

The **etoc** package

JEAN-FRANÇOIS BURNOL

jfbu (at) free (dot) fr

Package version: v1.06 (2012/12/07)

Documentation generated from the source file
with timestamp “07-12-2012 at 20:15:47 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.	
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> .	
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.	
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. 12
------------------------	----------

Part III. Package commands for global styles

Specifying the toc display style	7, p. 16
The command <code>\etocruledstyle</code>	7.1, p. 16
The command <code>\etocmulticolstyle</code>	7.2, p. 17
The command <code>\etocdisplaystyle</code> .	
A point of some importance	7.3, p. 17
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. 18
The starred variants of the <code>\tableofcontents</code> etc... commands	8, p. 18

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

Part IV. Using and customizing `etoc`

Summary of the main style commands	10, p. 21
Setting up local styles	10.1, p. 21
Setting up global styles	10.2, p. 21
Displaying Tables of Contents	10.3, p. 22
Labels and references	10.4, p. 22
An example	10.5, p. 22

Customizing <code>etoc</code>	11, p. 24	All subsections of this document	12.3, p. 28
Customizing the <code>etoc</code> line styles	11.1, p. 24	Compatibility with other packages	12.4, p. 28
Customizing the global styles	11.2, p. 25	T _E Xnical matters	12.5, p. 29
		Errors and catastrophes	12.6, p. 29
 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. 27	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 (L^AT_EX'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 T_EX 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`, the `\etocstandardlines` compatibility switch should be functional under the condition that the document class does things in a manner analogous to the standard classes. 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	21
10.1. Setting up local styles	21
10.2. Setting up global styles	21
10.3. Displaying Tables of Contents	22
10.4. Labels and references	22
10.5. An example	22
11. Customizing <code>etoc</code>	24
11.1. Customizing the <code>etoc</code> line styles	24
11.2. Customizing the global styles	25

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}`. Due to the way `etoc` implements the cloning,⁹ the doubly cloned TOC will be typeset as a full table of contents. 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.

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

⁹The double cloning had a different (but also wrong) effect in v1.05 compared to v1.06.

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}{-3}
\localtableofcontents \label{toc:partone}
```

The `tocdepth` being set to `-3`, nothing at all was inserted in the page (`etoc` cancels printing even the heading of the TOC if the `tocdepth` is `-2` or less, except for the `memoir` class for which it needs to be `-3` or less). We can then display the TOC with:

```
\setcounter{tocdepth}{3}
\subsection{Local table of contents for this part}
\tableofcontents \ref{toc:partone}
```

3.3. Local table of contents for this part

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 .toc 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 is to be found at its location.

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 12)

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:¹⁰

```
\setcounter{tocdepth}{3}
\etocsetstyle{section}
{\begin{enumerate}}
{\normalsize\bfseries\rmfamily\item}
```

¹⁰the 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

```
{\etocname (page \etocpage)}  
{\end{enumerate}}  
  
\etocsetstyle{subsection}  
{\begin{enumerate}}  
{\normalfont\item}  
{\etocname (p.\~\etocpage)}  
{\end{enumerate}}  
  
\etocsetstyle{subsubsection}  
{\par\nobreak\begin{group}\normalfont  
  \footnotesize\itshape\etocskipfirstprefix}  
{\allowbreak\,--\,}  
{\etocname\unskip}  
{.\hfil\par\end{group}\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.¹¹ 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:

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

```

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

¹¹ `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

4.1.3. `secnumdepth` and `tocdepth`

The \LaTeX counter `secnumdepth` decides down to which level division headings are numbered. Numbered or not they are (at least in the standard classes) written to the `.toc` file. The counter `tocdepth` decides then the finest level which will be displayed in the table of contents: its local value is obeyed by `etoc`, and as the `\tableofcontents` command and its variants may now be used multiple 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).¹²

The `\etocifnumbered{<A>}{}` command executes `<A>` if the number exists, and `` if not. So we use it in the code which was finally chosen for the section level:

```
\etocsetstyle{section}
{\begin{enumerate}[leftmargin=.75cm, label=\etocifnumbered
  {\fboxrule1pt\fcolorbox{green}{white}{\etocnumber}}}]
{\normalsize\bfseries\rmfamily\item}
{\etocname (page \etocpage)}
{\end{enumerate}}

\etocsetstyle{subsection}
{\begin{enumerate}[leftmargin=0cm, label=\etocnumber]}
{\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}
```

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

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:

¹²as seen below we use `\fcolorbox` rather than `\fbox`. Due to some redefinition made by package `xcolor`, had we used `\fbox` (and not used `hyperref`) we would have needed `\protect\fbox`.

5. Am I also in red?

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

This entire subsection 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 subsection 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 subsection 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 solution¹³ 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}
```

6. A comparison

The main table of contents was done in the **etoc** style. Let us issue here again two `\table-ofcontents` 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}
```

¹³one can insert by hand the language changes in each concerned sectioning command.

6. A comparison

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	II. Package commands for line styles	8
1. Initial motivation: nested lists	3	4. The <code>\etocsetstyle</code> command	8
1.1. Limitations intrinsic to the use of environments	3	4.1. An example	8
2. Line styles and global style	4	4.1.1. <code>\etocname</code> and <code>\etocpage</code>	9
2.1. <code>\etocsettocstyle</code> for the toc display	4	4.1.2. The <code>\etocskipfirstprefix</code> command	9
2.1.1. Multiple columns, ruled title, framed contents	4	4.1.3. <code>secnumdepth</code> and <code>tocdepth</code>	10
2.2. <code>\etocsetstyle</code> for the line styles	4	4.2. The <code>\etocsetlevel</code> command	10
2.2.1. Instructing <code>etoc</code> about the document sectional division names . . .	5	4.3. Further informations and let's make this a very long title to see how it will look like	10
2.3. Compatibility mode	5	4.3.1. The <code>\etocnumber</code> command . .	10
3. Arbitrarily many TOCs, and local ones too	5	4.3.2. The <code>\etocifnumbered</code> switch . . .	11
3.1. On manually adding layout commands to the .toc file	5	4.4. The <code>\etocfontxxx...</code> commands	11
3.2. Labeling and reusing elsewhere	6	4.4.1. This is a (pale) red subsubsection for illustrative purposes	11
3.3. Local table of contents for this part	7		

6. A comparison

5. Am I also in red?	12	8. The starred variants of the <code>\table-</code> <code>ofcontents</code> etc... commands	18
An unnumbered section	12	9. Table of contents for part III	19
6. A comparison	12	IV. Using and customizing <code>etoc</code>	21
III. Package commands for global styles	16	10. Summary of the main style commands	21
7. Specifying the toc display style	16	10.1. Setting up local styles . . .	21
7.1. The command <code>\etocruled-</code> <code>style</code>	16	10.2. Setting up global styles . .	21
7.2. The command <code>\etocmul-</code> <code>ticolstyle</code>	17	10.3. Displaying Tables of Con- tents	22
7.2.1. The command <code>\etoc-</code> <code>tocstyle</code>	17	10.4. Labels and references . . .	22
7.2.1.1. <code>\etoc-</code> <code>tocstyle-</code> <code>with-</code> <code>marks</code> .	17	10.5. An example	22
7.3. A point of some importance	17	11. Customizing <code>etoc</code>	24
7.3.0.2. Do we really want paragraph entries in the TOC? .	18	11.1. Customizing the <code>etoc</code> line styles	24
7.3.0.3. really? .	18	11.2. Customizing the global styles	25
7.4. The command <code>\etocruled</code>	18	V. Tips	27
7.5. The commands <code>\etoc-</code> <code>framedstyle</code> and <code>\etoc-</code> <code>framed</code>	18	12. ... and tricks	27
7.6. The command <code>\etocset-</code> <code>tocstyle</code>	18	12.1. Hacking framed parboxes .	27
7.7. The compatibility mode <code>\etocstandarddisplay-</code> <code>style</code>	18	12.2. Interverting the levels . . .	27
		12.3. All subsections of this doc- ument	28
		12.4. Compatibility with other packages	28
		12.5. TeXnical matters	29
		12.6. Errors and catastrophes . .	29
		VI. The code	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)

Instructing `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 `\etoc-skipfirstprefix` 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 12)

III Package commands for global styles

7 Specifying the toc display style (page 16)

- 7.1 The command `\etocruledstyle` (p. 16)

- 7.2 The command `\etocmulticolstyle` (p. 17)

The command `\etoctocstyle`.

- 7.3 A point of some importance (p. 17)

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

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

- 7.6 The command `\etocsettocstyle` (p. 18)

- 7.7 The compatibility mode `\etoc-standarddisplaystyle` (p. 18)

8 The starred variants of the `\tableofcontents` etc... commands (page 18)

9 Table of contents for part III (page 19)

IV Using and customizing `etoc`

10 Summary of the main style commands (page 21)

- 10.1 Setting up local styles (p. 21)

- 10.2 Setting up global styles (p. 21)

- 10.3 Displaying Tables of Contents (p. 22)

10.4 Labels and references (p. 22)

10.5 An example (p. 22)

11 Customizing `etoc` (page 24)

11.1 Customizing the `etoc` line styles (p. 24)

11.2 Customizing the global styles (p. 25)

12.2 Interverting the levels (p. 27)

12.3 All subsections of this document (p. 28)

12.4 Compatibility with other packages (p. 28)

12.5 T_EXnical matters (p. 29)

12.6 Errors and catastrophes (p. 29)

V Tips

12 ... and tricks (page 27)

12.1 Hacking framed parboxes (p. 27)

VI The code

13 Implementation (page 30)

Part III.

Package commands for global styles

```
\setcounter{tocdepth}{-3}
\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 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:

7. Specifying the toc display style

```
\renewcommand{\etocdisplayrule}{\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\fboxsep1ex  
  \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 `\etocdisplaystyle`

```
\etocdisplaystyle[<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.

