The code	30
12.5. T _E Xnical matters	29		
12.6. Errors and catastrophes . .	30	13. Implementation	30

We then switch to our set-up from the previous part and add to it:

```
\etocsetstyle{part}
{\begin{enumerate}[label=\etocnumber,leftmargin=1.5em]}
{\etocfontminusone\item}
{\etocname}
{\end{enumerate}}
\tableofcontents \label{toc:test} \ref{toc:main}
```

This is the global table of contents using quickly defined (and easily improvable) enumerate environments for part, section and subsection, and an ad hoc in-line italic style for subsubsection.

I Overview

1 Initial motivation: nested lists (page 3)

- 1.1 Limitations intrinsic to the use of environments (p. 3)

2 Line styles and global style (page 4)

- 2.1 `\etocsettocstyle` for the toc display (p. 4)

Multiple columns, ruled title, framed contents.

- 2.2 `\etocsetstyle` for the line styles (p. 4)

*Instructioning **etoc** about the document sectional division names.*

- 2.3 Compatibility mode (p. 5)

3 Arbitrarily many TOCs, and local ones too (page 5)

- 3.1 On manually adding layout commands to the `.toc` file (p. 5)
- 3.2 Labeling and reusing elsewhere (p. 6)
- 3.3 Local table of contents for this part (p. 7)

II Package commands for line styles

4 The `\etocsetstyle` command (page 8)

- 4.1 An example (p. 8)

`\etocname` and `\etocpage` – The `\etocskipfirstprefix` command – `secnum-depth` and `tocdepth`.

- 4.2 The `\etocsetlevel` command (p. 10)

- 4.3 Further informations and let's make this a very long title to see how it will look like (p. 10)

The `\etocnumber` command – The `\etocifnumbered` switch.

- 4.4 The `\etocfontxxx...` commands (p. 11)

This is a (pale) red subsubsection for illustrative purposes.

5 Am I also in red? (page 12) An unnumbered section (page 12)

6 A comparison (page 13)

III Package commands for global styles

7 Specifying the toc display style (page 16)

7.1 The command `\etocruled-style` (p. 17)

7.2 The command `\etocmulti-colstyle` (p. 17)

The command `\etocstyle`.

7.3 A point of some importance (p. 18)

7.4 The command `\etocruled` (p. 18)

7.5 The commands `\etocframed-style` and `\etocframed` (p. 18)

7.6 The command `\etocsettoc-style` (p. 18)

7.7 The compatibility mode `\etoc-standarddisplaystyle` (p. 19)

8 The starred variants of the `\tableofcontents` etc... commands (page 19)

9 Table of contents for part III (page 19)

IV Using and customizing `etoc`

10 Summary of the main style commands (page 22)

10.1 Setting up local styles (p. 22)

10.2 Setting up global styles (p. 22)

10.3 Displaying Tables of Contents (p. 22)

10.4 Labels and references (p. 22)

10.5 An example design (p. 23)

11 Customizing `etoc` (page 25)

11.1 Customizing the `etoc` line styles (p. 25)

11.2 Customizing the global styles (p. 26)

V Tips

12 ... and tricks (page 27)

12.1 Hacking framed parboxes (p. 27)

12.2 Interverting the levels (p. 28)

12.3 All subsections of this document (p. 28)

12.4 Compatibility with other packages (p. 29)

12.5 T_EXnical matters (p. 29)

12.6 Errors and catastrophes (p. 30)

VI The code

13 Implementation (page 30)

Part III.

Package commands for global styles

```
\setcounter{tocdepth}{-2}
\localtableofcontents \label{toc:globalcmds}
```

7. Specifying the toc display style

The *toc display* style says whether the TOC appears with multiple columns or just one, whether the title is typeset as in the `article` or `book` class, or should be centered above

7. Specifying the toc display style

the entries, with rules on its sides, or if the entire TOC should be put in a frame. For example, to opt for a ruled heading and single column layout, one issues commands of the following type:

```
\etocruledstyle[1]{Title}
  \tableofcontents (or \localtableofcontents)
shortcuts:
  \etocruled[1]{Title} (or \etoclocalruled[1]{Title})
```

7.1. The command `\etocruledstyle`

The general format of `\etocruledstyle` is:

```
\etocruledstyle[<number of columns>]{<title of the toc>}
```

Note that the title is horizontal material, if it does not fit on one line it should be put in a `\parbox` of a given width. We did this and even enclosed the parboxes in `\fbox`s to get frames around them. For the example with the standard formatting we did not use an `\fbox` and got rid of the horizontal rules via:

```
\renewcommand{\etocstoprule}{\hrule height 0pt}
```

The green frame for the heading of the table of contents at the [start of the second part of this document](#) was obtained with:

```
\etocruledstyle[1]{\etocfontminusone\color{green}
  \fboxrule1pt\fboxseplex
  \framebox[\linewidth]
    {\normalcolor\hss Contents of this second part\hss}}
```

7.2. The command `\etocmulticolstyle`

This is also a command with one optional and one mandatory argument:

```
\etocmulticolstyle[<number_of_columns>]{<heading>}
```

The *<number_of_columns>* can go from 1 to 10 (it defaults to 2, and from 2 on is passed to a `multicols` environment). The *<heading>* should be some ‘vertical’ material like:

```
<heading> = \section*{<title>}
```

After such a command, future `\tableofcontents` will use the specified style. A shortcut for just one table of contents and not affecting the styles of later TOCs is:

```
\etocmulticol[<number_of_columns>]{<heading>}
```

And there is also `\etoclocalmulticol[<number_of_columns>]{<heading>}`.

7.2.1. The command `\etoc tocstyle`

```
\etoc tocstyle[<kind>]{<number_of_columns>}{<title>}
= \etocmulticolstyle[number_of_columns]{\kind*{title}}
```

where *kind* is one of `chapter`, `section`, ... and defaults to `chapter` or `section` depending on the document class.

7. Specifying the toc display style

7.2.1.1. `\etocstylewithmarks` [*<kind>*] {*<number_of_columns>*} {*<title>*} {*<mark>*}
=`\etocmulticolstyle` [*number_of_columns*] {*\kind** {*title* \markboth{\MakeUppercase{*mark*}}}}
where *kind* is one of `chapter`, `section`, ... The actual display of the marks depends on the settings of the page style. There is variant `\etocstylewithmarksnouc` which does not uppercase.

7.3. A point of some importance

There is an important difference between the `\etocmulticolstyle` and `\etocruledstyle` commands. The mandatory argument of the former can not be just some *naked text*. One can use things such as

`\section*{some not so naked text}`

but `<heading>=naked text` gives an error:

ERROR: LaTeX Error: Something's wrong--perhaps a missing \item.

In the special case where the command was also given the optional argument [1], it is possible to feed it with `'naked text\par'`.

On the other hand, it is exactly the *opposite* for `\etocruledstyle` (and `\etocframedstyle`) which expects things which fit inside a horizontal box, hence it will gladly accept a *naked text* as argument, but will choke on `\section*` or things ending with a `\par`.

7.3.0.2. Do we really want paragraph entries in the TOC?

7.3.0.3. really?

7.4. The command `\etocruled`

As a shortcut to set the style and issue a `\tableofcontents`, all inside a group so that future table of contents will not be affected, there is:

`\etocruled` [*<number_of_columns>*] {*<title>*}

And the local form will be `\etoclocalruled`.

7.5. The commands `\etocframedstyle` and `\etocframed`

Same mechanism:

`\etocframedstyle` [*<number_of_columns>*] {*<title>*}

and the accompanying shortcut:

`\etocframed` [*<number_of_columns>*] {*<title>*}

Here the entire table of contents is framed, hence this can only work if it fits on a page. Note that the title itself is not framed, if one wants a frame one should set it up inside the *<title>* argument to `\etocframedstyle` or `\etocframed`. There is also `\etoclocalframedstyle` and `\etoclocalframed`.

7.6. The command `\etocsettocstyle`

This is a command with two mandatory arguments:

`\etocsettocstyle` {*<before_toc>*} {*<after_toc>*}

The {*<before_toc>*} part is responsible for typesetting the heading, for example it can be

8. The starred variants of the `\tableofcontents` etc... commands

something like `\section*{\contentsname}`. It can also contain instructions to mark the page headings. Or it could check (book class) to see if two-column mode is on, and switch to one-column style, and the `\after_toc` part would then reenact the two-column mode.

The previously described commands `\etocmulticolstyle`, `\etocruledstyle`, and `\etocframedstyle` actually call `\etocsettocstyle` as a lower-level routine.

7.7. The compatibility mode `\etocstandarddisplaystyle`

`etoc` will then emulate what the document class would have done regarding the global display style of the table of contents, in its absence. All customizing from inside the class should be obeyed, too.

8. The starred variants of the `\tableofcontents` etc... commands

The `\tableofcontents`, `\localtableofcontents`, `\etocmulticol`, etc... have starred variants (the star must be before the other arguments). For all but the `memoir` class, they are like the original. For the `memoir` class, the original prints an entry in the `.toc` file, as is the usage for the original `\tableofcontents` command in that class, whereas the starred variants do not, as is the habit in that class.

As soon as one starts using local table of contents one discovers that the default `memoir` thing which is to create a chapter entry for each TOC is not convenient. The command `\etocmemoirtotocfmt{<kind>}{<name>}` will change the format (`<kind>` is `chapter`, `section`, `subsection`... and `<name>` can be for example `\contentsname`.) The initial set-up is with `chapter` and `\contentsname`.

The format of the actual heading of the TOC should also be set appropriately (for example with `\etocstyle`), to use the identical division unit as in the first argument to `\etocmemoirtotocfmt`.

A weird situation arises when one has two successive `\localtableofcontents` (obviously this is not a truly real life situation), just after a `\part` for example. The first one creates (if the default has not been modified as indicated above) a Chapter heading which is written to the `.toc`. Then the second one thinks to be local to this chapter . . . and as a result it displays nothing. The fix is to define the second one to be a clone of the first one.

Independently of the situation with the `memoir` class there is generally speaking a hook macro called `\etocaftertitlehook` which is inhibited by using the starred variants of the displaying commands. Except for the `memoir` class, this hook is initially defined to do nothing.

