

The **etoc** package

JEAN-FRANÇOIS BURNOL

jfbu (at) free (dot) fr

Package version: v1.07f (2013/03/07)

Documentation generated from the source file
with timestamp “06-04-2013 at 13:48:01 CEST”

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. Simpler ones are given as examples in the documentation. The simplest usage will be to take advantage of the layout facilities of packages dealing with list environments.

The `\tableofcontents` command may be used arbitrarily many times and it has a variant `\localtableofcontents` which prints tables of contents ‘local’ to the current surrounding document unit. An extension of the `\label/\ref` syntax allows to reproduce (with another layout) a TOC defined somewhere else in the document.

The formatting inherited (and possibly customized by other packages) from the document class may also be used in *compatibility mode*. Regarding the *global toc display*, **etoc** provides pre-defined styles based on a multi-column format, optionally with a frame or a ruled title.

As the assignment of levels to the sectioning units can be changed at any time, **etoc** can be used in a quite general manner to also create custom “lists of”, additionally to the tables of contents related to the document sectioning units. No auxiliary file is used apart from the usual `.toc` file.

Foreword

Popular packages dealing with TOCs include `tocloft`, `titletoc` and `minitoc`. Why another one? well, initially 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 \LaTeX 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.

etoc is compatible with the `article`, `book`, `report`, `scrartcl`, `scrbook`, `scrreprt` and `memoir` classes.

CONTENTS

Change history

- v1.07f [2013/03/07]
new macros `\etocthelinkedname`, `\etocthelinkednumber`, `\etocthelinkedpage`, and `\etocthelink`.
- v1.07e [2013/03/01]
improvements in the package own line styles with regards to penalties and vertical spaces.
addition to the documentation of an example of a tree-like table of contents (uses `tikz`).
more such examples added 2013/03/03.
- v1.07d [2013/02/24]
minor code improvements and new documentation section "Another compatibility mode".
- v1.07b [2013/02/02]
removal of the `\xspace` from the macros `\etocname`, `\etocnumber`, `\etocpage`.
additional examples in the documentation.
- v1.07 [2013/01/29]
new commands:
`\etocthename`, `\etocthenumber`, `\etocthepage`, `\etoclink`,
`\etoccontentline`, `\etoccontentline*`
`\etocnopar`, `\etocaftercontentshook`
modified command: `\etocmulticolstyle`
- new documentation section "Surprising uses of `etoc`" which explains how to do "Lists of arbitrary things", in addition to the tables of contents.
- v1.06 [2012/12/07]
the standard macros `\@section` etc... are modified only during the calls to `\tableofcontents`; they can thus be customized as will by the user (with the help of a package like `tocloft`) and this will be taken into account by `etoc` for the TOCs typeset in compatibility mode.
- v1.05 [2012/12/01]
`\localtableofcontents` replaces `\tableofcontents*` (for compatibility with the `memoir` class).
compatibility with KOMA-script and `memoir` document classes.
- v1.04 [2012/11/24]
a (possibly local) table of contents can be labeled:
`\tableofcontents \label{toc:1}`
and reproduced elsewhere in the document (with a possibly completely different layout):
`\tableofcontents \ref{toc:1}`
- v1.02 [2012/11/18]
initial version.

TABLE OF CONTENTS

Part I. Overview

Initial motivation: nested lists	1, p. 4
Limitations intrinsic to the use of environments.	
Line styles and toc display style	2, p. 5
<code>\etocsettocstyle</code> for the toc display – <code>\etocsetstyle</code> for the line styles – Compatibility mode.	
Arbitrarily many TOCs, and local ones too	3, p. 6
Labeling and reusing elsewhere – The <code>hyperref</code> option <code>bookmarksdepth</code> – On manually adding layout commands to the <code>.toc</code> file – Shuffling the levels with <code>\etocsetlevel</code> .	
Two examples	4, p. 8
Again the table of contents of this part.	
Surprising uses of <code>etoc</code>	5, p. 10
The TOC of TOCs – Arbitrary "Lists Of..." – A TOC with a fancy layout – Another compatibility mode – The TOC as a tree – The TOC as a molecule.	