7.2.1.1. `\etocdisplaystylewithmarks[<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 `\etocdisplaystylewithmarksnouc` 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 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.

9. Table of contents for part III

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
\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{\etocbottomrule}{\hrule height 1pt}
\renewcommand{\etocbottomrulecolorcmd}{\color{red!25}}
% set up the left and right rules
\renewcommand{\etoclefttrule}{\vrule width 5pt}
\renewcommand{\etocrighttrule}{\vrule width 5pt}
\renewcommand{\etoclefttrulecolorcmd}{\color{red!25}}
\renewcommand{\etocrighttrulecolorcmd}{\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|.}}}}
```

Contents

```
% 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. 16

The command `\etocmulticolstyle`

7.2, p. 17

The command `\etocstyle`

A point of some importance

7.3, p. 17

The command `\etocruled`

7.4, p. 18

The commands `\etocframedstyle` and `\etocframed`

7.5, p. 18

The command `\etocsettocstyle`

7.6, p. 18

The compatibility mode `\etocstandarddisplaystyle`

7.7, p. 18

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

8, p. 18

Table of contents for part III

9, p. 19

We can also display it in the document standard manner:¹⁴

```
\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>	16
7.2. The command <code>\etocmulticolstyle</code>	17
7.2.1. The command <code>\etocstyle</code>	17
7.3. A point of some importance	17
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>	18

¹⁴the present document uses the `scrartcl` class, and we check here that the **etoc** compatibility mode does respect the customizing done via the class commands.

8. The starred variants of the <code>\tableofcontents</code> etc... commands	18
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 10, p. 21	An example 10.5, p. 22
Setting up local styles . . . 10.1, p. 21	Customizing etoc 11, p. 24
Setting up global styles . . . 10.2, p. 21	Customizing the etoc line styles
Displaying Tables of Contents 10.3, p. 22 11.1, p. 24
Labels and references . . . 10.4, p. 22	Customizing the global styles
 11.2, p. 25

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

I got motivated by a question¹⁵ 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 `\etocframedstyle` 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{\etoclefttrulecolorcmd}{\color{tocbackground}}
\renewcommand{\etocrighttrulecolorcmd}{\color{tocbackground}}
\renewcommand{\etocbottomrulecolorcmd}{\color{tocbackground}}
\renewcommand{\etoctoprerulecolorcmd}{\color{tocbackground}}
```

¹⁵tex.stackexchange.com/questions/83184

10. Summary of the main style commands

```

\renewcommand{\etoclefttrule}{\vrule width 1cm}
\renewcommand{\etocrighttrule}{\vrule width .5cm}
\renewcommand{\etocbottomrule}{\hrule height 12pt}
\renewcommand{\etocoprurule}{\hrule height 12pt}

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

\newcommand\shiftedwhiterule[2]{%
  \hbox to \linewidth{\color{white}%
    \hskip#1\leaders\vrule height1pt\hfil}\nointerlineskip
  \vskip#2}

\etocsetstyle{subsubsection}{\etocskipfirstprefix}
{\shiftedwhiterule{\leftskip}{6pt}}
{\sffamily\footnotesize
\leftskip2.5cm\hangindent1cm\rightskip1cm\noindent
\hbox to 1cm{\color{subsecnum}\etocnumber\hss}%
\color{black}\etocname\leaders\hbox to .2cm{\hss.}\hfill
\rlap{\hbox to 1cm{\hss\etocpage\hskip.2cm}}\par
\nointerlineskip\vskip3pt}
{}}

\etocsetstyle{subsection}{\etocskipfirstprefix}
{\shiftedwhiterule{1.5cm}{6pt}}
{\sffamily\small
\leftskip1.5cm\hangindent1cm\rightskip1cm\noindent
\hbox to 1cm{\color{subsecnum}\etocnumber\hss}%
\color{black}\etocname\leaders\hbox to .2cm{\hss.}\hfill
\rlap{\hbox to 1cm{\hss\etocpage\hskip.2cm}}\par
\nointerlineskip\vskip6pt}
{}}

\newcommand{\coloredstuff}[2]{%
  \leftskip0pt\rightskip0pt\parskip0pt
  \fboxsep0pt % \colorbox uses \fboxsep also when no frame!
  \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]{}
\tableofcontents \label{toc:clone} \ref{toc:globalcmds}

```

11. Customizing **etoc**

`\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>	16
7.2	The command <code>\etocmulticolstyle</code>	17
7.2.1	The command <code>\etocstyle</code>	17
7.3	A point of some importance	17
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> ..	18
8	The starred variants of the <code>\tableofcontents</code> etc... commands	18
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\etocsepminustwo{4ex plus .5ex minus .5ex}
\newcommand\etocsepminusone{4ex plus .5ex minus .5ex}
\newcommand\etocsepzero{2.5ex plus .4ex minus .4ex}
```


11. Customizing *etoc*

```
\newcommand\etocseppone{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\etoclineleaders
{\hbox{\normalfont\normalsize\hbox to 2ex {\hss.\hss}}}

\newcommand\etocabbrevpagename{p.\sim}
\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*.¹⁶

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 \LaTeX default manner (existence of the macros `\l@something...`).

Using the `\etoclines` (obviously) or even just one `\etocsetstyle` command turns off the compatibility mode (all of this obey the scoping mechanism of $\text{\TeX}/\text{\LaTeX}$ 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}
```

¹⁶with 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.

11. Customizing *etoc*

```
\newcommand\etoccolumnsep{2em}
\newcommand\etocmulticolsep{0ex}
\newcommand\etocmulticolpretolerance{-1}
\newcommand\etocmulticoltolerance{200}
\newcommand\etocdefaultnbcol{2}
\newcommand\etocinnertopsep{2ex}
\newcommand\etoctoprule{\hrule}
\newcommand\etoctoprulecolorcmd{\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\etocleftrulecolorcmd{\relax}
\newcommand\etocrightrulecolorcmd{\relax}
\newcommand\etocbottomrulecolorcmd{\relax}