9. Table of contents for part III

As a third example we now print the local table of contents for this part. We opt for a “framed” style, and specify some colors (including a background color). We modify various things from the package defaults. The original was invisibly defined with a label at the beginning of this part III.

```
\etoclines
```

9. Table of contents for part III

```
\begingroup % we use a group to limit the scope of the next commands
\renewcommand{\etoccolumnsep}{3em}
\renewcommand{\etocinnerleftsep}{2em}
\renewcommand{\etocinnerrightsep}{2em}
% specify a background color for the toc contents
\renewcommand{\etocbkgcolorcmd}{\color{yellow!10}}
%\renewcommand{\etocbkgcolorcmd}{\relax}
% set up the top and bottom rules
\renewcommand{\etoctoprule}{\hrule height 1pt}
\renewcommand{\etoctoprulecolorcmd}{\color{red!25}}
\renewcommand{\etocbottomrule}{\hrule height 1pt}
\renewcommand{\etocbottomrulecolorcmd}{\color{red!25}}
% set up the left and right rules
\renewcommand{\etocleftrule}{\vrule width 5pt}
\renewcommand{\etocrightrule}{\vrule width 5pt}
\renewcommand{\etocleftrulecolorcmd}{\color{red!25}}
\renewcommand{\etocrightrulecolorcmd}{\color{red!25}}
% use \fcolorbox to set up a colored frame for the title
\fbboxrule1pt
\etocframedstyle{\normalsize\rmfamily\itshape
  \fcolorbox{red}{white}{\parbox{.8\linewidth}{\centering
    This is a table of contents \a la \etoc, but just for
    the sections and subsections in this part. As it is put
    in a frame, it has to be small enough to fit on the
    current page. It has the label |toc:b|.}}}
% set up a label for future (or earlier...) reference
\tableofcontents \label{toc:b} \ref{toc:globalcmds}
\endgroup
```

*This is a table of contents à la **etoc**, but just for the sections and subsections in this part. As it is put in a frame, it has to be small enough to fit on the current page. It has the label `toc:b`.*

Specifying the toc display style
 7, p. 16

The command `\etocruledstyle`
 7.1, p. 17

The command `\etocmulticol-style` 7.2, p. 17

The command `\etocstyle`.

A point of some importance 7.3, p. 18

The command `\etocruled` 7.4, p. 18

The commands `\etocframed-style` and `\etocframed` 7.5, p. 18

The command `\etocsettocstyle`
 7.6, p. 18

The compatibility mode `\etoc-standarddisplaystyle` 7.7, p. 19

The starred variants of the `\table-ofcontents` etc... commands
 8, p. 19

Table of contents for part III 9, p. 19

We can also display it in the document standard manner:¹²

```
\KOMAoptions{toc=left}
\etocstandarddisplaystyle % necessary for the display to obey toc=left
\etocstandardlines
\tableofcontents \ref{toc:globalcmds}
```

Contents

7. Specifying the toc display style	16
7.1. The command <code>\etocruledstyle</code>	17
7.2. The command <code>\etocmulticolstyle</code>	17
7.2.1. The command <code>\etoctocstyle</code>	17
7.3. A point of some importance	18
7.4. The command <code>\etocruled</code>	18
7.5. The commands <code>\etocframedstyle</code> and <code>\etocframed</code>	18
7.6. The command <code>\etocsettocstyle</code>	18
7.7. The compatibility mode <code>\etocstandarddisplaystyle</code>	19
8. The starred variants of the <code>\tableofcontents</code> etc... commands	19
9. Table of contents for part III	19

Part IV.

Using and customizing **etoc**

This is a table of contents for the sections and subsections in this part. It carries the label `toc:c`

Summary of the main style commands	An example design	10.5, p. 23
10, p. 22		
Setting up local styles	Customizing etoc	11, p. 25
10.1, p. 22		
Setting up global styles	Customizing the etoc line styles	11.1, p. 25
10.2, p. 22		
Displaying Tables of Contents	Customizing the global styles	11.2, p. 26
10.3, p. 22		
Labels and references		
10.4, p. 22		

¹²the present document uses the `scrartcl` class, and we check here that the **etoc** compatibility mode does respect the customizing done via the class commands.

10. Summary of the main style commands

10.1. Setting up local styles

```
\etocsetstyle{<levelname>}{<start>}{<prefix>}{<contents>}{<finish>}
\etocname, \etocnumber, \etocpage, \etocifnumbered{<A>}{<B>}
```

10.2. Setting up global styles

```
\etocmulticolstyle[<number_of_columns>]{<heading>}
\etocstyle[<kind>]{<number_of_columns>}{<title>}
\etocstylewithmarks[<kind>]{<number_of_columns>}{<title>}{<mark>}
\etocstylewithmarksnouc[<kind>]{<number_of_columns>}{<title>}{<mark>}
\etocruledstyle[<number_of_columns>]{<title>}
\etocframedstyle[<number_of_columns>]{<title>}
\etocsettocstyle{<before_toc>}{<after_toc>}
```

10.3. Displaying Tables of Contents

```
\tableofcontents
\localtableofcontents
\etocmulticol[<number_of_columns>]{<heading>}
\etoclocalmulticol[<number_of_columns>]{<heading>}
\etocruled[<number_of_columns>]{<title>}
\etoclocalruled[<number_of_columns>]{<title>}
\etocframed[<number_of_columns>]{<title>}
\etoclocalframed[<number_of_columns>]{<title>}
and their starred variants
```

10.4. Labels and references

The commands (starred or not) to actually display the table of contents can be followed with optional labels or references:

```
\tableofcontents \label{toc:here}
\tableofcontents \ref{toc:far}
\tableofcontents \label{toc:here} \ref{toc:far}
\localtableofcontents \label{toc:here}
similarly with \etocmulticol etc . . .
```

The commands for local tables of contents do *not* react to a `\ref` following them.

When re-displaying another toc, only its contents are transferred: both the line styles and the global style are the ones currently defined, not the ones from the cloned toc.

10.5. An example design

I got motivated by a question¹³ I saw on the T_EX StackExchange site. I copied the color RGB specifications from an answer which had been provided to the question. The `\etoc-framedstyle` puts the title on the top rule in a centered position. This is not very convenient for this example so we included the title as part of the `\start` code at section level, to get it *inside* the frame.

```
\setcounter{tocdepth}{3}
\begingroup
\definecolor{subsecnum}{RGB}{13,151,225}
\definecolor{secbackground}{RGB}{0,177,235}
\definecolor{tocbackground}{RGB}{212,237,252}

\renewcommand{\etocbkgcolorcmd}{\color{tocbackground}}
\renewcommand{\etocleftrulecolorcmd}{\color{tocbackground}}
\renewcommand{\etocrightrulecolorcmd}{\color{tocbackground}}
\renewcommand{\etocbottomrulecolorcmd}{\color{tocbackground}}
\renewcommand{\etocoprerulecolorcmd}{\color{tocbackground}}

\renewcommand{\etocleftrule}{\vrule width 1cm}
\renewcommand{\etocrightrule}{\vrule width .5cm}
\renewcommand{\etocbottomrule}{\hrule height 12pt}
\renewcommand{\etocoprerule}{\hrule height 12pt}

\renewcommand{\etocinnertopsep}{0pt}
\renewcommand{\etocinnerbottomsep}{0pt}
\renewcommand{\etocinnerleftsep}{0pt}
\renewcommand{\etocinnerrightsep}{0pt}

\newcommand\shiftedwhiterule[1]{%
\hbox to \linewidth{\color{white}%
\hskip\leftskip\leaders\vrule height1pt\hfil}\nointerlineskip
\vskip#1}

\etocsetstyle{subsubsection}{}
{\shiftedwhiterule{6pt}}
{\sffamily\footnotesize
\leftskip2.5cm\noindent
\hbox to 1cm{\color{subsecnum}\etocnumber\hskip.2cm}%
\color{black}\etocname\hfill\hbox{\etocpage\hskip.2cm}\par
\nointerlineskip\vskip3pt}
{}

\etocsetstyle{subsection}{\etocskipfirstprefix}
{\shiftedwhiterule{6pt}}
{\sffamily\small
\leftskip1.5cm\noindent
\hbox to 1cm{\color{subsecnum}\etocnumber\hskip.2cm}%
\color{black}\etocname\hfill\hbox{\etocpage\hskip.2cm}\par
\nointerlineskip\vskip3pt}
{}

\newcommand{\coloredstuff}[2]{%
\leftskip0pt\rightskip0pt\parskip0pt
```

¹³tex.stackexchange.com/questions/83184

10. Summary of the main style commands

```

\fbxsep0pt % \colorbox uses \fbxsep!
\noindent\colorbox{secbackground}
{\parbox{\linewidth}{%
\vskip5pt
{\noindent\color{#1}#2\par\nointerlineskip}
\vskip3pt}}%
\par\nointerlineskip}

\etocsetstyle{section}{\coloredstuff{white}
{\hfil \hyperref[toc:b]{\bfseries\large I am a twin of
that other TOC (click me!)}\hfil}}
{\vskip3pt\sffamily\small}
{\coloredstuff{white}{\hbox to 1.5cm{\hss\etocnumber\hskip.2cm}%
\etocname\hfill\hbox{\etocpage\hskip.2cm}}}%
\vskip6pt}
{}
\etocframedstyle[1]{} % title is not there!
\tableofcontents \label{toc:clone} \ref{toc:globalcmds}
\endgroup

```

The coding is a bit involved as it does not use any additional package. A better solution would be to use some package to set up a background color possibly extending accross pages, as the framed style (which we used to get this background color) can only deal with material short enough to fit on one page.

Regarding colors, generally speaking all color commands inside **etoc** are initially defined to do nothing, and the choice to use or not colors is left to the user.

I am a twin of that other TOC (click me!)		
7	Specifying the toc display style	16
7.1	The command <code>\etocruledstyle</code>	17
7.2	The command <code>\etocmulticolstyle</code>	17
7.2.1	The command <code>\etoc tocstyle</code>	17
7.3	A point of some importance	18
7.4	The command <code>\etocruled</code>	18
7.5	The commands <code>\etocframedstyle</code> and <code>\etocframed</code>	18
7.6	The command <code>\etocsettocstyle</code>	18
7.7	The compatibility mode <code>\etocstandarddisplaystyle</code>	19
8	The starred variants of the <code>\tableofcontents</code> etc... commands	19
9	Table of contents for part III	19

11. Customizing **etoc**

11.1. Customizing the **etoc** line styles

We will simply list the relevant commands as defined in the package. Customizing them goes through suitable `\renewcommands`:

```
\newcommand\etocfontminustwo{\normalfont \LARGE \bfseries}
\newcommand\etocfontminusone{\normalfont \large \bfseries}
\newcommand\etocfontzero{\normalfont \normalsize \bfseries}
\newcommand\etocfontone{\normalfont \normalsize \bfseries}
\newcommand\etocfonttwo{\normalfont \normalsize}
\newcommand\etocfontthree{\normalfont \footnotesize}
\newcommand\etocfontfour{\normalfont \footnotesize}
\newcommand\etocfontfive{\normalfont \footnotesize}

\newcommand\etocsepminustwo{4ex plus .5ex minus .5ex}
\newcommand\etocsepminusone{4ex plus .5ex minus .5ex}
\newcommand\etocsepzero{2.5ex plus .4ex minus .4ex}
\newcommand\etocsepone{1.5ex plus .3ex minus .3ex}
\newcommand\etocseptwo{1ex plus .15ex minus .15ex}
\newcommand\etocsepthree{.25ex plus .05ex minus .05ex}

\newcommand\etocminustwoleftmargin{1.5em plus 0.5fil}
\newcommand\etocminustworightmargin{1.5em plus -0.5fil}
\newcommand\etocminusoneleftmargin{1em}
\newcommand\etocminusonerightmargin{1em}

\newcommand\etocbaselinespreadminustwo{1}
\newcommand\etocbaselinespreadminusone{1}
\newcommand\etocbaselinespreadzero{1}
\newcommand\etocbaselinespreadone{1}
\newcommand\etocbaselinespreadtwo{1}
\newcommand\etocbaselinespreadthree{.9}

\newcommand\etococlineleaders
{\hbox{\normalfont\normalsize\hbox to 2ex {\hss.\hss}}}}

\newcommand\etocabbrevpagename{p.~}
\newcommand\etocpartname{\partname}
\newcommand\etocbookname{Book}
```

No customizing of the standard line styles is possible from within **etoc**. As already explained, when `\etocstandardlines` has been issued, the package just makes itself very discrete and acts only at the global level, and the TOC entries are (hopefully) formatted as would have happened in the absence of **etoc**.¹⁴

¹⁴with the KOMA-script classes, we noticed that `\etocstandarddisplaystyle` was apparently needed for the KOMA options `toc=left` to be active at the level of the line entries.