CONTENTS

Part II. Package commands for line styles

The `\etocsetstyle` command 6, p. 23

The `\etocname` and `\etocpage` commands—The `\etocskipfirstprefix` command—The `\etocnumber` command—The `\etocthename`, `\etocthenumber`, and `\etocthepage` commands—The `\etoclink` command—The `\etocthelinkedname`, `\etocthelinkednumber`, `\etocthelinkedpage` and `\etocthelink` commands..

The `\etocsetlevel` command 7, p. 26

Scope of commands added to the `.toc` file 8, p. 27

Testing the scope—This is a (pale) red subsection for illustrative purposes.

Am I also red? 9, p. 27

Part III. Package commands for toc display styles

Specifying the toc display style 10, p. 28

The command `\etocruledstyle`—The command `\etocmulticolstyle`—The command `\etocruled`—The commands `\etocframedstyle` and `\etocframed`—Headings, titles, `\etocinnertopsep`—The command `\etocsettocstyle`—The compatibility mode `\etocstandarddisplaystyle`.

Starred variants of the `\tableofcontents` etc... commands 11, p. 31

Table of contents for this part 12, p. 32

Testing the compatibility mode—A framed display—A (crazy) inline display.

Part IV. Using and customizing `etoc`

Summary of the main style commands 13, p. 35

Setting up local styles—Setting up toc display styles—Displaying tables of contents—Labels and references—The package default line styles: `\etocdefaultlines`—One more example TOC layout.

Customizing `etoc` 14, p. 41

Customizing the `etoc` pre-defined line styles—Customizing the toc display styles.

Part V. Tips

... and tricks 15, p. 44

Hacking framed parboxes—Interverting the levels—All subsections of this document—Displaying statistics—Compatibility with other packages—T_EXnical matters—Errors and catastrophes.

Part VI. The code

Implementation 16, p. 49

Part I.

Overview

Here are some statistics for this part: it contains 5 sections and 15 subsections. The name of the first section is [Initial motivation: nested lists](#) and the corresponding number is [1](#). The name of the last section is [Surprising uses of `etoc`](#) and its number is [5](#). The name of the first subsection is [Limitations intrinsic to the use of environments](#) and the corresponding number is [1.1](#). The name of the last subsection is [The TOC as a molecule](#) and its number is [5.6](#).

1. Initial motivation: nested lists

The initial impetus was to feed nested list environments with the data consisting of the *name* (`\etocname`), *number* (`\etocnumber`), and *page number* (`\etocpage`) as recorded¹ in the `.toc` file. 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 (\LaTeX '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 . . . ²). 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 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.

The built-in default “line styles” provided by the package do not make use of environments.

¹the *number* has to be disentangled from the *name*, and in case `hyperref` is present, the hyperlink has to be redistributed around each of them.

²this is surely a well-known issue which I did not at all investigate any further.

2. Line styles and toc display style

A distinction shall be made between the *line styles*, *i.e.* the way the name, number and page numbers are used at each level, and the *toc display style* (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... the latter is governed by the command `\etocsettocstyle` (or some higher-level commands) and the former by the command `\etocsetstyle`.

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. It has two arguments, the first one is executed before the TOC contents (typically it will print “Contents” and define suitable head-marks) and the second is executed after the TOC contents.

`etoc` provides four (customizable) higher level toc styles: `\etocmulticolstyle`, `\etocdisplaystyle`, `\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`.

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*.

2.3. Compatibility mode

Both for the “line styles” and the “toc display style”, it is possible to switch into a compatibility mode which uses the defaults from the document class.³ This is activated by:

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

If the command `\etocsetstyle` has not been used in the preamble the package will be at `\begin{document}` in this compatibility mode: hence just adding `\usepackage{etoc}` should hopefully not change anything to the look of a previously existing document, under the `article`, `book`, `report`, `scrartcl`, `scrbook`, `scrreprt` and `memoir` classes.