\newcommand\etocbkgcolorcmd{\relax}

% hooks
\newcommand\etocframedmphook{\relax}
\newcommand\etocaftertitlehook{\relax}
```

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

The `\dots 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{\etoctoprule}{\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\etocparbox{\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	The commands <code>\etocframedstyle</code> and <code>\etocframed</code>
7, p. 16	7.5, p. 18
The command <code>\etocruledstyle</code>	The command <code>\etocsettocstyle</code>
7.1, p. 16	7.6, p. 18
The command <code>\etocmulticolstyle</code>	The compatibility mode <code>\etocstandarddisplaystyle</code>
7.2, p. 17	7.7, p. 18
The command <code>\etocstyle</code> .	The starred variants of the <code>\tableof-contents</code> etc... commands 8, p. 18
A point of some importance . 7.3, p. 17	
The command <code>\etocruled</code> . 7.4, p. 18	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)	xi. The <code>\etocfontxxx...</code> commands (4.4, p. 11)	xxii. Labels and references (10.4, p. 22)
ii. <code>\etocsettocstyle</code> for the toc display (2.1, p. 4)	xii. The command <code>\etocruledstyle</code> (7.1, p. 16)	xxiii. An example (10.5, p. 22)
iii. <code>\etocsetstyle</code> for the line styles (2.2, p. 4)	xiii. The command <code>\etocmulticolstyle</code> (7.2, p. 17)	xxiv. Customizing the <code>etoc</code> line styles (11.1, p. 24)
iv. Compatibility mode (2.3, p. 5)	xiv. A point of some importance (7.3, p. 17)	xxv. Customizing the global styles (11.2, p. 25)
v. On manually adding layout commands to the <code>.toc</code> file (3.1, p. 5)	xv. The command <code>\etocruled</code> (7.4, p. 18)	xxvi. Hacking framed parboxes (12.1, p. 27)
vi. Labeling and reusing elsewhere (3.2, p. 6)	xvi. The commands <code>\etocframedstyle</code> and <code>\etocframed</code> (7.5, p. 18)	xxvii. Interverting the levels (12.2, p. 27)
vii. Local table of contents for this part (3.3, p. 7)	xvii. The command <code>\etocsettocstyle</code> (7.6, p. 18)	xxviii. All subsections of this document (12.3, p. 28)
viii. An example (4.1, p. 8)	xviii. The compatibility mode <code>\etocstandarddisplaystyle</code> (7.7, p. 18)	xxix. Compatibility with other packages (12.4, p. 28)
ix. The <code>\etocsetlevel</code> command (4.2, p. 10)	xix. Setting up local styles (10.1, p. 21)	xxx. T _E Xnical matters (12.5, p. 29)
x. Further informations and let's make this a very long title to see how it will look like (4.3, p. 10)	xx. Setting up global styles (10.2, p. 21)	xxxi. Errors and catastrophes (12.6, p. 29)
	xxi. Displaying Tables of Contents (10.3, p. 22)	

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 (if it exists) are read into memory by `etoc` once, at the time of `\usepackage{etoc}`. The `.toc` file will be opened for write operations only at

the time of the first TOC displaying command.^{17 18}

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. The correct local table of contents will be displayed only on the next `latex` run.

This is the only kind of data written by `etoc` to the `.toc` file, but of course it does mean that if you remove usage of `etoc` from the document, you should trash the `.toc` file. The commands written by `etoc` to the `.toc` file do nothing when not used by `etoc` itself, so other packages may also read the `.toc` file and execute its contents.¹⁹

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.

The macros `\l@section`, `\l@subsection` . . ., more generally `\l@kind` where *kind* has been declared to `etoc` with `\etocsetlevel` may be customized throughout the document and it is their current definitions which will be used in compatibility mode.²⁰

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.

¹⁷up to version v1.05 both things were done by `etoc` at `\begin{document}`.

¹⁸As soon as the `.toc` file is opened for write, it becomes empty, so its contents should be read by other packages or user defined macros before the first TOC displaying command (this is the same situation in the standard classes with the standard `\tableofcontents` command.)

¹⁹see previous footnote.

²⁰in versions up to v1.05, the default definitions of `\l@section` etc . . . were re-established (to their values at the `\begin{document}`, or at the latest `\etocsetlevel`) only temporarily by `etoc` at the time of displaying a TOC in compatibility mode. This made it impossible to customize them by trying to add for example some font changing prefix to the pre-existing macro (but customization via commands of packages like `tocloft` which re-define `\l@section` to have customizing hooks did work as advertised). The new implementation in v1.06 changes the meaning of `\l@section` and similar macros only at the time of actually typesetting the tables 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).²¹

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 T_EX installation.

13. Implementation

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

```

1 \ProvidesPackage{etoc}
2 [2012/12/07 v1.06 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
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

```

²¹ Note however that an error results when one adds `\usepackage{hyperref}` to a document having some `\label's` and `\ref's`, and that one must run latex another time or trash the `.aux` file to get rid of the error. This is related neither with **etoc** nor with the `.toc` file.

13. Implementation

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
28 \newif\ifEtoc@tocopenedforwrite
```

placeholder for comments

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

placeholder for comments

```
48 \def\Etoc@levellist{}
49 \def\Etoc@newlevel#1{%
50   \def\Etoc@levellist@elt{\noexpand\Etoc@levellist@elt\noexpand}%
51   \edef\Etoc@levellist{\Etoc@levellist\Etoc@levellist@elt#1}}
52 \def\etocsetlevel#1#2{%
53   \let\Etoc@next\@firstofone
54   \ifcase#2\or\or\or\or\or
55     \else\ifnum#2=\m@ne\else\ifnum#2=-\tw@\else
56       \PackageWarning{etoc}
57         {unexpected value ‘#2’ in \string\etocsetlevel.^^J%
58         Should be -2,-1, 0, 1, 2, 3, 4 or 5. Set to -1}%
59       \expandafter\def\csname Etoc@#1@\endcsname{minusone}%
60       \expandafter\let\csname Etoc@#1@\endcsname\m@ne
61   \let\Etoc@next\@gobble\fi\fi\fi
62   \expandafter\Etoc@newlevel\csname l@#1\endcsname
63   \Etoc@next
64   {\ifcase#2\relax
65     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@zero@@
66     \expandafter\def \csname Etoc@#1@\endcsname{zero}%
67   \or
68     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@one@@
69     \expandafter\def \csname Etoc@#1@\endcsname{one}%
```

13. Implementation

```

70 \or
71   \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@two@@
72   \expandafter\def \csname Etoc@#1@\endcsname{two}%
73 \or
74   \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@three@@
75   \expandafter\def \csname Etoc@#1@\endcsname{three}%
76 \or
77   \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@four@@
78   \expandafter\def \csname Etoc@#1@\endcsname{four}%
79 \or
80   \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@five@@
81   \expandafter\def \csname Etoc@#1@\endcsname{five}%
82 \else
83   \ifnum#2=\m@ne
84     \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@minusone@@
85     \expandafter\def \csname Etoc@#1@\endcsname{minusone}%
86   \else
87     \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@minustwo@@
88     \expandafter\def \csname Etoc@#1@\endcsname{minustwo}%
89   \fi
90 \fi}}
91 \etocsetlevel{book}{-2}
92 \etocsetlevel{part}{-1}
93 \etocsetlevel{chapter}{0}
94 \etocsetlevel{section}{1}
95 \etocsetlevel{subsection}{2}
96 \etocsetlevel{subsubsection}{3}
97 \etocsetlevel{paragraph}{4}
98 \etocsetlevel{subparagraph}{5}

placeholder for comments
99 \AtBeginDocument{%
100 \ifpackageloaded{hyperref}
101 {\Etoc@hyperreftrue
102 \def\Etoc@et@hop#1#2#3#4#5{#1{#3}{#4}{#5}#2}}
103 {\Etoc@hyperreffalse
104 \def\Etoc@et@hop#1#2#3#4{#1{#3}{#4}#2}}}

placeholder for comments
105 \def\Etoc@swa#1{%
106   \Etoc@et@hop
107     {\Etoc@savedcontentsline{#1}}
108     {\Etoc@prefix\Etoc@contents}}
109 \def\Etoc@swb#1{%
110   \Etoc@et@hop
111     {\Etoc@savedcontentsline{#1}}
112     {\Etoc@contents}}
113 \let\etocskipfirstprefix\@thirdofthree

placeholder for comments
114 \def\Etoc@setflags #1{%
115   \ifcase #1\relax
116     \global\Etoc@vfalse
117     \global\Etoc@ivfalse

```


13. Implementation

```

118     \global\Etoc@iiifalse
119     \global\Etoc@iifalse
120     \global\Etoc@ifalse
121     \global\Etoc@true
122 \or
123     \global\Etoc@vfalse
124     \global\Etoc@ivfalse
125     \global\Etoc@iiifalse
126     \global\Etoc@iifalse
127     \global\Etoc@ittrue
128 \or
129     \global\Etoc@vfalse
130     \global\Etoc@ivfalse
131     \global\Etoc@iiifalse
132     \global\Etoc@iittrue
133 \or
134     \global\Etoc@vfalse
135     \global\Etoc@ivfalse
136     \global\Etoc@iiittrue
137 \or
138     \global\Etoc@vfalse
139     \global\Etoc@ivtrue
140 \or
141     \global\Etoc@vtrue
142 \else
143     \ifnum#1=\m@ne
144     \global\Etoc@vfalse
145     \global\Etoc@ivfalse
146     \global\Etoc@iiifalse
147     \global\Etoc@iifalse
148     \global\Etoc@ifalse
149     \global\Etoc@false
150     \global\Etoc@jtrue
151 \else
152     \global\Etoc@vfalse
153     \global\Etoc@ivfalse
154     \global\Etoc@iiifalse
155     \global\Etoc@iifalse
156     \global\Etoc@ifalse
157     \global\Etoc@false
158     \global\Etoc@jfalse
159     \global\Etoc@jjtrue
160 \fi
161 \fi}

placeholder for comments
162 \def\Etoc@etoccontentsline#1{%
163     \global\expandafter\let\expandafter\Etoc@tmp\csname Etoc@#1@@\endcsname
164     \Etoc@skipthisonefalse
165     \ifEtoc@hyperref
166         \def\Etoc@next{\expandafter\@gobbletwo\@gobblefour}%
167     \else
168         \def\Etoc@next{\expandafter\@gobble\@gobblefour}%