The `\etocstandardlines` mechanism will work also with sectioning commands made known to *etoc* via `\etocsetlevel`, under the condition of course that these sectioning commands are accompanied with all the relevant definitions for typesetting toc entries in the L^AT_EX default manner (existence of the macros `\l@something . . .`).

Using the `\etoc toc lines` (obviously) or even just one `\etocsetstyle` command turns off the compatibility mode (all of this obey the scoping mechanism of T_EX/L^AT_EX groups).

11.2. Customizing the global styles

Again we list the relevant macros, what they do should be legible from their names. Note that `\renewcommand`'s and not `\setlength`'s have to be used for what appear to be lengths, and that color commands are not just color specifications, they must include `\color`, and are canceled by re-defining them to do `\relax`.

```
\newcommand\etocabovetocskip{3.5ex plus 1ex minus .2ex}
\newcommand\etocbelowtocskip{3.5ex plus 1ex minus .2ex}
```

```
\newcommand\etoccolumnsep{2em}
\newcommand\etocmulticolsep{0ex}
\newcommand\etocmulticolpretolerance{-1}
\newcommand\etocmulticoltolerance{200}
\newcommand\etocdefaultnbcol{2}
\newcommand\etocinnertopsep{2ex}
\newcommand\etoc top rule{\hrule}
\newcommand\etoc top rule color cmd{\relax}
```

% for the framed style only:

```
\newcommand\etocinnerleftsep{2em}
\newcommand\etocinnerrightsep{2em}
\newcommand\etocinnerbottomsep{3.5ex}
```

```
\newcommand\etocleftrule{\vrule}
\newcommand\etocrightrule{\vrule}
\newcommand\etocbottomrule{\hrule}
\newcommand\etocleftrule color cmd{\relax}
\newcommand\etocrightrule color cmd{\relax}
\newcommand\etocbottomrule color cmd{\relax}
```

```
\newcommand\etocbkg color cmd{\relax}
```

% hooks

```
\newcommand\etoc framed mp hook{\relax}
\newcommand\etoc after title hook{\relax}
```

`\etoc after title hook` was commented upon before, and `\etoc framed mp hook` is positioned immediately after the beginning of a minipage environment where the contents of the framed TOC are typeset.

The `\...colorcmd` things are initially set to be `\relax`, so there is no need to do `\usepackage{color}` if the document does not use colors. If the scope of a change to a color command such as `\etocbkgcolorcmd` has not been limited to a group and one then wishes to let it again be `\relax` one must use a `\renewcommand` and not `\let\etocbkgcolorcmd\relax`.

Regarding the dimensions of the top rule they can be specified in ex's or em's as in this example:

```
\renewcommand{\etoc_toprule}{\hrule height 1ex}
```

The package code is done in such a manner that it is the font size in instance at the end of typesetting the title argument to `\etocruledtoc` or `\etocframedtoc` which will be used for the meaning of the '1ex'. Of course also the other rule commands can have their dimensions in font relative units, but their values are decided on the basis of the font in effect just before the table of contents.

The top and bottom rules do not have to be rules and can be horizontal *leaders* (of a specified height) in the general T_EX sense. However the left and right rules are not used as (horizontal) leaders but as objects of a given specified width. Note that *only* the Plain T_EX syntax for rules is accepted here.

Part V.

Tips

12. ... and tricks

12.1. Hacking framed parboxes

```
\renewcommand\etoc_toprule{\hrule height 2pt depth 2pt}
\etocruled{\color{green}\fboxrule2pt\fboxsep1ex
  \fbox{\raisebox{-\fontdimen22\textfont2}
    {\color{blue}\parbox{.5\linewidth}
      {\normalfont This text is perfectly centered
        vertically with respect to the
        surrounding horizontal rules.}}}}
\ref{toc:globalcmds}
```

This text is perfectly centered vertically with respect to the surrounding horizontal rules.

Specifying the toc display style . . .
 **7, p. 16**

The command `\etocruledstyle` . . .
 7.1, p. 17

The command `\etocmulticolstyle`
 7.2, p. 17

The command `\etoc tocstyle`.

A point of some importance . 7.3, p. 18

The command `\etocruled` . 7.4, p. 18

The commands `\etocframedstyle` and
`\etocframed` 7.5, p. 18

The command <code>\etocsettocstyle</code>	7.6, p. 18	The starred variants of the <code>\tableof-contents</code> etc... commands	8, p. 19
The compatibility mode <code>\etocstandard-displaystyle</code>	7.7, p. 19	Table of contents for part III	9, p. 19

12.2. Interverting the levels

Let us display and count all subsections occurring in this document.

```
\setcounter{tocdepth}{2}
\etocsetlevel{part}{3}
\etocsetlevel{section}{3}
\etocsetstyle{subsection}{\begin{enumerate}[itemsep=0pt,%
label=,leftmargin=0pt]}
{\normalfont\bfseries\item}
{\roman{enumi}. \normalfont\etocname (\etocnumber, p.\~\etocpage)}
{\end{enumerate}}
\renewcommand{\etoccolumnsep}{2.75em}
\renewcommand{\columnseprule}{1pt}
\etocmarkbothnouc{List of all subsections}
\etocmulticol[3]{\subsection{All subsections of this document}}
```

12.3. All subsections of this document

i. Limitations intrinsic to the use of environments (1.1, p. 3)	p. 8)	xvi. The commands <code>\etocframedstyle</code> and <code>\etocframed</code> (7.5, p. 18)
ii. <code>\etocsettocstyle</code> for the toc display (2.1, p. 4)	ix. The <code>\etocsetlevel</code> command (4.2, p. 10)	xvii. The command <code>\etocsettocstyle</code> (7.6, p. 18)
iii. <code>\etocsetstyle</code> for the line styles (2.2, p. 4)	x. Further informations and let's make this a very long title to see how it will look like (4.3, p. 10)	xviii. The compatibility mode <code>\etocstandard-displaystyle</code> (7.7, p. 19)
iv. Compatibility mode (2.3, p. 5)	xi. The <code>\etocfontxxx...</code> commands (4.4, p. 11)	xix. Setting up local styles (10.1, p. 22)
v. On manually adding layout commands to the .toc file (3.1, p. 5)	xii. The command <code>\etocruledstyle</code> (7.1, p. 17)	xx. Setting up global styles (10.2, p. 22)
vi. Labeling and reusing elsewhere (3.2, p. 6)	xiii. The command <code>\etocmulticolstyle</code> (7.2, p. 17)	xxi. Displaying Tables of Contents (10.3, p. 22)
vii. Local table of contents for this part (3.3, p. 7)	xiv. A point of some importance (7.3, p. 18)	xxii. Labels and references (10.4, p. 22)
viii. An example (4.1,	xv. The command <code>\etocruled</code> (7.4, p. 18)	xxiii. An example design (10.5, p. 23)
		xxiv. Customizing the

etoc line styles (11.1, p. 25)	xxvii. Interverting the levels (12.2, p. 28)	p. 29)
xxv. Customizing the global styles (11.2, p. 26)	xxviii. All subsections of this document (12.3, p. 28)	xxx. T _E Xnical matters (12.5, p. 29)
xxvi. Hacking framed parboxes (12.1, p. 27)	xxix. Compatibility with other packages (12.4,	xxxi. Errors and catastrophes (12.6, p. 30)

12.4. Compatibility with other packages

etoc uses the packages `multicol` and `xspace`. It is `hyperref` aware and hopefully `hyperref` compatible! It doesn't matter whether **etoc** or `hyperref` is loaded first.

The macros `\etocname`, `\etocnumber`, and `\etocpage` contain the `hyperref` links, if present (note that the `linktoc=all` option of `hyperref` tells it to put a link also in the page number corresponding to a given toc entry). The tables of contents of the present document are fully linked.

The contents of the `.toc` file are read into memory by **etoc** once, at the `\begin{document}` (and not at the first `\tableofcontents`). The input stream opened by **etoc** for reading is then closed but the `.toc` file is of course left opened for write operations, with the exact same instructions as used by the standard L^AT_EX classes.

When a `\localtableofcontents` is inserted by the user in the document, a line containing an **etoc** inner command and an identification number is added to the `.toc` file. This is the only kind of data written by **etoc** to the `.toc` file (of course it does mean that if you remove usage of **etoc** from the document, you should trash the `.toc` file). The correct local table of contents will be displayed only on the next latex run.

If the document class does not use the extension `.toc` for the file where the TOC data is preserved, then **etoc** will fail. If the data written in the `.toc` file does not contain the name, number and page numbers in `\contentsline` lines, then **etoc** will fail. If the typesetting of entries of type *kind* is not done by a macro with name `\l@kind`, then **etoc** will fail. If these macros are redefined after the `\begin{document}` then **etoc** will fail. However if they are defined before and **etoc** is instructed of their level by commands such as `\etocsetlevel{kind}{2}` then `\etocstandardlines` should be functional. You may customize the standard styles throughout the document (with `tocloft` for example) and **etoc** should respond in compatibility mode as expected.

12.5. T_EXnical matters

The `\etocname`, `\etocnumber`, `\etocpage` and `\etocifnumbered` macros are protected against premature expansion. They end with an `\xspace`. The style customizing commands (local and global) such as `\etocsetstyle`, `\etocsetlevel`, `\etocsettocstyle`, `\etocmulticolstyle`, `\etocruledstyle`, `\etocframedstyle` do obey L^AT_EX's groups. All TOCs are typeset inside groups.

When using multiple `\tableofcontents` commands in a document, one should beware from adding typesetting instructions directly in the `.toc` file, as they will be executed by **etoc** for all TOCs: even for a `\localtableofcontents` it doesn't matter if that instruction seems to concern material outside of its scope, it will get executed nevertheless.

If absolutely necessary these instructions should be done in such a way that they can be activated or deactivated easily from the document source, as need be.

12.6. Errors and catastrophes

After using `\etocsetstyle` for one level, the remaining uncustomized levels use the **etoc**'s default styles. One has to make sure that all levels needed for the next table of contents are mutually compatible.

Do not clone a clone but produce rather twins by cloning the same original local table of contents.

As is usual with toc and labels, after each change, one has to run latex a certain number of times to let the produced document get its final appearance (at least twice).

In certain extreme cases (*e.g.* if one removes the `hyperref` package from a previously hyperlinked document), one must trash the previous `.aux` and `.toc` files. But adding `\usepackage{hyperref}` to a non-hyperlinked source should cause no trouble at all (the old `.toc` file will not be used, the new one, whose format differs, will be used on the next run).

Part VI.

The code

This source file `etoc.dtx` produces `etoc.sty` when one does `latex etoc.dtx` or `pdflatex etoc.dtx` (an `etoc.ins` file is also produced, for distributions expecting it for installation). Two more runs are necessary to finish producing the documentation. The `etoc.sty` file should be moved to a suitable location within the \TeX installation.

13. Implementation

Writing-up source code comments is hopefully for a future release.

```
1 \ProvidesPackage{etoc}
2 [2012/12/01 v1.05 easily customizable TOCs (jfb)]
3 \NeedsTeXFormat{LaTeX2e}
4 \RequirePackage{multicol}
5 \RequirePackage{xspace}
6 \DeclareOption*{\PackageWarning{etoc}{Option '\CurrentOption' is unknown.}}
7 \ProcessOptions\relax

placeholder for comments

8 \newtoks\Etoc@toctoks
9 \newif\ifEtoc@jj % book
10 \newif\ifEtoc@j % part
11 \newif\ifEtoc@ % chapter
12 \newif\ifEtoc@i % section
13 \newif\ifEtoc@ii % subsection
14 \newif\ifEtoc@iii % subsubsection
```

13. Implementation

```
15 \newif\ifEtoc@iv % paragraph
16 \newif\ifEtoc@v % subparagraph
17 \newif\ifEtoc@number
18 \newif\ifEtoc@hyperref
19 \newif\ifEtoc@tocwithid
20 \newif\ifEtoc@standard
21 \newif\ifEtoc@part

placeholder for comments
22 \newif\ifEtoc@localtoc
23 \newif\ifEtoc@skipthisone
24 \newif\ifEtoc@stoptoc
25 \newif\ifEtoc@notactive
26 \newcounter{etoc@tocid}
27 \newif\ifEtoc@mustclosegroup

placeholder for comments
28 \@ifclassloaded{memoir}{\def\Etoc@minf{-\thr@@}}{\def\Etoc@minf{-\tw@}}
29 \def\Etoc@@@minustwo@@{-\tw@}
30 \let\Etoc@@@minusone@@\m@ne
31 \chardef\Etoc@@@zero@@\z@
32 \chardef\Etoc@@@one@@\@ne
33 \chardef\Etoc@@@two@@\tw@
34 \chardef\Etoc@@@three@@\thr@@
35 \chardef\Etoc@@@four@@ 4
36 \chardef\Etoc@@@five@@ 5
37 \chardef\Etoc@@@six@@ 6
38 \let\Etoc@localtop\Etoc@@@minustwo@@
39 \def\Etoc@@@minustwo@{minustwo}
40 \def\Etoc@@@minusone@{minusone}
41 \def\Etoc@@@zero@{zero}
42 \def\Etoc@@@one@{one}
43 \def\Etoc@@@two@{two}
44 \def\Etoc@@@three@{three}
45 \def\Etoc@@@four@{four}
46 \def\Etoc@@@five@{five}

placeholder for comments
47 \def\Etoc@Lazarus{%
48 \def\Etoc@Lazarus@elta##1\Etoc@Lazarus@eltb##2{\let##1##2}%
49 \Etoc@Lazaruslist}
50 \def\Etoc@Lazaruslist{}

placeholder for comments
51 \def\Etoc@newdisciple#1#2{%
52 \def\Etoc@Lazarus@elta{\noexpand\Etoc@Lazarus@elta\noexpand}%
53 \def\Etoc@Lazarus@eltb{\noexpand\Etoc@Lazarus@eltb\noexpand}%
54 \edef\Etoc@Lazaruslist{\Etoc@Lazaruslist
55 \Etoc@Lazarus@elta#1
56 \Etoc@Lazarus@eltb#2}}
57 \def\etocsetlevel#1#2{%
58 \let\Etoc@next\@firstofone
59 \ifcase#2\or\or\or\or\or
60 \else\ifnum#2=\m@ne\else\ifnum#2=-\tw@\else
```

13. Implementation

```

61 \PackageWarning{etoc}
62 {unexpected value ‘#2’ in \string\etocsetlevel.^^}%
63 Should be -2,-1, 0, 1, 2, 3, 4 or 5. Set to -1}%
64 \expandafter\def\csname Etoc@#1@\endcsname{minusone}%
65 \expandafter\let\csname Etoc@#1@\endcsname\m@ne
66 \let\Etoc@next\@gobble\fi\fi\fi
67 \if@noskipsec
68 \else
69 \expandafter\expandafter\expandafter
70 \let\expandafter\expandafter\csname Etoc@savdl@#1\endcsname
71 \csname l@#1\endcsname
72 \expandafter\let \csname l@#1\endcsname\Etoc@lxyz
73 \fi
74 \expandafter\expandafter\expandafter\Etoc@newdisciple
75 \expandafter\expandafter
76 \csname l@#1\endcsname\csname Etoc@savdl@#1\endcsname
77 \Etoc@next
78 {\ifcase#2\relax
79 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@zero@@
80 \expandafter\def \csname Etoc@#1@\endcsname{zero}%
81 \or
82 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@one@@
83 \expandafter\def \csname Etoc@#1@\endcsname{one}%
84 \or
85 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@two@@
86 \expandafter\def \csname Etoc@#1@\endcsname{two}%
87 \or
88 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@three@@
89 \expandafter\def \csname Etoc@#1@\endcsname{three}%
90 \or
91 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@four@@
92 \expandafter\def \csname Etoc@#1@\endcsname{four}%
93 \or
94 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@five@@
95 \expandafter\def \csname Etoc@#1@\endcsname{five}%
96 \else
97 \ifnum#2=\m@ne
98 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@minusone@@
99 \expandafter\def \csname Etoc@#1@\endcsname{minusone}%
100 \else
101 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@minustwo@@
102 \expandafter\def \csname Etoc@#1@\endcsname{minustwo}%
103 \fi
104 \fi}}
105 \etocsetlevel{book}{-2}
106 \etocsetlevel{part}{-1}
107 \etocsetlevel{chapter}{0}
108 \etocsetlevel{section}{1}
109 \etocsetlevel{subsection}{2}
110 \etocsetlevel{subsubsection}{3}
111 \etocsetlevel{paragraph}{4}
112 \etocsetlevel{subparagraph}{5}

```


13. Implementation

placeholder for comments

```
113 \AtBeginDocument{%
114 \def\Etoc@Lazarus@elta#1\Etoc@Lazarus@eltb#2{\let#2#1\let#1\Etoc@lxyz}%
115 \Etoc@Lazaruslist
116 \let\Etoc@savedcontentsline\contentsline
117 \let\contentsline\Etoc@etoccontentsline
118 \@ifpackageloaded{hyperref}
119 {\Etoc@hyperreftrue
120 \def\Etoc@et@hop#1#2#3#4#5{#1{#3}{#4}{#5}#2}}
121 {\Etoc@hyperreffalse
122 \def\Etoc@et@hop#1#2#3#4{#1{#3}{#4}#2}}}
```

placeholder for comments

```
123 \def\Etoc@swa#1{%
124 \Etoc@et@hop
125 {\Etoc@savedcontentsline{#1}}
126 {\Etoc@prefix\Etoc@contents}}
127 \def\Etoc@swb#1{%
128 \Etoc@et@hop
129 {\Etoc@savedcontentsline{#1}}
130 {\Etoc@contents}}
131 \let\etocskipfirstprefix\@thirdofthree
```

placeholder for comments

```
132 \def\Etoc@setflags #1{%
133 \ifcase #1\relax
134 \global\Etoc@vfalse
135 \global\Etoc@ivfalse
136 \global\Etoc@iiifalse
137 \global\Etoc@iifalse
138 \global\Etoc@ifalse
139 \global\Etoc@true
140 \or
141 \global\Etoc@vfalse
142 \global\Etoc@ivfalse
143 \global\Etoc@iiifalse
144 \global\Etoc@iifalse
145 \global\Etoc@itrue
146 \or
147 \global\Etoc@vfalse
148 \global\Etoc@ivfalse
149 \global\Etoc@iiifalse
150 \global\Etoc@iitrue
151 \or
152 \global\Etoc@vfalse
153 \global\Etoc@ivfalse
154 \global\Etoc@iiitrue
155 \or
156 \global\Etoc@vfalse
157 \global\Etoc@ivtrue
158 \or
159 \global\Etoc@vtrue
160 \else
```

13. Implementation

```
161 \ifnum#1=\m@ne
162 \global\Etoc@vfalse
163 \global\Etoc@ivfalse
164 \global\Etoc@iiifalse
165 \global\Etoc@iifalse
166 \global\Etoc@ifalse
167 \global\Etoc@false
168 \global\Etoc@jtrue
169 \else
170 \global\Etoc@vfalse
171 \global\Etoc@ivfalse
172 \global\Etoc@iiifalse
173 \global\Etoc@iifalse
174 \global\Etoc@ifalse
175 \global\Etoc@false
176 \global\Etoc@jfalse
177 \global\Etoc@jjtrue
178 \fi
179 \fi}

placeholder for comments
180 \def\Etoc@etoccontentsline#1{%
181 \global\expandafter\let\expandafter\Etoc@tmp\csname Etoc@#1@@\endcsname
182 \global\Etoc@partfalse
183 \Etoc@skipthisonefalse
184 \ifEtoc@hyperref
185 \def\Etoc@next{\expandafter\@gobbletwo\@gobblefour}%
186 \else
187 \def\Etoc@next{\expandafter\@gobble\@gobblefour}%
188 \fi
189 \ifEtoc@localtoc
190 \let\Etoc@prenext\relax
191 \ifEtoc@stoptoc
192 \Etoc@skipthisonettrue
193 \fi
194 \ifnum\Etoc@tmp<\Etoc@localtop
195 \def\Etoc@prenext{\global\Etoc@stoptoctrue}%
196 \Etoc@skipthisonettrue
197 \fi
198 \ifEtoc@notactive
199 \def\Etoc@prenext{\Etoc@setflags{\Etoc@tmp}}%
200 \Etoc@skipthisonettrue
201 \fi
202 \Etoc@prenext
203 \fi
204 \ifnum\c@tocdepth<\Etoc@tmp\relax\else
205 \ifEtoc@skipthisone\else
206 \global\let\Etoc@next\relax
207 \ifcase\Etoc@tmp
208 \ifEtoc@v \Etoc@end@five\fi
209 \ifEtoc@iv \Etoc@end@four\fi
210 \ifEtoc@iii \Etoc@end@three\fi
211 \ifEtoc@ii \Etoc@end@two\fi
```