```

13. Implementation

```
169 \fi
170 \ifEtoc@localtoc
171   \let\Etoc@prenext\relax
172   \ifEtoc@stoptoc
173     \Etoc@skipthisonetrue
174   \fi
175   \ifnum\Etoc@tmp<\Etoc@localtop
176     \def\Etoc@prenext{\global\Etoc@stoptoctrue}%
177     \Etoc@skipthisonetrue
178   \fi
179   \ifEtoc@notactive
180     \def\Etoc@prenext{\Etoc@setflags{\Etoc@tmp}}%
181     \Etoc@skipthisonetrue
182   \fi
183   \Etoc@prenext
184 \fi
185 \ifnum\c@tocdepth<\Etoc@tmp\relax\else
186 \ifEtoc@skipthisone\else
187 \global\let\Etoc@next\relax
188 \ifcase\Etoc@tmp
189   \ifEtoc@v \Etoc@end@five\fi
190   \ifEtoc@iv \Etoc@end@four\fi
191   \ifEtoc@iii \Etoc@end@three\fi
192   \ifEtoc@ii \Etoc@end@two\fi
193   \ifEtoc@i \Etoc@end@one\fi
194   \ifEtoc@ \else \def\Etoc@next{\Etoc@begin@zero}\fi
195   \def\Etoc@contents{\Etoc@contents@zero}%
196   \def\Etoc@prefix{\Etoc@prefix@zero}%
197 \or
198   \ifEtoc@v \Etoc@end@five\fi
199   \ifEtoc@iv \Etoc@end@four\fi
200   \ifEtoc@iii \Etoc@end@three\fi
201   \ifEtoc@ii \Etoc@end@two\fi
202   \ifEtoc@i \else \def\Etoc@next{\Etoc@begin@one}\fi
203   \def\Etoc@contents{\Etoc@contents@one}%
204   \def\Etoc@prefix{\Etoc@prefix@one}%
205 \or
206   \ifEtoc@v \Etoc@end@five\fi
207   \ifEtoc@iv \Etoc@end@four\fi
208   \ifEtoc@iii \Etoc@end@three\fi
209   \ifEtoc@ii \else \def\Etoc@next{\Etoc@begin@two}\fi
210   \def\Etoc@contents{\Etoc@contents@two}%
211   \def\Etoc@prefix{\Etoc@prefix@two}%
212 \or
213   \ifEtoc@v \Etoc@end@five\fi
214   \ifEtoc@iv \Etoc@end@four\fi
215   \ifEtoc@iii \else \def\Etoc@next{\Etoc@begin@three}\fi
216   \def\Etoc@contents{\Etoc@contents@three}%
217   \def\Etoc@prefix{\Etoc@prefix@three}%
218 \or
219   \ifEtoc@v \Etoc@end@five\fi
220   \ifEtoc@iv \else \def\Etoc@next{\Etoc@begin@four}\fi
```

13. Implementation

```

221 \def\Etoc@contents{\Etoc@contents@four}%
222 \def\Etoc@prefix{\Etoc@prefix@four}%
223 \or
224 \ifEtoc@v \else \def\Etoc@next{\Etoc@begin@five}\fi
225 \def\Etoc@contents{\Etoc@contents@five}%
226 \def\Etoc@prefix{\Etoc@prefix@five}%
227 \else
228 \ifnum\Etoc@tmp=\m@ne
229 \ifEtoc@v \Etoc@end@five\fi
230 \ifEtoc@iv \Etoc@end@four\fi
231 \ifEtoc@iii \Etoc@end@three\fi
232 \ifEtoc@ii \Etoc@end@two\fi
233 \ifEtoc@i \Etoc@end@one\fi
234 \ifEtoc@ \Etoc@end@zero\fi
235 \ifEtoc@j \else \def\Etoc@next{\Etoc@begin@minusone}\fi
236 \def\Etoc@contents{\Etoc@contents@minusone}%
237 \def\Etoc@prefix{\Etoc@prefix@minusone}%
238 \else
239 \ifEtoc@v \Etoc@end@five\fi
240 \ifEtoc@iv \Etoc@end@four\fi
241 \ifEtoc@iii \Etoc@end@three\fi
242 \ifEtoc@ii \Etoc@end@two\fi
243 \ifEtoc@i \Etoc@end@one\fi
244 \ifEtoc@ \Etoc@end@zero\fi
245 \ifEtoc@j \Etoc@end@minusone\fi
246 \ifEtoc@jj \else \def\Etoc@next{\Etoc@begin@minustwo}\fi
247 \def\Etoc@contents{\Etoc@contents@minustwo}%
248 \def\Etoc@prefix{\Etoc@prefix@minustwo}%
249 \fi
250 \fi
251 \ifnum\Etoc@tmp=\m@ne\Etoc@parttrue\else\Etoc@partfalse\fi
252 \Etoc@setflags{\Etoc@tmp}%
253 \fi\fi
254 \Etoc@next
255 \@firstoftwo{\Etoc@swa{#1}}{\Etoc@swb{#1}}

```

placeholder for comments

```

256 \def\Etoc@lxyz #1#2{\@namedef{etocname }{\leavevmode #1\xspace}%
257 \@namedef{etocpage }{\leavevmode #2\xspace}%
258 \Etoc@getnb #1\relax\relax\etoc@
259 \ifEtoc@number\else\ifEtoc@part
260 \Etoc@getit #1\hspace\relax\etoc@\fi\fi}

```

placeholder for comments

```

261 \def\Etoc@getnb #1{\let\Etoc@next\Etoc@getnb@nohyp
262 \ifEtoc@hyperref\ifx #1\hyper@linkstart
263 \let\Etoc@next\Etoc@getnb@hyp\fi\fi\Etoc@next #1}
264 \def\Etoc@getit #1{\let\Etoc@next\Etoc@getit@nohyp
265 \ifEtoc@hyperref\ifx #1\hyper@linkstart
266 \let\Etoc@next\Etoc@getit@hyp\fi\fi\Etoc@next #1}

```

placeholder for comments

```

267 \def\Etoc@getnb@nohyp #1#2#3\etoc@{%
268 \ifx #1\numberline

```

13. Implementation

```

269 \@namedef{etocnumber }{\leavevmode #2\hspace}\Etoc@numbertrue
270 \else
271 \@namedef{etocnumber }{\leavevmode\hspace}\Etoc@numberfalse
272 \fi}

placeholder for comments
273 \def\Etoc@getnb@hyp #1#2#3#4#5#6\etoc@{%
274 \def\Etoc@getnbr ##1##2##3\etoc@{%
275 \ifx ##1\numberline
276 \@namedef{etocnumber }{\leavevmode #1{#2}{#3}{##2}#5\hspace}%
277 \Etoc@numbertrue
278 \else
279 \@namedef{etocnumber }{\leavevmode\hspace}%
280 \Etoc@numberfalse
281 \fi}%
282 \Etoc@getnbr #4\relax\relax\etoc@}

placeholder for comments
283 \def\Etoc@getit@nohyp #1\hspace#2#3\etoc@{%
284 \def\Etoc@getname ##1\hspace\relax\etoc@{%
285 \@namedef{etocname }{\leavevmode ##1\hspace}}%
286 \ifx \relax#2\else
287 \@namedef{etocnumber }{\leavevmode #1\hspace}%
288 \Etoc@numbertrue
289 \Etoc@getname #3\etoc@\fi}

placeholder for comments
290 \def\Etoc@getit@hyp #1#2#3#4#5#6\etoc@{%
291 \def\Etoc@getname ##1\hspace\relax\etoc@{%
292 \@namedef{etocname }{\leavevmode #1{#2}{#3}{##1}#5\hspace}}%
293 \def\Etoc@getnbr ##1\hspace##2##3\etoc@{%
294 \ifx\relax##2\else
295 \@namedef{etocnumber }{\leavevmode #1{#2}{#3}{##1}#5\hspace}%
296 \Etoc@numbertrue
297 \Etoc@getname ##3\etoc@\fi}%
298 \Etoc@getnbr #4\hspace\relax\etoc@}

placeholder for comments
299 \edef\etocpage{\noexpand\protect\expandafter\noexpand
300 \csname etocpage \endcsname}
301 \edef\etocname{\noexpand\protect\expandafter\noexpand
302 \csname etocname \endcsname}
303 \edef\etocnumber{\noexpand\protect\expandafter\noexpand
304 \csname etocnumber \endcsname}

placeholder for comments
305 \@namedef{etocifnumbered }{\ifEtoc@number
306 \expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}

placeholder for comments
307 \edef\etocifnumbered{\noexpand\protect\expandafter\noexpand
308 \csname etocifnumbered \endcsname}

placeholder for comments
309 \def\Etoc@readtoc#1{%