13. Implementation

```
212 \ifEtoc@i \Etoc@end@one\fi
213 \ifEtoc@ \else \def\Etoc@next{\Etoc@begin@zero}\fi
214 \gdef\Etoc@contents{\Etoc@contents@zero}%
215 \gdef\Etoc@prefix{\Etoc@prefix@zero}%
216 \or
217 \ifEtoc@v \Etoc@end@five\fi
218 \ifEtoc@iv \Etoc@end@four\fi
219 \ifEtoc@iii \Etoc@end@three\fi
220 \ifEtoc@ii \Etoc@end@two\fi
221 \ifEtoc@i \else \def\Etoc@next{\Etoc@begin@one}\fi
222 \gdef\Etoc@contents{\Etoc@contents@one}%
223 \gdef\Etoc@prefix{\Etoc@prefix@one}%
224 \or
225 \ifEtoc@v \Etoc@end@five\fi
226 \ifEtoc@iv \Etoc@end@four\fi
227 \ifEtoc@iii \Etoc@end@three\fi
228 \ifEtoc@ii \else \def\Etoc@next{\Etoc@begin@two}\fi
229 \gdef\Etoc@contents{\Etoc@contents@two}%
230 \gdef\Etoc@prefix{\Etoc@prefix@two}%
231 \or
232 \ifEtoc@v \Etoc@end@five\fi
233 \ifEtoc@iv \Etoc@end@four\fi
234 \ifEtoc@iii \else \def\Etoc@next{\Etoc@begin@three}\fi
235 \gdef\Etoc@contents{\Etoc@contents@three}%
236 \gdef\Etoc@prefix{\Etoc@prefix@three}%
237 \or
238 \ifEtoc@v \Etoc@end@five\fi
239 \ifEtoc@iv \else \def\Etoc@next{\Etoc@begin@four}\fi
240 \gdef\Etoc@contents{\Etoc@contents@four}%
241 \gdef\Etoc@prefix{\Etoc@prefix@four}%
242 \or
243 \ifEtoc@v \else \def\Etoc@next{\Etoc@begin@five}\fi
244 \gdef\Etoc@contents{\Etoc@contents@five}%
245 \gdef\Etoc@prefix{\Etoc@prefix@five}%
246 \else
247 \ifnum\Etoc@tmp=\m@ne
248 \ifEtoc@v \Etoc@end@five\fi
249 \ifEtoc@iv \Etoc@end@four\fi
250 \ifEtoc@iii \Etoc@end@three\fi
251 \ifEtoc@ii \Etoc@end@two\fi
252 \ifEtoc@i \Etoc@end@one\fi
253 \ifEtoc@ \Etoc@end@zero\fi
254 \ifEtoc@j \else \def\Etoc@next{\Etoc@begin@minusone}\fi
255 \global\Etoc@parttrue
256 \gdef\Etoc@contents{\Etoc@contents@minusone}%
257 \gdef\Etoc@prefix{\Etoc@prefix@minusone}%
258 \else
259 \ifEtoc@v \Etoc@end@five\fi
260 \ifEtoc@iv \Etoc@end@four\fi
261 \ifEtoc@iii \Etoc@end@three\fi
262 \ifEtoc@ii \Etoc@end@two\fi
263 \ifEtoc@i \Etoc@end@one\fi
```

13. Implementation

```

264 \ifEtoc@ \Etoc@end@zero\fi
265 \ifEtoc@j \Etoc@end@minusone\fi
266 \ifEtoc@jj \else \def\Etoc@next{\Etoc@begin@minustwo}\fi
267 \gdef\Etoc@contents{\Etoc@contents@minustwo}%
268 \gdef\Etoc@prefix{\Etoc@prefix@minustwo}%
269 \fi
270 \fi
271 \Etoc@setflags{\Etoc@tmp}%
272 \fi\fi
273 \Etoc@next
274 \@firstoftwo{\Etoc@swa{#1}}{\Etoc@swb{#1}}

placeholder for comments
275 \def\Etoc@lxyz #1#2{\global\@namedef{etocname }{\leavevmode #1\hspace}%
276 \global\@namedef{etocpage }{\leavevmode #2\hspace}%
277 \Etoc@getnb #1\relax\relax\etoc@
278 \ifEtoc@number\else\ifEtoc@part
279 \Etoc@getit #1\hspace\relax\etoc@\fi\fi}

placeholder for comments
280 \def\Etoc@getnb #1{\let\Etoc@next\Etoc@getnb@nohyp
281 \ifEtoc@hyperref\ifx #1\hyper@linkstart
282 \let\Etoc@next\Etoc@getnb@hyp\fi\fi\Etoc@next #1}
283 \def\Etoc@getit #1{\let\Etoc@next\Etoc@getit@nohyp
284 \ifEtoc@hyperref\ifx #1\hyper@linkstart
285 \let\Etoc@next\Etoc@getit@hyp\fi\fi\Etoc@next #1}

placeholder for comments
286 \def\Etoc@getnb@nohyp #1#2#3\etoc@{%
287 \ifx #1\numberline
288 \global\@namedef{etocnumber }{\leavevmode #2\hspace}\global\Etoc@numbertrue
289 \else
290 \global\@namedef{etocnumber }{\leavevmode\hspace}\global\Etoc@numberfalse
291 \fi}

placeholder for comments
292 \def\Etoc@getnb@hyp #1#2#3#4#5#6\etoc@{%
293 \def\Etoc@getnbr ##1##2##3\etoc@{%
294 \ifx ##1\numberline
295 \global\@namedef{etocnumber }{\leavevmode #1{#2}{#3}{##2}#5\hspace}%
296 \global\Etoc@numbertrue
297 \else
298 \global\@namedef{etocnumber }{\leavevmode\hspace}%
299 \global\Etoc@numberfalse
300 \fi}%
301 \Etoc@getnbr #4\relax\relax\etoc@}

placeholder for comments
302 \def\Etoc@getit@nohyp #1\hspace#2#3\etoc@{%
303 \def\Etoc@getname ##1\hspace\relax\etoc@{%
304 \global\@namedef{etocname }{\leavevmode ##1\hspace}}%
305 \ifx \relax#2\else
306 \global\@namedef{etocnumber }{\leavevmode #1\hspace}%
307 \global\Etoc@numbertrue
308 \Etoc@getname #3\etoc@\fi}

```

13. Implementation

placeholder for comments

```

309 \def\Etoc@getit@hyp #1#2#3#4#5#6\etoc@{%
310   \def\Etoc@getname ##1\hspace\relax\etoc@{%
311     \global\@namedef{etocname }{\leavevmode #1{#2}{#3}{##1}#5\hspace}}%
312   \def\Etoc@getnbr ##1\hspace##2##3\etoc@{%
313     \ifx\relax##2\else
314       \global\@namedef{etocnumber }{\leavevmode #1{#2}{#3}{##1}#5\hspace}%
315       \global\Etoc@numbertrue
316       \Etoc@getname ##3\etoc@\fi}%
317   \Etoc@getnbr #4\hspace\relax\etoc@}

```

placeholder for comments

```

318 \edef\etocpage{\noexpand\protect\expandafter\noexpand
319   \csname etocpage \endcsname}
320 \edef\etocname{\noexpand\protect\expandafter\noexpand
321   \csname etocname \endcsname}
322 \edef\etocnumber{\noexpand\protect\expandafter\noexpand
323   \csname etocnumber \endcsname}

```

placeholder for comments

```

324 \@namedef{etocifnumbered }{\ifEtoc@number
325 \expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}

```

placeholder for comments

```

326 \edef\etocifnumbered{\noexpand\protect\expandafter\noexpand
327   \csname etocifnumbered \endcsname}

```

placeholder for comments

```

328 \def\Etoc@readtoc#1{%
329   \ifeof #1
330     \let\Etoc@nextread\@gobble
331   \else
332     \let\Etoc@nextread\Etoc@readtoc
333     \read #1 to \Etoc@buffer
334     \global\Etoc@toctoks=\expandafter\expandafter\expandafter
335       {\expandafter\the\expandafter\Etoc@toctoks\Etoc@buffer}%
336   \fi
337   \Etoc@nextread{#1}%
338 }

```

placeholder for comments

```

339 \def\Etoc@starttoc{%
340 \begingroup
341 \makeatletter
342 \@ifpackageloaded{hyperref}
343   {\ifx\hyper@last\@undefined
344     \ltx@ifpackageloaded {parskip}{\parskip \z@ }{}}%
345     \IfFileExists {\jobname .toc}{\Hy@WarningNoLine {old toc file
346 detected, not used; run LaTeX again (cheers from etoc)}}{}}%
347     \expandafter\@gobble\fi}}%
348 {\IfFileExists{\jobname .toc}
349   {\endlinechar=-1\relax
350     \newread\Etoc@tf
351     \openin\Etoc@tf\@filef@und

```

13. Implementation

```
352         \Etoc@readtoc\Etoc@tf
353         \closein\Etoc@tf}
354     {\typeout{No file \jobname .toc.}}}%
355 \if@filesw \newwrite \tf@toc \immediate
356 \openout \tf@toc \jobname .toc\relax \fi
357 \@nobreakfalse \endgroup}

placeholder for comments
358 \AtBeginDocument{\Etoc@starttoc}
359 \def\Etoc@toctoc{%
360     \global\Etoc@vfalse
361     \global\Etoc@ivfalse
362     \global\Etoc@iiifalse
363     \global\Etoc@iifalse
364     \global\Etoc@ifalse
365     \global\Etoc@false
366     \global\Etoc@jfalse
367     \global\Etoc@jjfalse
368 \ifEtoc@standard
369 \Etoc@Lazarus
370 \etoc@setstyle{@minustwo}{}{}{}{}%
371 \etoc@setstyle{@minusone}{}{}{}{}%
372 \etoc@setstyle{@zero}{}{}{}{}%
373 \etoc@setstyle{@one}{}{}{}{}%
374 \etoc@setstyle{@two}{}{}{}{}%
375 \etoc@setstyle{@three}{}{}{}{}%
376 \etoc@setstyle{@four}{}{}{}{}%
377 \etoc@setstyle{@five}{}{}{}{}%
378 \else
379 \let\numberline\@gobble
380 \let\booknumberline\numberline
381 \let\partnumberline\numberline
382 \let\chapternumberline\numberline
383 \fi
384 \the\Etoc@toctoks
385 \ifEtoc@notactive\else
386     \ifEtoc@v \Etoc@end@five\fi
387     \ifEtoc@iv \Etoc@end@four\fi
388     \ifEtoc@iii \Etoc@end@three\fi
389     \ifEtoc@ii \Etoc@end@two\fi
390     \ifEtoc@i \Etoc@end@one\fi
391     \ifEtoc@ \Etoc@end@zero\fi
392     \ifEtoc@j \Etoc@end@minusone\fi
393     \ifEtoc@jj \Etoc@end@minustwo\fi
394 \fi}

placeholder for comments
395 \def\etoc@@startlocaltoc#1#2{%
396 \let\Etoc@next\relax
397     \ifEtoc@localtoc
398     \ifEtoc@notactive
399         \ifnum #1=#2\relax
400             \ifEtoc@jj \global\let\Etoc@localtop\Etoc@@minusone@@ \fi
```