```

13. Implementation

```

310 \ifeof #1
311   \let\Etoc@nextread\@gobble
312   \global\Etoc@toctoks=\expandafter{\the\Etoc@toctoks}%
313 \else
314   \let\Etoc@nextread\Etoc@readtoc
315   \read #1 to \Etoc@buffer
316   \Etoc@toctoks=\expandafter\expandafter\expandafter
317     {\expandafter\the\expandafter\Etoc@toctoks\Etoc@buffer}%
318 \fi
319 \Etoc@nextread{#1}%
320 }
321 \IfFileExists{\jobname .toc}
322   {{\endlinechar=-1 \makeatletter
323     \newread\Etoc@tf
324     \openin\Etoc@tf\@filef@und
325     \Etoc@readtoc\Etoc@tf
326     \closein\Etoc@tf}}{\typeout{No file \jobname .toc.}}
placeholder for comments
327 \def\Etoc@starttoc{%
328 \begingroup
329 \makeatletter
330 \ifEtoc@hyperref
331   \ifx\hyper@last\@undefined
332     \ltx@ifpackageloaded {parskip}{\parskip \z@ }{}%
333     \IfFileExists {\jobname .toc}
334       {\Hy@WarningNoLine
335         {old toc file detected, not used; run LaTeX again (cheers from etoc)}}%
336     \global\Etoc@toctoks={}
337 {} \fi \fi
338 \if@filesw \newwrite \tf@toc
339 \immediate \openout \tf@toc \jobname .toc\relax \fi \@nobreakfalse
340 \endgroup}
placeholder for comments
341 \def\Etoc@toctoc{%
342   \global\Etoc@vfalse
343   \global\Etoc@ivfalse
344   \global\Etoc@iiifalse
345   \global\Etoc@iifalse
346   \global\Etoc@ifalse
347   \global\Etoc@false
348   \global\Etoc@jfalse
349   \global\Etoc@jjfalse
350 \ifEtoc@standard
351 \etoc@setstyle{@minustwo}{}{}{}{}%
352 \etoc@setstyle{@minusone}{}{}{}{}%
353 \etoc@setstyle{@zero}{}{}{}{}%
354 \etoc@setstyle{@one}{}{}{}{}%
355 \etoc@setstyle{@two}{}{}{}{}%
356 \etoc@setstyle{@three}{}{}{}{}%
357 \etoc@setstyle{@four}{}{}{}{}%
358 \etoc@setstyle{@five}{}{}{}{}%

```

13. Implementation

```

359 \else
360 \def\Etoc@levellist@elt##1{\let##1\Etoc@lxyz}%
361 \Etoc@levellist
362 \let\numberline\@gobble
363 \let\booknumberline\numberline
364 \let\partnumberline\numberline
365 \let\chapternumberline\numberline
366 \fi
367 \the\Etoc@toctoks
368 \ifEtoc@notactive\else
369   \ifEtoc@v   \Etoc@end@five\fi
370   \ifEtoc@iv  \Etoc@end@four\fi
371   \ifEtoc@iii \Etoc@end@three\fi
372   \ifEtoc@ii  \Etoc@end@two\fi
373   \ifEtoc@i   \Etoc@end@one\fi
374   \ifEtoc@    \Etoc@end@zero\fi
375   \ifEtoc@j   \Etoc@end@minusone\fi
376   \ifEtoc@jj  \Etoc@end@minustwo\fi
377 \fi}

placeholder for comments
378 \def\etoc@@startlocaltoc#1#2{%
379 \let\Etoc@next\relax
380   \ifEtoc@localtoc
381   \ifEtoc@notactive
382     \ifnum #1=#2\relax
383       \ifEtoc@jj \let\Etoc@localtop\Etoc@@minusone@@ \fi
384       \ifEtoc@j  \let\Etoc@localtop\Etoc@@zero@@   \fi
385       \ifEtoc@   \let\Etoc@localtop\Etoc@@one@@    \fi
386       \ifEtoc@i  \let\Etoc@localtop\Etoc@@two@@    \fi
387       \ifEtoc@ii \let\Etoc@localtop\Etoc@@three@@   \fi
388       \ifEtoc@iii \let\Etoc@localtop\Etoc@@four@@   \fi
389       \ifEtoc@iv \let\Etoc@localtop\Etoc@@five@@   \fi
390       \ifEtoc@v  \let\Etoc@localtop\Etoc@@six@@    \fi
391       \def\Etoc@next{\global\Etoc@notactivefalse
392         \global\Etoc@vfalse
393         \global\Etoc@ivfalse
394         \global\Etoc@iiifalse
395         \global\Etoc@iifalse
396         \global\Etoc@ifalse
397         \global\Etoc@false
398         \global\Etoc@jfalse
399         \global\Etoc@jjfalse}%
400       \fi
401     \fi\fi
402 \Etoc@next}
403 \let\etoc@startlocaltoc\@gobble

placeholder for comments
404 \long\def\Etoc@localtableofcontents#1{%
405   \edef\Etoc@tmp{#1}%
406   \ifnum\Etoc@tmp<\@ne
407     \PackageWarning{etoc}

```

13. Implementation

```

408      {Unknown toc id: run LaTeX to get references right}%
409      \leavevmode --unknown etoc ref: run latex again--\par
410      \let\Etoc@next\@gobble\else\let\Etoc@next\@firstofone\fi
411      \Etoc@next
412      {\edef\etoc@startlocaltoc##1{%
413        \noexpand\etoc@@startlocaltoc{##1}{#1}}
414      \Etoc@localtoctrue
415      \let\Etoc@localtop\Etoc@@minustwo@@
416      \global\Etoc@stoptocfalse
417      \global\Etoc@notactivetrue
418      \Etoc@tableofcontents}%
419      \endgroup\ifEtoc@mustclosegroup\expandafter\endgroup\fi}

placeholder for comments
420 \def\Etoc@getrefno #1#2\etoc@{#1}
421 \def\Etoc@getref #1{\@ifundefined{r@#1}{0}{\expandafter\expandafter\expandafter
422   \Etoc@getrefno\csname r@#1\endcsname\relax\etoc@}}
423 \def\Etoc@ref#1{\Etoc@localtableofcontents{\Etoc@getref{#1}}}
424 \def\Etoc@label#1{\label{#1}}%
425   \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
426 {\def\1{\Etoc@again}\expandafter
427 \gdef\1 {\futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}}

placeholder for comments
428 \def\Etoc@t@bleofcontents{%
429   \ifx\Etoc@nexttoken\label
430     \def\Etoc@next{\expandafter\Etoc@label\@gobble}\else
431     \ifx\Etoc@nexttoken\@sptoken
432       \let\Etoc@next\Etoc@again\else
433       \ifEtoc@tocwithid
434         \def\Etoc@next{\Etoc@localtableofcontents{\c@etoc@tocid}}%
435       \else
436         \ifx\Etoc@nexttoken\ref
437           \def\Etoc@next{\expandafter\Etoc@ref\@gobble}%
438         \else
439           \def\Etoc@next{\Etoc@localtocfalse
440             \global\Etoc@notactivefalse
441             \Etoc@tableofcontents
442             \endgroup
443             \ifEtoc@mustclosegroup\expandafter\endgroup\fi}%
444         \fi
445       \fi\fi\fi\Etoc@next}

placeholder for comments
446 \def\table@fcontents{\par
447   \refstepcounter{etoc@tocid}%
448   \Etoc@tocwithidfalse
449   \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
450 \def\localtable@fcontents{\par
451   \refstepcounter{etoc@tocid}%
452   \addtocontents{toc}
453   {\string\etoc@startlocaltoc\string{\arabic{etoc@tocid}\string}}%
454   \Etoc@tocwithidtrue
455   \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}

```

13. Implementation

placeholder for comments

```

456 \newcommand\etocaftertitlehook{}
457 \def\tableofcontents{\ifEtoc@tocopenedforwrite\else\Etoc@starttoc\fi
458   \global\Etoc@tocopenedforwritetrue
459   \begingroup
460   \def\etoc@startlocaltoc##1{\etoc@@startlocaltoc{##1}{\c@etoc@tocid}}%
461   \@ifstar
462   {\let\Etoc@aftertitlehook\relax\table@fcontents}
463   {\let\Etoc@aftertitlehook\etocaftertitlehook\table@fcontents}}
464 \def\localtableofcontents{\ifEtoc@tocopenedforwrite\else\Etoc@starttoc\fi
465   \global\Etoc@tocopenedforwritetrue
466   \begingroup\@ifstar
467   {\let\Etoc@aftertitlehook\relax\localtable@fcontents}
468   {\let\Etoc@aftertitlehook\etocaftertitlehook\localtable@fcontents}}

```

placeholder for comments

```

469 \newcommand\etocsettocstyle[2]{%
470 \long\def\Etoc@tableofcontents
471 {\ifnum\c@tocdepth>\Etoc@minf\let\Etoc@@next\@firstofone\else
472 \let\Etoc@@next\@gobble\fi
473 \Etoc@@next{#1\Etoc@aftertitlehook
474   \let\Etoc@savedcontentsline\contentsline
475   \let\contentsline\Etoc@etoccontentsline
476   \Etoc@toctoc\let\Etoc@nxt\relax
477   \ifEtoc@tocwithid\else
478   \ifEtoc@localtoc
479   \ifEtoc@notactive
480   \def\Etoc@nxt{\Etoc@localtocfalse
481     \global\Etoc@notactivefalse
482     \Etoc@toctoc}%
483   \fi\fi\fi\Etoc@nxt
484   #2}}
485 \newcommand\etocextndtocstyle[2]{%
486 \toks0=\expandafter{\Etoc@tableofcontents #2}%
487 \toks2={#1}%
488 \long\edef\Etoc@tableofcontents{\the\toks2 \the\toks0}}
489 %           % je me fais avoir ^ à chaque fois!