13. Implementation

```

401      \ifEtoc@j    \global\let\Etoc@localtop\Etoc@@zero@@ \fi
402      \ifEtoc@     \global\let\Etoc@localtop\Etoc@@one@@ \fi
403      \ifEtoc@i    \global\let\Etoc@localtop\Etoc@@two@@ \fi
404      \ifEtoc@ii   \global\let\Etoc@localtop\Etoc@@three@@ \fi
405      \ifEtoc@iii  \global\let\Etoc@localtop\Etoc@@four@@ \fi
406      \ifEtoc@iv   \global\let\Etoc@localtop\Etoc@@five@@ \fi
407      \ifEtoc@v    \global\let\Etoc@localtop\Etoc@@six@@ \fi
408      \def\Etoc@next{\global\Etoc@notactivefalse
409                      \global\Etoc@vfalse
410                      \global\Etoc@ivfalse
411                      \global\Etoc@iiifalse
412                      \global\Etoc@iifalse
413                      \global\Etoc@ifalse
414                      \global\Etoc@false
415                      \global\Etoc@jfalse
416                      \global\Etoc@jjfalse}%
417      \fi
418      \fi\fi
419 \Etoc@next}
420 \def\etoc@startlocaltoc#1{\etoc@@startlocaltoc{#1}{\c@etoc@tocid}}
placeholder for comments
421 \long\def\Etoc@localtableofcontents#1{%
422     \edef\Etoc@tmp{#1}%
423     \ifnum\Etoc@tmp<\@ne
424     \PackageWarning{etoc}
425         {Unknown toc id: run LaTeX to get references right}%
426     \leavevmode --unknown etoc ref: run latex again--\par
427     \let\Etoc@next\@gobble\else\let\Etoc@next\@firstofone\fi
428     \Etoc@next
429     {\begingroup
430     \edef\etoc@startlocaltoc##1{%
431         \noexpand\etoc@@startlocaltoc{##1}{#1}}
432     \Etoc@localtoctrue
433     \let\Etoc@localtop\Etoc@@minustwo@@
434     \global\Etoc@stoptocfalse
435     \global\Etoc@notactivetrue
436     \Etoc@tableofcontents
437     \endgroup}%
438     \ifEtoc@mustclosegroup\expandafter\endgroup\fi}
placeholder for comments
439 \def\Etoc@getrefno #1#2\etoc@{#1}
440 \def\Etoc@getref #1{\@ifundefined{r@#1}{0}{\expandafter\expandafter\expandafter
441     \Etoc@getrefno\csname r@#1\endcsname\relax\etoc@}}
442 \def\Etoc@ref#1{\Etoc@localtableofcontents{\Etoc@getref{#1}}}
443 \def\Etoc@label#1{\label{#1}}%
444     \ifEtoc@tocwithid\else
445     \addtocontents{toc}
446         {\string\etoc@startlocaltoc\string{\arabic{etoc@tocid}\string}}}%
447     \fi
448     \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
449 {\def\1{\Etoc@again}\expandafter

```

13. Implementation

```

450 \gdef\1 {\futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}}
placeholder for comments
451 \def\Etoc@t@bleofcontents{%
452 \ifx\Etoc@nexttoken\label
453 \def\Etoc@next{\expandafter\Etoc@label\@gobble}\else
454 \ifx\Etoc@nexttoken\@sptoken
455 \let\Etoc@next\Etoc@again\else
456 \ifEtoc@tocwithid
457 \def\Etoc@next{\Etoc@localtableofcontents{\c@etoc@tocid}}%
458 \else
459 \ifx\Etoc@nexttoken\ref
460 \def\Etoc@next{\expandafter\Etoc@ref\@gobble}%
461 \else
462 \def\Etoc@next{\Etoc@localtocfalse
463 \global\Etoc@notactivefalse
464 \begingroup\Etoc@tableofcontents
465 \ifEtoc@mustclosegroup\expandafter\endgroup\fi
466 \endgroup}%
467 \fi
468 \fi\fi\fi\Etoc@next}
placeholder for comments
469 \newcommand\etocaftertitlehook{}
470 \def\table@fcontents{\par
471 \refstepcounter{etoc@tocid}%
472 \Etoc@tocwithidfalse
473 \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
474 \def\localtable@fcontents{\par
475 \refstepcounter{etoc@tocid}%
476 \addtocontents{toc}
477 {\string\etoc@startlocaltoc\string{\arabic{etoc@tocid}\string}}
478 \Etoc@tocwithidtrue
479 \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
480 \def\tableofcontents{\@ifstar
481 {\let\Etoc@aftertitlehook\relax\table@fcontents}
482 {\let\Etoc@aftertitlehook\etocaftertitlehook\table@fcontents}}
483 \def\localtableofcontents{\@ifstar
484 {\let\Etoc@aftertitlehook\relax\localtable@fcontents}
485 {\let\Etoc@aftertitlehook\etocaftertitlehook\localtable@fcontents}}
placeholder for comments
486 \newcommand\etocsettocstyle[2]{%
487 \long\def\Etoc@tableofcontents
488 {\ifnum\c@tocdepth>\Etoc@minf\let\Etoc@@next\@firstofone\else
489 \let\Etoc@@next\@gobble\fi
490 \Etoc@@next{#1\Etoc@aftertitlehook
491 \Etoc@toctoc\let\Etoc@n@xt\relax
492 \ifEtoc@tocwithid\else
493 \ifEtoc@localtoc
494 \ifEtoc@notactive
495 \def\Etoc@n@xt{\Etoc@localtocfalse
496 \global\Etoc@notactivefalse
497 \Etoc@toctoc}%

```


13. Implementation

```

498          \fi\fi\fi\Etoc@n@xt
499          #2}}}}
500 \newcommand\etocextendtocstyle[2]{%
501 \toks0=\expandafter{\Etoc@tableofcontents #2}%
502 \toks2={#1}%
503 \long\edef\Etoc@tableofcontents{\the\toks2 \the\toks0}}
504 %          % je me fais avoir ^ à chaque fois!

placeholder for comments

505 \newcommand\etocsetstyle{\Etoc@standardfalse\etoc@setstyle}
506 \long\def\etoc@setstyle#1#2#3#4#5{%
507 \long\expandafter\def
508     \csname Etoc@begin@\csname Etoc@#1@\endcsname\endcsname {#2}%
509 \long\expandafter\def
510     \csname Etoc@prefix@\csname Etoc@#1@\endcsname\endcsname {#3}%
511 \long\expandafter\def
512     \csname Etoc@contents@\csname Etoc@#1@\endcsname\endcsname {#4}%
513 \long\expandafter\def
514     \csname Etoc@end@\csname Etoc@#1@\endcsname\endcsname {#5}}

placeholder for comments

515 \newcommand\etocfontminustwo{\normalfont \LARGE \bfseries}
516 \newcommand\etocfontminusone{\normalfont \large \bfseries}
517 \newcommand\etocfontzero{\normalfont \large \bfseries}
518 \newcommand\etocfontone{\normalfont \normalsize \bfseries}
519 \newcommand\etocfonttwo{\normalfont \normalsize}
520 \newcommand\etocfontthree{\normalfont \footnotesize}
521 \newcommand\etocfontfour{\normalfont \footnotesize}
522 \newcommand\etocfontfive{\normalfont \footnotesize}

placeholder for comments

523 \newcommand\etocsepminustwo{4ex \@plus .5ex \@minus .5ex}
524 \newcommand\etocsepminusone{4ex \@plus .5ex \@minus .5ex}
525 \newcommand\etocsepzero{2.5ex \@plus .4ex \@minus .4ex}
526 \newcommand\etocsepone{1.5ex \@plus .3ex \@minus .3ex}
527 \newcommand\etocseptwo{1ex \@plus .15ex \@minus .15ex}
528 \newcommand\etocsepthree{.25ex \@plus .05ex \@minus .05ex}

placeholder for comments

529 \newcommand\etocbaselinespreadminustwo{1}
530 \newcommand\etocbaselinespreadminusone{1}
531 \newcommand\etocbaselinespreadzero{1}
532 \newcommand\etocbaselinespreadone{1}
533 \newcommand\etocbaselinespreadtwo{1}
534 \newcommand\etocbaselinespreadthree{.9}

placeholder for comments

535 \newcommand\etocminustwoleftmargin{1.5em plus 0.5fil}
536 \newcommand\etocminustworightmargin{1.5em plus -0.5fil}
537 \newcommand\etocminusoneleftmargin{1em}
538 \newcommand\etocminusonerightmargin{1em}
539 \newcommand\etocclineleaders
540     {\hbox{\normalfont\normalsize\hb@xt@2ex {\hss.\hss}}}
541 \newcommand\etocabbrevpagename{p.~}
542 \newcommand\etocpartname{\partname}

```

13. Implementation

```
543 \newcommand\etocbookname{Book}
placeholder for comments
544 \def\etococlines{\Etoc@standardfalse
placeholder for comments
545 \etoc@setstyle{@minustwo}{\etocskipfirstprefix}{\pagebreak[3]}
546 {\beginingroup \etocfontminustwo
547 \addvspace{\etocsepminustwo}%
548 \parindent \z@
549 \leftskip \etocminustwoleftmargin
550 \rightskip \etocminustworightmargin
551 \parfillskip \@flushglue
552 \vbox{\etocifnumbered{\etocbookname~\etocnumber:~}{ }\etocname
553 \baselineskip\etocbaselinespreadminustwo\baselineskip\par}%
554 \addvspace{\etocsepminusone}\nobreak
555 \endgroup}
556 {}%
557 \etoc@setstyle{@minusone}{\etocskipfirstprefix}{\pagebreak[3]}
558 {\beginingroup \etocfontminusone
559 \addvspace{\etocsepminusone}%
560 \parindent \z@
561 \leftskip \etocminusoneleftmargin
562 \rightskip \etocminusonerightmargin
563 \parfillskip \@flushglue
564 \vbox{\etocpartname\etocifnumbered{ \etocnumber.~}{ } \etocname
565 \baselineskip\etocbaselinespreadminusone\baselineskip\par}%
566 \addvspace{\etocsepzero}\nobreak
567 \endgroup}
568 {}%
569 \etoc@setstyle{@zero}
570 {\addpenalty\@highpenalty\etocskipfirstprefix}{\pagebreak[2]}
571 {\beginingroup
572 \etocfontzero
573 \addvspace{\etocsepzero}%
574 \parindent \z@ \parfillskip \@flushglue
575 \vbox{\etocifnumbered{\etocnumber\space}{ }\etocname
576 \baselineskip\etocbaselinespreadzero\baselineskip\par}%
577 \addvspace{\etocsepone}\addpenalty\@itempenalty
578 \endgroup}
579 {\addpenalty\@secpenalty}%
placeholder for comments
580 \etoc@setstyle{@one}
581 {\addpenalty\@medpenalty\etocskipfirstprefix}{\pagebreak[1]}
582 {\beginingroup\etocfontone
583 \addvspace{\etocsepone}%
584 \parindent \z@ \parfillskip \z@
585 \setbox\z@\vbox{\parfillskip \@flushglue
586 \etocname\par\setbox\tw@\lastbox
587 \global\setbox\@ne\hbox{\unhbox\tw@}}}%
588 \dimen\z@=\wd\@ne
589 \setbox\z@=\etococlineleaders
590 \advance\dimen\z@\wd\z@
```

13. Implementation

```
591 \etocifnumbered
592 {\setbox\tw@\hbox{\etocnumber, \etocabbrevpagename\etocpage}}
593 {\setbox\tw@\hbox{\etocabbrevpagename\etocpage}}}%
594 \advance\dimen\z@\wd\tw@
595 \ifdim\dimen\z@ < \linewidth
596     \vbox{\etocname\nobreak
597         \leaders\box\z@\hfil\box\tw@
598         \baselineskip\etocbaselinespreadone\baselineskip\par}
599 \else
600     \vbox{\etocname\nobreak
601         \ifdim\wd\@ne<\linewidth\leaders\copy\z@\hfil\break\fi
602         \hbox{}}\leaders\box\z@\hfil\box\tw@
603         \baselineskip\etocbaselinespreadone\baselineskip\par}
604 \fi
605 \addvspace{\etocsepttwo}\addpenalty\@itempenalty
606 \endgroup}
607 {\addpenalty\@secpenalty}%

placeholder for comments

608 \etoc@setstyle{@two}
609 {\addpenalty\@medpenalty\etocskipfirstprefix}
610 {\pagebreak[1]}
611 {\begingroup\etocfonttwo
612 \addvspace{\etocsepttwo}%
613 \parindent \z@ \parfillskip \z@
614 \setbox\z@\vbox{\parfillskip\@flushglue
615 \etocname\par\setbox\tw@\lastbox
616 \global\setbox\@ne\hbox{\unhbox\tw@}}}%
617 \dimen\z@=\wd\@ne
618 \setbox\z@=\etoclineleaders
619 \advance\dimen\z@\wd\z@
620 \etocifnumbered
621 {\setbox\tw@\hbox{\etocnumber, \etocabbrevpagename\etocpage}}
622 {\setbox\tw@\hbox{\etocabbrevpagename\etocpage}}}%
623 \advance\dimen\z@\wd\tw@
624 \ifdim\dimen\z@ < \linewidth
625     \vbox{\etocname\nobreak
626         \leaders\box\z@\hfil\box\tw@
627         \baselineskip\etocbaselinespreadtwo\baselineskip\par}
628 \else
629     \vbox{\etocname\nobreak
630         \ifdim\wd\@ne<\linewidth\leaders\copy\z@\hfil\break\fi
631         \hbox{}}\leaders\box\z@\hfil\box\tw@
632         \baselineskip\etocbaselinespreadtwo\baselineskip\par}
633 \fi
634 \addvspace{\etocseptthree}\addpenalty\@itempenalty
635 \endgroup}
636 {\addpenalty\@secpenalty}%

placeholder for comments

637 \etoc@setstyle{@three}
638 {\nopagebreak
639 \etocfontthree
```

13. Implementation

```

640 \addvspace{\etocsepthree}%
641 \nointerlineskip\noindent
642 \etocskipfirstprefix}
643 {\allowbreak\,--\,}
644 {\etocname\unskip} % \xspace in \etocname
645 {\. \hfil\begin{group}
646 \baselineskip\etocbaselinespreadthree\baselineskip
647 \par\end{group}
648 \addpenalty{-\@highpenalty}}%

placeholder for comments
649 \etoc@setstyle{@four}{}{}{}{}%

placeholder for comments
650 \etoc@setstyle{@five}{}{}{}{}%
651 }

placeholder for comments
652 \newcommand\etocabovetocskip{3.5ex \@plus 1ex \@minus .2ex}
653 \newcommand\etocbelowtocskip{3.5ex \@plus 1ex \@minus .2ex}
654 \newcommand\etoccolumnsep{2em}
655 \newcommand\etocmulticolsep{0ex}
656 \newcommand\etocmulticolpretolerance{-1}
657 \newcommand\etocmulticoltolerance{200}
658 \newcommand\etocdefaultnbcol{2}

placeholder for comments
659 \newcommand\etocmulticolstyle[2][\etocdefaultnbcol]{%
660 \etocsettocstyle
661   {\addvspace{\etocabovetocskip}}%
662   \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
663     \else \let\Etoc@next\@secondoftwo\fi
664   \Etoc@next{%
665     \multicolpretolerance\etocmulticolpretolerance
666     \multicoltolerance\etocmulticoltolerance
667     \setlength{\columnsep}{\etoccolumnsep}%
668     \setlength{\multicolsep}{\etocmulticolsep}%
669     \begin{multicols}{#1}[#2\addvspace{\etocsepminusone}]}
670     {#2\addvspace{\etocsepminusone}}%
671     \pretolerance\etocmulticolpretolerance
672     \tolerance\etocmulticoltolerance}}
673 {\ifnum #1>\@ne\let\Etoc@next\@firstofone
674   \else \let\Etoc@next\@gobble\fi
675   \Etoc@next{\end{multicols}}}%
676   \addvspace{\etocbelowtocskip}}

placeholder for comments
677 \newcommand\etocinnertopsep{2ex}
678 \newcommand\etocinnerbottomsep{3.5ex}
679 \newcommand\etocinnerleftsep{2em}
680 \newcommand\etocinnerrightsep{2em}
681 \newcommand\etoctoprule{\hrule}
682 \newcommand\etocleftrule{\vrule}
683 \newcommand\etocrightrule{\vrule}
684 \newcommand\etocbottomrule{\hrule}

```

13. Implementation

```

685 \newcommand\etocprulecolorcmd{\relax}
686 \newcommand\etocbottomrulecolorcmd{\relax}
687 \newcommand\etocleftrulecolorcmd{\relax}
688 \newcommand\etocrightrulecolorcmd{\relax}

placeholder for comments
689 \newcommand\etoc@ruledheading[1]{%
690   \hb@xt@\linewidth{\color@begingroup
691     \hss #1\hss\hskip-\linewidth
692     \etocprulecolorcmd\leaders\etocprule\hss
693     \phantom{#1}%
694     \leaders\etocprule\hss\color@endgroup}%
695     \nointerlineskip\vskip\etocinnertopsep}

placeholder for comments
696 \newcommand\etocruledstyle[2][\etocdefaultnbcoll]{%
697 \etocsettocstyle
698   {\addvspace{\etocabovetocskip}}%
699   \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
700     \else \let\Etoc@next\@secondoftwo\fi
701   \Etoc@next
702     {\multicolpretolerance\etocmulticolpretolerance
703     \multicoltolerance\etocmulticoltolerance
704     \setlength{\columnsep}{\etoccolumnsep}%
705     \setlength{\multicolsep}{\etocmulticolsep}%
706     \begin{multicols}{#1}[\etoc@ruledheading{#2}]}
707     {\etoc@ruledheading{#2}\nobreak
708     \pretolerance\etocmulticolpretolerance
709     \tolerance\etocmulticoltolerance}}
710   {\ifnum #1>\@ne\let\Etoc@next\@firstofone
711     \else \let\Etoc@next\@gobble\fi
712     \Etoc@next{\end{multicols}}}%
713   \addvspace{\etocbelowtocskip}}

placeholder for comments
714 \newcommand\etocframedmphook{\relax}
715 \newcommand\etocbkgcolorcmd{\relax}
716 \newcommand\Etoc@relax{\relax}
717 \newbox\etoc@framed@titlebox
718 \newbox\etoc@framed@contentsbox
719 \newcommand\etocframedstyle[2][\etocdefaultnbcoll]{%
720 \etocsettocstyle{%
721   \addvspace{\etocabovetocskip}%
722   \sbox\z@{#2}%
723   \dimen\z@=\dp\z@
724   \ifdim\wd\z@<\linewidth \dp\z@=\z@ \else \dimen\z@=\z@ \fi
725   \setbox\etoc@framed@titlebox=\hb@xt@\linewidth{\color@begingroup
726     \hss
727     \ifx\etocbkgcolorcmd\Etoc@relax\else
728       \sbox\tw@{\color{white}}%
729       \vrule\@width\wd\z@\@height\ht\z@\@depth\dimen\z@}%
730     \ifdim\wd\z@<\linewidth \dp\tw@=\z@\fi
731     \box\tw@
732     \hskip-\wd\z@

```

13. Implementation

```

733     \fi
734     \copy\z@
735     \hss
736     \hskip-\linewidth
737     \etocoprulcolorcmd\leaders\etocoprul\hss%
738     \hskip\wd\z@
739     \etocoprulcolorcmd\leaders\etocoprul\hss\color@endgroup}%
740 \setbox\z@\hbox{\etocleftrule\etocrightrule}%
741 \dimen\tw@\linewidth\advance\dimen\tw@-\wd\z@
742     \advance\dimen\tw@-\etocinnerleftsep
743     \advance\dimen\tw@-\etocinnerrightsep
744 \setbox\etoc@framed@contentsbox=\vbox\bgroup
745     \hsize\dimen\tw@
746     \kern\dimen\z@
747     \vskip\etocinnertopsep
748     \hbox\bgroup
749     \begin{minipage}{\hsize}%
750     \etocframedmphook
751 \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
752 \else \let\Etoc@next\@secondoftwo\fi
753     \Etoc@next
754     {\multicolpretolerance\etocmulticolpretolerance
755     \multicoltolerance\etocmulticoltolerance
756     \setlength{\columnsep}{\etoccolumnsep}%
757     \setlength{\multicolsep}{\etocmulticolsep}%
758     \begin{multicols}{#1}}
759     {\pretolerance\etocmulticolpretolerance
760     \tolerance\etocmulticoltolerance}}
761 {\ifnum #1>\@ne\let\Etoc@next\@firstofone
762     \else \let\Etoc@next\@gobble\fi
763 \Etoc@next\end{multicols}\unskip}%
764 \end{minipage}%
765 \egroup
766 \vskip\etocinnerbottomsep
767 \egroup
768 \vbox{\hsize\linewidth
769     \ifx\etocbkgcolorcmd\Etoc@relax\else
770     \kern\ht\etoc@framed@titlebox
771     \kern\dp\etoc@framed@titlebox
772     \hb@xt@\linewidth{\color@begingroup
773     \etocleftrulecolorcmd\etocleftrule
774     \etocbkgcolorcmd
775     \leaders\vrule
776         \@height\ht\etoc@framed@contentsbox
777         \@depth\dp\etoc@framed@contentsbox
778     \hss
779     \etocrightrulecolorcmd\etocrightrule
780     \color@endgroup}\nointerlineskip
781     \vskip-\dp\etoc@framed@contentsbox
782     \vskip-\ht\etoc@framed@contentsbox
783     \vskip-\dp\etoc@framed@titlebox
784     \vskip-\ht\etoc@framed@titlebox