```

placeholder for comments

```

490 \newcommand\etocsetstyle{\Etoc@standardfalse\etoc@setstyle}
491 \long\def\etoc@setstyle#1#2#3#4#5{%
492 \long\expandafter\def
493   \csname Etoc@begin@\csname Etoc@#1@\endcsname\endcsname {#2}%
494 \long\expandafter\def
495   \csname Etoc@prefix@\csname Etoc@#1@\endcsname\endcsname {#3}%
496 \long\expandafter\def
497   \csname Etoc@contents@\csname Etoc@#1@\endcsname\endcsname {#4}%
498 \long\expandafter\def
499   \csname Etoc@end@\csname Etoc@#1@\endcsname\endcsname {#5}}

```

placeholder for comments

```

500 \newcommand\etocfontminustwo{\normalfont \LARGE \bfseries}
501 \newcommand\etocfontminusone{\normalfont \large \bfseries}
502 \newcommand\etocfontzero{\normalfont \large \bfseries}

```


13. Implementation

```

503 \newcommand\etocfontone{\normalfont \normalsize \bfseries}
504 \newcommand\etocfonttwo{\normalfont \normalsize}
505 \newcommand\etocfontthree{\normalfont \footnotesize}

placeholder for comments
506 \newcommand\etocsepminustwo{4ex \@plus .5ex \@minus .5ex}
507 \newcommand\etocsepminusone{4ex \@plus .5ex \@minus .5ex}
508 \newcommand\etocsepzero{2.5ex \@plus .4ex \@minus .4ex}
509 \newcommand\etocsepone{1.5ex \@plus .3ex \@minus .3ex}
510 \newcommand\etocseptwo{1ex \@plus .15ex \@minus .15ex}
511 \newcommand\etocsepthree{.25ex \@plus .05ex \@minus .05ex}

placeholder for comments
512 \newcommand\etocbaselinespreadminustwo{1}
513 \newcommand\etocbaselinespreadminusone{1}
514 \newcommand\etocbaselinespreadzero{1}
515 \newcommand\etocbaselinespreadone{1}
516 \newcommand\etocbaselinespreadtwo{1}
517 \newcommand\etocbaselinespreadthree{.9}

placeholder for comments
518 \newcommand\etocminustwoleftmargin{1.5em plus 0.5fil}
519 \newcommand\etocminustworightmargin{1.5em plus -0.5fil}
520 \newcommand\etocminusoneleftmargin{1em}
521 \newcommand\etocminusonerightmargin{1em}
522 \newcommand\etoclineleaders
523   {\hbox{\normalfont\normalsize\hb@xt@2ex {\hss.\hss}}}
524 \newcommand\etocabbrevpagename{p.~}
525 \newcommand\etocpartname{\partname}
526 \newcommand\etocbookname{Book}

placeholder for comments
527 \def\etoclines{\Etoc@standardfalse}

placeholder for comments
528 \etoc@setstyle{@minustwo}{\etocskipfirstprefix}{\pagebreak[3]}
529 {\begingroup \etocfontminustwo
530   \addvspace{\etocsepminustwo}%
531   \parindent \z@
532   \leftskip \etocminustwoleftmargin
533   \rightskip \etocminustworightmargin
534   \parfillskip \@flushglue
535   \vbox{\etocifnumbered{\etocbookname~\etocnumber:~}{ }\etocname
536     \baselineskip\etocbaselinespreadminustwo\baselineskip\par}%
537   \addvspace{\etocsepminusone}\nobreak
538 \endgroup}
539 {}%
540 \etoc@setstyle{@minusone}{\etocskipfirstprefix}{\pagebreak[3]}
541 {\begingroup \etocfontminusone
542   \addvspace{\etocsepminusone}%
543   \parindent \z@
544   \leftskip \etocminusoneleftmargin
545   \rightskip \etocminusonerightmargin
546   \parfillskip \@flushglue
547   \vbox{\etocpartname\etocifnumbered{ \etocnumber.~}{ }\etocname

```

13. Implementation

```

548 \baselineskip\etocbaselinespreadminusone\baselineskip\par}%
549 \addvspace{\etocsepzero}\nobreak
550 \endgroup}
551 {}%
552 \etoc@setstyle{@zero}
553 {\addpenalty\@highpenalty\etocskipfirstprefix}{\pagebreak[2]}
554 {\beginingroup
555   \etocfontzero
556   \addvspace{\etocsepzero}%
557   \parindent \z@ \parfillskip \@flushglue
558   \vbox{\etocifnumbered{\etocnumber\space}{\etocname
559     \baselineskip\etocbaselinespreadzero\baselineskip\par}%
560     \addvspace{\etocsepone}\addpenalty\@itempenalty
561   \endgroup}
562 {\addpenalty\@secpenalty}%
  placeholder for comments
563 \etoc@setstyle{@one}
564 {\addpenalty\@medpenalty\etocskipfirstprefix}{\pagebreak[1]}
565 {\beginingroup\etocfontone
566 \addvspace{\etocsepone}%
567 \parindent \z@ \parfillskip \z@
568 \setbox\z@\vbox{\parfillskip\@flushglue
569 \etocname\par\setbox\tw@\lastbox
570 \global\setbox\@ne\hbox{\unhbox\tw@}}}%
571 \dimen\z@=\wd\@ne
572 \setbox\z@=\etococlineleaders
573 \advance\dimen\z@\wd\z@
574 \etocifnumbered
575 {\setbox\tw@\hbox{\etocnumber, \etocabbrevpagename\etocpage}}
576 {\setbox\tw@\hbox{\etocabbrevpagename\etocpage}}}%
577 \advance\dimen\z@\wd\tw@
578 \ifdim\dimen\z@ < \linewidth
579   \vbox{\etocname\nobreak
580     \leaders\box\z@\hfil\box\tw@
581     \baselineskip\etocbaselinespreadone\baselineskip\par}
582   \else
583     \vbox{\etocname\nobreak
584       \ifdim\wd\@ne<\linewidth\leaders\copy\z@\hfil\break\fi
585       \hbox{} \leaders\box\z@\hfil\box\tw@
586       \baselineskip\etocbaselinespreadone\baselineskip\par}
587 \fi
588 \addvspace{\etocseptwo}\addpenalty\@itempenalty
589 \endgroup}
590 {\addpenalty\@secpenalty}%
  placeholder for comments
591 \etoc@setstyle{@two}
592 {\addpenalty\@medpenalty\etocskipfirstprefix}
593 {\pagebreak[1]}
594 {\beginingroup\etocfonttwo
595 \addvspace{\etocseptwo}%
596 \parindent \z@ \parfillskip \z@

```

13. Implementation

```

597 \setbox\z@\vbox{\parfillskip\@flushglue
598 \etocname\par\setbox\tw@\lastbox
599 \global\setbox\@ne\hbox{\unhbox\tw@}}%
600 \dimen\z@=\wd\@ne
601 \setbox\z@=\etocheadlineleaders
602 \advance\dimen\z@\wd\z@
603 \etocifnumbered
604 {\setbox\tw@\hbox{\etocnumber, \etocabbrevpagename\etocpage}}
605 {\setbox\tw@\hbox{\etocabbrevpagename\etocpage}}%
606 \advance\dimen\z@\wd\tw@
607 \ifdim\dimen\z@ < \linewidth
608     \vbox{\etocname\nobreak
609     \leaders\box\z@\hfil\box\tw@
610     \baselineskip\etocbaselinespreadtwo\baselineskip\par}
611 \else
612     \vbox{\etocname\nobreak
613     \ifdim\wd\@ne<\linewidth\leaders\copy\z@\hfil\break\fi
614     \hbox{\leaders\box\z@\hfil\box\tw@
615     \baselineskip\etocbaselinespreadtwo\baselineskip\par}
616 \fi
617 \addvspace{\etocsepthree}\addpenalty\@itempenalty
618 \endgroup}
619 {\addpenalty\@secpenalty}%
  placeholder for comments
620 \etoc@setstyle{@three}
621 {\nopagebreak
622 \etocfontthree
623 \addvspace{\etocsepthree}%
624 \nointerlineskip\noindent
625 \etocskipfirstprefix}
626 {\allowbreak\,--\,}
627 {\etocname\unskip} % \xspace in \etocname
628 {\hfil\begingroup
629 \baselineskip\etocbaselinespreadthree\baselineskip
630 \par\endgroup
631 \addpenalty{-\@highpenalty}}%
  placeholder for comments
632 \etoc@setstyle{@four}{}}{}{}{}%
  placeholder for comments
633 \etoc@setstyle{@five}{}}{}{}{}%
634 }
  placeholder for comments
635 \newcommand\etocabovetocskip{3.5ex \@plus 1ex \@minus .2ex}
636 \newcommand\etocbelowtocskip{3.5ex \@plus 1ex \@minus .2ex}
637 \newcommand\etoccolumnsep{2em}
638 \newcommand\etocmulticolsep{0ex}
639 \newcommand\etocmulticolpretolerance{-1}
640 \newcommand\etocmulticoltolerance{200}
641 \newcommand\etocdefaultnbcol{2}
  placeholder for comments