```

13. Implementation

```
785 \fi
786 \box\etoc@framed@titlebox\nointerlineskip
787 \hb@xt@\linewidth{\color@begingroup
788 {\etocleftrulecolorcmd\etocleftrule}%
789 \hss\box\etoc@framed@contentsbox\hss
790 \etocrightrulecolorcmd\etocrightrule\color@endgroup}
791 \nointerlineskip
792 \vskip\ht\etoc@framed@contentsbox
793 \vskip\dp\etoc@framed@contentsbox
794 \hb@xt@\linewidth{\color@begingroup\etocbottomrulecolorcmd
795 \leaders\etocbottomrule\hss\color@endgroup}}
796 \addvspace{\etocbelowtocskip}}}

placeholder for comments
797 \newcommand\etoc@multicoltoc[2][\etocdefaultnbcol]{%
798 \etocmulticolstyle[#1]{#2}%
799 \tableofcontents}
800 \newcommand\etoc@multicoltoi[2][\etocdefaultnbcol]{%
801 \etocmulticolstyle[#1]{#2}%
802 \tableofcontents*}
803 \newcommand\etoc@local@multicoltoc[2][\etocdefaultnbcol]{%
804 \etocmulticolstyle[#1]{#2}%
805 \localtableofcontents}
806 \newcommand\etoc@local@multicoltoi[2][\etocdefaultnbcol]{%
807 \etocmulticolstyle[#1]{#2}%
808 \localtableofcontents*}

placeholder for comments
809 \newcommand\etoc@ruledtoc[2][\etocdefaultnbcol]{%
810 \etocruledstyle[#1]{#2}%
811 \tableofcontents}
812 \newcommand\etoc@ruledtoi[2][\etocdefaultnbcol]{%
813 \etocruledstyle[#1]{#2}%
814 \tableofcontents*}
815 \newcommand\etoc@local@ruledtoc[2][\etocdefaultnbcol]{%
816 \etocruledstyle[#1]{#2}%
817 \localtableofcontents}
818 \newcommand\etoc@local@ruledtoi[2][\etocdefaultnbcol]{%
819 \etocruledstyle[#1]{#2}%
820 \localtableofcontents*}

placeholder for comments
821 \newcommand\etoc@framedtoc[2][\etocdefaultnbcol]{%
822 \etocframedstyle[#1]{#2}%
823 \tableofcontents}
824 \newcommand\etoc@framedtoi[2][\etocdefaultnbcol]{%
825 \etocframedstyle[#1]{#2}%
826 \tableofcontents*}
827 \newcommand\etoc@local@framedtoc[2][\etocdefaultnbcol]{%
828 \etocframedstyle[#1]{#2}%
829 \localtableofcontents}
830 \newcommand\etoc@local@framedtoi[2][\etocdefaultnbcol]{%
831 \etocframedstyle[#1]{#2}%
832 \localtableofcontents*}
```

13. Implementation

placeholder for comments

```
833 \newcommand\etocmulticol{\begingroup
834   \Etoc@mustclosegrouptrue
835   \@ifstar
836   {\etoc@multicoltoctoci}
837   {\etoc@multicoltoctoc}}
838 \newcommand\etocruled{\begingroup
839   \Etoc@mustclosegrouptrue
840   \@ifstar
841   {\etoc@ruledtoctoci}
842   {\etoc@ruledtoctoc}}
843 \newcommand\etocframed{\begingroup
844   \Etoc@mustclosegrouptrue
845   \@ifstar
846   {\etoc@framedtoctoci}
847   {\etoc@framedtoctoc}}
848 \newcommand\etoclocalmulticol{\begingroup
849   \Etoc@mustclosegrouptrue
850   \@ifstar
851   {\etoc@local@multicoltoctoci}
852   {\etoc@local@multicoltoctoc}}
853 \newcommand\etoclocalruled{\begingroup
854   \Etoc@mustclosegrouptrue
855   \@ifstar
856   {\etoc@local@ruledtoctoci}
857   {\etoc@local@ruledtoctoc}}
858 \newcommand\etoclocalframed{\begingroup
859   \Etoc@mustclosegrouptrue
860   \@ifstar
861   {\etoc@local@framedtoctoci}
862   {\etoc@local@framedtoctoc}}
```

placeholder for comments

```
863 \newcommand\etocarticlestyle{%
864   \etocsettocstyle
865   {\section *{\contentsname
866     \@mkboth {\MakeUppercase \contentsname}
867     {\MakeUppercase \contentsname}}}
868   {}}
869 \newcommand\etocarticlestylenomarks{%
870   \etocsettocstyle
871   {\section *{\contentsname}}
872   {}}
```

placeholder for comments

```
873 \newcommand\etocbookstyle{%
874   \etocsettocstyle
875   {\if@twocolumn \@restonecoltrue \onecolumn \else \@restonecolfalse \fi
876     \chapter *{\contentsname
877       \@mkboth {\MakeUppercase \contentsname}
878       {\MakeUppercase \contentsname}}}
879   {\if@restonecol \twocolumn \fi}}
880 \newcommand\etocbookstylenomarks{%
```


13. Implementation

```

881 \etocsettocstyle
882 {\if@twocolumn \@restonecoltrue \onecolumn \else \@restonecolfalse \fi
883 \chapter *{\contentsname}}
884 {\if@restonecol \twocolumn \fi}}
885 \let\etocreportstyle\etocbookstyle
886 \let\etocreportstylenomarks\etocbookstylenomarks

placeholder for comments. December 1, 11:33 Very weird problem with the two successive
\addtocontents (one for \etoc@startlocaltoc and the next one here for the memoir
class and unstarred \tableofcontents). They are not put in the correct order into the
aux file hence into the toc file. It is very weird that if the aux file already exists due to
a previous latex run where the user had put * after the TOC displaying command, the
things end up in the correct order in the new aux file, hence things work correctly. Anyway,
\etocaftertitlehook is defined here in a strange way which empirically seems to solve
the problem. v1.05a
887 \newcommand\Etoc@addcontentsline{}
888 \newcommand\etocmemoirtoc\tocfmt[2]{%
889 \renewcommand\Etoc@addcontentsline{\addcontentsline {toc}{#1}{#2}}%
890 \renewcommand\etocaftertitlehook{%
891 \ifmem@em@starred@listof
892 \else\phantomsection\aftergroup\Etoc@addcontentsline\fi}}
893 \newcommand\etocmemoirstyle{%
894 \etocsettocstyle
895 {\ensureonecol \par \begingroup \@nameuse {@tocmaketitle}
896 \Etoc@aftertitlehook\let\Etoc@aftertitlehook\relax
897 \parskip \cftparskip \@nameuse {cfttocbeforelisthook}}
898 {\@nameuse {cfttocafterlisthook}\endgroup\restorefromonecol}}

placeholder for comments
899 \newcommand\etocscartclstyle{%
900 \etocsettocstyle
901 {\let\if@dynlist\if@tocleft
902 \iftocfeature {toc}{onecolumn}
903 {\iftocfeature {toc}{leveldown}
904 {}
905 {\if@twocolumn \aftergroup \twocolumn \onecolumn \fi }}
906 {}}%
907 \tocbasic@listhead {\listoftocname}%
908 \begingroup \expandafter \expandafter \expandafter
909 \endgroup \expandafter
910 \ifx
911 \csname microtypesetup\endcsname \relax
912 \else
913 \iftocfeature {toc}{noptrusion}}
914 {\microtypesetup {protrusion=false}%
915 \PackageInfo {tocbasic}%
916 {character protrusion at toc deactivated}}}%
917 \fi
918 \setlength {\parskip }{\z@ }%
919 \setlength {\parindent }{\z@ }%
920 \setlength {\parfillskip }{\z@ \@plus 1fil}%
921 \csname tocbasic@@before@hook\endcsname
922 \csname tb@toc@before@hook\endcsname}

```

13. Implementation

```

923      {\csname tb@toc@after@hook\endcsname
924      \csname tocbasic@@after@hook\endcsname}}
925 \let\etocscrbookstyle\etocscrartclstyle
926 \let\etocscrreprtstyle\etocscrartclstyle

placeholder for comments
927 \newcommand\etocnakedtocdisplaystyle{\etocsettocstyle{}}{}
928 \newcommand\etocstandarddisplaystyle{\etocarticlestyle}
929 \newcommand\etocmarkboth[1]{%
930   \@mkboth{\MakeUppercase{#1}}{\MakeUppercase{#1}}}
931 \newcommand\etocmarkbothnouc[1]{\@mkboth{#1}{#1}}
932 \newcommand\etoc tocstyle[3][section]{\etocmulticolstyle[#2]%
933   {\csname #1\endcsname *{#3}}}
934 \newcommand\etoc tocstylewithmarks[4][section]{\etocmulticolstyle[#2]%
935   {\csname #1\endcsname *{#3\etocmarkboth{#4}}}}
936 \newcommand\etoc tocstylewithmarksnouc[4][section]{\etocmulticolstyle[#2]%
937   {\csname #1\endcsname *{#3\etocmarkbothnouc{#4}}}}

placeholder for comments
938 \def\Etoc@redefetocstyle#1{%
939   \renewcommand\etoc tocstylewithmarks[4][#1]
940   {\etocmulticolstyle[##2]%
941     {\csname ##1\endcsname *{##3\etocmarkboth{##4}}}}
942   \renewcommand\etoc tocstylewithmarksnouc[4][#1]
943   {\etocmulticolstyle[##2]%
944     {\csname ##1\endcsname *{##3\etocmarkbothnouc{##4}}}}
945   \renewcommand\etoc tocstyle[3][#1]{%
946     \etocmulticolstyle[##2]{\csname ##1\endcsname *{##3}}}
947 \@ifclassloaded{scrartcl}
948   {\renewcommand\etocstandarddisplaystyle{\etocscrartclstyle}}{}
949 \@ifclassloaded{book}
950   {\renewcommand\etocfontone{\normalfont\normalsize}
951     \renewcommand\etocstandarddisplaystyle{\etocbookstyle}
952     \Etoc@redefetocstyle{chapter}}{}
953 \@ifclassloaded{report}
954   {\renewcommand\etocfontone{\normalfont\normalsize}
955     \renewcommand\etocstandarddisplaystyle{\etocreportstyle}
956     \Etoc@redefetocstyle{chapter}}{}
957 \@ifclassloaded{scrbook}
958   {\renewcommand\etocfontone{\normalfont\normalsize}
959     \renewcommand\etocstandarddisplaystyle{\etocscrbookstyle}
960     \Etoc@redefetocstyle{chapter}}{}
961 \@ifclassloaded{scrreprt}
962   {\renewcommand\etocfontone{\normalfont\normalsize}
963     \renewcommand\etocstandarddisplaystyle{\etocscrreprtstyle}
964     \Etoc@redefetocstyle{chapter}}{}
965 \@ifclassloaded{memoir}
966   {\renewcommand\etocfontone{\normalfont\normalsize}
967     \etocmemoirtoctotocfmt{chapter}{\contentsname}%
968     \renewcommand\etocstandarddisplaystyle{\etocmemoirstyle}
969     \Etoc@redefetocstyle{chapter}}{}

placeholder for comments
970 \newcommand\etocstandardlines{\Etoc@standardtrue}

```

13. *Implementation*

```
971 \etoclines
972 \etocstandardlines
973 \etocstandarddisplaystyle
974 \endinput
```