```

13. Implementation

```

642 \newcommand\etocmulticolstyle[2][\etocdefaultnbcol]{%
643 \etocsettocstyle
644   {\addvspace{\etocabovetocskip}}%
645   \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
646   \else \let\Etoc@next\@secondoftwo\fi
647   \Etoc@next{%
648     \multicolpretolerance\etocmulticolpretolerance
649     \multicoltolerance\etocmulticoltolerance
650     \setlength{\columnsep}{\etoccolumnsep}%
651     \setlength{\multicolsep}{\etocmulticolsep}%
652     \begin{multicols}{#1}[#2\addvspace{\etocsepminusone}]}
653     {#2\addvspace{\etocsepminusone}%
654       \pretolerance\etocmulticolpretolerance
655       \tolerance\etocmulticoltolerance}}
656   {\ifnum #1>\@ne\let\Etoc@next\@firstofone
657   \else \let\Etoc@next\@gobble\fi
658   \Etoc@next{\end{multicols}}}%
659   \addvspace{\etocbelowtocskip}}

```

placeholder for comments

```

660 \newcommand\etocinnertopsep{2ex}
661 \newcommand\etocinnerbottomsep{3.5ex}
662 \newcommand\etocinnerleftsep{2em}
663 \newcommand\etocinnerrightsep{2em}
664 \newcommand\etoctoprule{\hrule}
665 \newcommand\etocleftrule{\vrule}
666 \newcommand\etocrightrule{\vrule}
667 \newcommand\etocbottomrule{\hrule}
668 \newcommand\etoctoprulecolorcmd{\relax}
669 \newcommand\etocbottomrulecolorcmd{\relax}
670 \newcommand\etocleftrulecolorcmd{\relax}
671 \newcommand\etocrightrulecolorcmd{\relax}

```

placeholder for comments

```

672 \newcommand\etoc@ruledheading[1]{%
673   \hb@xt@\linewidth{\color@begingroup
674     \hss #1\hss\hskip-\linewidth
675     \etoctoprulecolorcmd\leaders\etoctoprule\hss
676     \phantom{#1}%
677     \leaders\etoctoprule\hss\color@endgroup}}%
678   \nointerlineskip\vskip\etocinnertopsep}

```

placeholder for comments

```

679 \newcommand\etocruledstyle[2][\etocdefaultnbcol]{%
680 \etocsettocstyle
681   {\addvspace{\etocabovetocskip}}%
682   \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
683   \else \let\Etoc@next\@secondoftwo\fi
684   \Etoc@next
685     {\multicolpretolerance\etocmulticolpretolerance
686     \multicoltolerance\etocmulticoltolerance
687     \setlength{\columnsep}{\etoccolumnsep}%
688     \setlength{\multicolsep}{\etocmulticolsep}%
689     \begin{multicols}{#1}[\etoc@ruledheading{#2}]}

```

13. Implementation

```

690      {\etoc@ruledheading{#2}\nobreak
691        \pretolerance\etocmulticolpretolerance
692        \tolerance\etocmulticoltolerance}}
693    {\ifnum #1>\@ne\let\Etoc@next\@firstofone
694      \else \let\Etoc@next\@gobble\fi
695      \Etoc@next{\end{multicols}}}%
696    \addvspace{\etocbelowtocskip}}}

placeholder for comments

697 \newcommand\etocframedmphook{\relax}
698 \newcommand\etocbkgcolorcmd{\relax}
699 \newcommand\Etoc@relax{\relax}
700 \newbox\etoc@framed@titlebox
701 \newbox\etoc@framed@contentsbox
702 \newcommand\etocframedstyle[2][\etocdefaultnbcol]{%
703 \etocsettocstyle{%
704   \addvspace{\etocabovetocskip}%
705   \sbox\z@{#2}%
706   \dimen\z@\dp\z@
707   \ifdim\wd\z@<\linewidth \dp\z@\z@ \else \dimen\z@\z@ \fi
708   \setbox\etoc@framed@titlebox=\hb@xt@\linewidth{\color@begingroup
709     \hss
710     \ifx\etocbkgcolorcmd\Etoc@relax\else
711       \sbox\tw@{\color{white}%
712         \vrule\@width\wd\z@\@height\ht\z@\@depth\dimen\z@}%
713       \ifdim\wd\z@<\linewidth \dp\tw@\z@\fi
714       \box\tw@
715       \hskip-\wd\z@
716     \fi
717     \copy\z@
718     \hss
719     \hskip-\linewidth
720     \etocoprulcolorcmd\leaders\etocoprul\hss%
721     \hskip\wd\z@
722     \etocoprulcolorcmd\leaders\etocoprul\hss\color@endgroup}%
723   \setbox\z@\hbox{\etocleftrule\etocrightrule}%
724   \dimen\tw@\linewidth\advance\dimen\tw@-\wd\z@
725   \advance\dimen\tw@-\etocinnerleftsep
726   \advance\dimen\tw@-\etocinnerrightsep
727   \setbox\etoc@framed@contentsbox=\vbox\bgroup
728     \hsize\dimen\tw@
729     \kern\dimen\z@
730     \vskip\etocinnertopsep
731     \hbox\bgroup
732     \begin{minipage}{\hsize}%
733     \etocframedmphook
734     \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
735     \else \let\Etoc@next\@secondoftwo\fi
736     \Etoc@next
737     {\multicolpretolerance\etocmulticolpretolerance
738      \multicoltolerance\etocmulticoltolerance
739      \setlength{\columnsep}{\etoccolumnsep}%
740      \setlength{\multicolsep}{\etocmulticolsep}%

```

13. Implementation

```

741     \begin{multicols}{#1}}
742     {\pretolerance\etocmulticolpretolerance
743      \tolerance\etocmulticoltolerance}}
744     {\ifnum #1>\@ne\let\Etoc@next\@firstofone
745      \else \let\Etoc@next\@gobble\fi
746     \Etoc@next{\end{multicols}\unskip}%
747     \end{minipage}%
748     \egroup
749     \vskip\etocinnerbottomsep
750     \egroup
751     \vbox{\hsize\linewidth
752           \ifx\etocbkgcolorcmd\Etoc@relax\else
753             \kern\ht\etoc@framed@titlebox
754             \kern\dp\etoc@framed@titlebox
755             \hb@xt@\linewidth{\color@begingroup
756             \etocleftrulecolorcmd\etocleftrule
757             \etocbkgcolorcmd
758             \leaders\vrule
759               \@height\ht\etoc@framed@contentsbox
760               \@depth\dp\etoc@framed@contentsbox
761             \hss
762             \etocrightrulecolorcmd\etocrightrule
763             \color@endgroup}\nointerlineskip
764             \vskip-\dp\etoc@framed@contentsbox
765             \vskip-\ht\etoc@framed@contentsbox
766             \vskip-\dp\etoc@framed@titlebox
767             \vskip-\ht\etoc@framed@titlebox
768           \fi
769           \box\etoc@framed@titlebox\nointerlineskip
770           \hb@xt@\linewidth{\color@begingroup
771           {\etocleftrulecolorcmd\etocleftrule}%
772           \hss\box\etoc@framed@contentsbox\hss
773           \etocrightrulecolorcmd\etocrightrule\color@endgroup}
774           \nointerlineskip
775           \vskip\ht\etoc@framed@contentsbox
776           \vskip\dp\etoc@framed@contentsbox
777           \hb@xt@\linewidth{\color@begingroup\etocbottomrulecolorcmd
778           \leaders\etocbottomrule\hss\color@endgroup}}
779     \addvspace{\etocbelowtocskip}}}

placeholder for comments

780 \newcommand\etoc@multicoltoc[2][\etocdefaultnbcol]{%
781 \etocmulticolstyle[#1]{#2}%
782 \tableofcontents}
783 \newcommand\etoc@multicoltoCi[2][\etocdefaultnbcol]{%
784 \etocmulticolstyle[#1]{#2}%
785 \tableofcontents*}
786 \newcommand\etoc@local@multicoltoc[2][\etocdefaultnbcol]{%
787 \etocmulticolstyle[#1]{#2}%
788 \localtableofcontents}
789 \newcommand\etoc@local@multicoltoCi[2][\etocdefaultnbcol]{%
790 \etocmulticolstyle[#1]{#2}%
791 \localtableofcontents*}

```

13. Implementation

placeholder for comments

```
792 \newcommand\etoc@ruledtoc[2][\etocdefaultnbcol]{%
793 \etocruledstyle[#1]{#2}%
794 \tableofcontents}
795 \newcommand\etoc@ruledtoci[2][\etocdefaultnbcol]{%
796 \etocruledstyle[#1]{#2}%
797 \tableofcontents*}
798 \newcommand\etoc@local@ruledtoc[2][\etocdefaultnbcol]{%
799 \etocruledstyle[#1]{#2}%
800 \localtableofcontents}
801 \newcommand\etoc@local@ruledtoci[2][\etocdefaultnbcol]{%
802 \etocruledstyle[#1]{#2}%
803 \localtableofcontents*}
```

placeholder for comments

```
804 \newcommand\etoc@framedtoc[2][\etocdefaultnbcol]{%
805 \etocframedstyle[#1]{#2}%
806 \tableofcontents}
807 \newcommand\etoc@framedtoci[2][\etocdefaultnbcol]{%
808 \etocframedstyle[#1]{#2}%
809 \tableofcontents*}
810 \newcommand\etoc@local@framedtoc[2][\etocdefaultnbcol]{%
811 \etocframedstyle[#1]{#2}%
812 \localtableofcontents}
813 \newcommand\etoc@local@framedtoci[2][\etocdefaultnbcol]{%
814 \etocframedstyle[#1]{#2}%
815 \localtableofcontents*}
```

placeholder for comments

```
816 \newcommand\etocmulticol{\begingroup
817   \Etoc@mustclosegrouptrue
818   \@ifstar
819   {\etoc@multicoltoctoci}
820   {\etoc@multicoltoctoc}}
821 \newcommand\etocruled{\begingroup
822   \Etoc@mustclosegrouptrue
823   \@ifstar
824   {\etoc@ruledtoctoci}
825   {\etoc@ruledtoctoc}}
826 \newcommand\etocframed{\begingroup
827   \Etoc@mustclosegrouptrue
828   \@ifstar
829   {\etoc@framedtoctoci}
830   {\etoc@framedtoctoc}}
831 \newcommand\etoclocalmulticol{\begingroup
832   \Etoc@mustclosegrouptrue
833   \@ifstar
834   {\etoc@local@multicoltoctoci}
835   {\etoc@local@multicoltoctoc}}
836 \newcommand\etoclocalruled{\begingroup
837   \Etoc@mustclosegrouptrue
838   \@ifstar
839   {\etoc@local@ruledtoctoci}}
```

13. Implementation

```

840    {\etoc@local@ruledtoc}}
841 \newcommand\etoclocalframed{\begingroup
842   \Etoc@mustclosegrouptrue
843   \@ifstar
844   {\etoc@local@framedtoci}
845   {\etoc@local@framedtoc}}

placeholder for comments
846 \newcommand\etocarticlestyle{%
847   \etocsettocstyle
848   {\section *{\contentsname
849     \@mkboth {\MakeUppercase \contentsname}
850     {\MakeUppercase \contentsname}}}
851   {}}
852 \newcommand\etocarticlestylenomarks{%
853   \etocsettocstyle
854   {\section *{\contentsname}}
855   {}}

placeholder for comments
856 \newcommand\etocbookstyle{%
857   \etocsettocstyle
858   {\if@twocolumn \@restonecoltrue \onecolumn \else \@restonecolfalse \fi
859     \chapter *{\contentsname
860       \@mkboth {\MakeUppercase \contentsname}
861       {\MakeUppercase \contentsname}}}
862   {\if@restonecol \twocolumn \fi}}
863 \newcommand\etocbookstylenomarks{%
864   \etocsettocstyle
865   {\if@twocolumn \@restonecoltrue \onecolumn \else \@restonecolfalse \fi
866     \chapter *{\contentsname}}
867   {\if@restonecol \twocolumn \fi}}
868 \let\etocreportstyle\etocbookstyle
869 \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
870 \newcommand\Etoc@addcontentsline{}
871 \newcommand\etocmemoirtotocfmt[2]{%
872   \renewcommand\Etoc@addcontentsline{\addcontentsline {toc}{#1}{#2}}%
873   \renewcommand\etocaftertitlehook{%
874     \ifmem@em@starred@listof
875     \else\phantomsection\aftergroup\Etoc@addcontentsline\fi}}
876 \newcommand\etocmemoirstyle{%
877   \etocsettocstyle
878   {\ensureonecol \par \begingroup \@nameuse {\@tocmaketitle}
879     \Etoc@aftertitlehook\let\Etoc@aftertitlehook\relax

```


13. Implementation

```

880      \parskip \cftparskip \@nameuse {cfttocbeforelisthook}}
881      {\@nameuse {cfttocafterlisthook}\endgroup\restorefromonecol}}
placeholder for comments
882 \newcommand\etocscartclstyle{%
883     \etocsettocstyle
884     {\let\if@dynlist\if@tocleft
885       \iftocfeature {toc}{onecolumn}
886       {\iftocfeature {toc}{leveldown}
887        {}
888        {\if@twocolumn \aftergroup \twocolumn \onecolumn \fi }}
889       {}%
890       \tocbasic@listhead {\listoftocname}%
891       \begingroup \expandafter \expandafter \expandafter
892       \endgroup \expandafter
893       \ifx
894       \csname microtypesetup\endcsname \relax
895       \else
896       \iftocfeature {toc}{noprotrusion}{%
897         {\microtypesetup {protrusion=false}%
898          \PackageInfo {tocbasic}%
899          {character protrusion at toc deactivated}}%
900       \fi
901       \setlength {\parskip }{\z@ }%
902       \setlength {\parindent }{\z@ }%
903       \setlength {\parfillskip }{\z@ \@plus 1fil}%
904       \csname tocbasic@@before@hook\endcsname
905       \csname tb@toc@before@hook\endcsname}
906       {\csname tb@toc@after@hook\endcsname
907        \csname tocbasic@@after@hook\endcsname}}
908 \let\etocscrbookstyle\etocscartclstyle
909 \let\etocscrreprtstyle\etocscartclstyle
placeholder for comments
910 \newcommand\etocnakedtocdisplaystyle{\etocsettocstyle{}}{}
911 \newcommand\etocstandarddisplaystyle{\etocarticlestyle}
912 \newcommand\etocmarkboth[1]{%
913     \@mkboth{\MakeUppercase{#1}}{\MakeUppercase{#1}}}
914 \newcommand\etocmarkbothnouc[1]{\@mkboth{#1}{#1}}
915 \newcommand\etoc tocstyle[3][section]{\etocmulticolstyle[#2]%
916     {\csname #1\endcsname *{#3}}}
917 \newcommand\etoc tocstylewithmarks[4][section]{\etocmulticolstyle[#2]%
918     {\csname #1\endcsname *{#3\etocmarkboth{#4}}}}
919 \newcommand\etoc tocstylewithmarksnouc[4][section]{\etocmulticolstyle[#2]%
920     {\csname #1\endcsname *{#3\etocmarkbothnouc{#4}}}}
placeholder for comments
921 \def\Etoc@redefetocstyle#1{%
922     \renewcommand\etoc tocstylewithmarks[4][#1]
923     {\etocmulticolstyle[##2]%
924      {\csname ##1\endcsname *{##3\etocmarkboth{##4}}}}
925     \renewcommand\etoc tocstylewithmarksnouc[4][#1]
926     {\etocmulticolstyle[##2]%
927      {\csname ##1\endcsname *{##3\etocmarkbothnouc{##4}}}}

```

13. Implementation

```
928 \renewcommand\etocstyle[3][#1]{%
929 \etocmulticolstyle[##2]{\csname ##1\endcsname *{##3}}}%
930 \ifclassloaded{scrartcl}
931 {\renewcommand\etocstandarddisplaystyle{\etocscrartclstyle}}%
932 \ifclassloaded{book}
933 {\renewcommand\etocfontone{\normalfont\normalsize}
934 \renewcommand\etocstandarddisplaystyle{\etocbookstyle}
935 \Etoc@redefetocstyle{chapter}}%
936 \ifclassloaded{report}
937 {\renewcommand\etocfontone{\normalfont\normalsize}
938 \renewcommand\etocstandarddisplaystyle{\etocreportstyle}
939 \Etoc@redefetocstyle{chapter}}%
940 \ifclassloaded{scrbook}
941 {\renewcommand\etocfontone{\normalfont\normalsize}
942 \renewcommand\etocstandarddisplaystyle{\etocscrbookstyle}
943 \Etoc@redefetocstyle{chapter}}%
944 \ifclassloaded{scrreprt}
945 {\renewcommand\etocfontone{\normalfont\normalsize}
946 \renewcommand\etocstandarddisplaystyle{\etocscrreprtstyle}
947 \Etoc@redefetocstyle{chapter}}%
948 \ifclassloaded{memoir}
949 {\renewcommand\etocfontone{\normalfont\normalsize}
950 \etocmemoirtocformat{chapter}{\contentsname}%
951 \renewcommand\etocstandarddisplaystyle{\etocmemoirstyle}
952 \Etoc@redefetocstyle{chapter}}%

placeholder for comments

953 \newcommand\etocstandardlines{\Etoc@standardtrue}
954 \etoclines
955 \etocstandardlines
956 \etocstandarddisplaystyle
957 \endinput
```