³for the “toc display style” `etoc` checks if it knows the class, and if not defaults to the `article` class layout.

3. Arbitrarily many TOCs, and local ones too

Any use of `\etocsetstyle` in the preamble or body of the document turns off the compatibility mode for line styles (but not for the global display style; for this one needs to use the command `\etocsettocstyle`).

To exit after `\etocstandardlines` from compatibility mode one uses the command `\etoclocalines`, which re-activates the latest line styles as defined by `\etocsetstyle` (if their scope was not limited to a group or environment). The command `\etocdefaultlines` resets the line styles to be the package initial default ones.

3. Arbitrarily many TOCs, and local ones too

etoc allows arbitrarily many `\tableofcontents` commands in your document. The line styles and the toc display style may of course be changed in-between. Furthermore `\localtableofcontents` will print local tables of contents⁴: *i.e.* all sections and sub-units inside a given chapter, or all subsubsections and lower inside a given subsection, etc . . .

3.1. 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}`. The actual layout (title inclusive) used for the cloned TOC will be decided locally. The line styles and toc display 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	
13. Summary of the main style commands	35
13.1. Setting up local styles	35
13.2. Setting up toc display styles	35
13.3. Displaying tables of contents	36
13.4. Labels and references	36
13.5. The package default line styles: <code>\etocdefaultlines</code>	36
13.6. One more example TOC layout	39
14. Customizing etoc	41
14.1. Customizing the etoc pre-defined line styles	41
14.2. Customizing the toc display styles	42

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

⁴Up 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

of this TOC⁵ 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, 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}`.

3.2. The `hyperref` option `bookmarksdepth`

When modifying the counter `tocdepth` for the purposes of multiple uses of `\tableofcontents` or `\localtableofcontents`, one should consider that package `hyperref` by default takes into account the *current* value of the `tocdepth` counter to decide whether the final pdf output will contain a bookmark corresponding to the used sectioning command. Thus, one often will have to reset `tocdepth` to its previous value immediately after the display of the table of contents.

Or, there is the `bookmarksdepth=n` option of package `hyperref`, with n the desired document bookmarks depth, which can be numeric or the name of a level known to `hyperref`. The present document passed `bookmarksdepth=3` as option to `hyperref`, so as to not have to reset `tocdepth` each time its value was changed.

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

When displaying in that way many tables of contents in the same document one should of course beware of the impact of adding manually things to the `.toc` file. For example, 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}`, to have a `\let\myclearpage\relax` at the top level of the document and to use where needed something like:

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

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 `\begingroup \renewcommand*\changetocdepth[1]{} \localtableofcontents \endgroup`, and to set the desired level for the local table of contents with the other memoir command `\max-tocdepth`.⁶

⁵*i.e* 4, there was an invisible TOC with id 2 and another one whose identity is left to the reader's sagacity.

⁶The 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.

4. Two examples

3.4. Shuffling the levels with `\etocsetlevel`

The intrinsic levels manipulated by `etoc` are numeric: from -2 (which corresponds to book in the memoir class) down to 5 (subparagraph). But the assignment of a numeric level to a given name can be modified at any time with the command `\etocsetlevel{<level_name>}{<n>}`. In conjunction with the use of the L^AT_EX `tocdepth` counter, this has powerful applications: `<level_name>` does not have to coincide with an actual document sectioning command, and `etoc` can be used to print arbitrary “lists of things”, using no other auxiliary file than the `.toc` file. This is explained further in the section 5.

4. Two examples

Here is a simple example of use of the package functionalities. Immediately after the `\part{Overview}` line in the source file we inserted:

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

The `tocdepth` having been set to -3, nothing at all was typeset, as `etoc` cancels printing even the heading of the TOC if the `tocdepth` is -3 or less (and it is even “-2 or less”, except for the memoir class). We will then display here this TOC with:

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

But first let us set up some line styles. We choose a style for sections and sub-sections which would be suitable for, respectively, sections and sub-sections in an average length memoir. The line style specifications have some redundancy for clarity, and do not care about what to do at possible page breaks. Also, they do not worry about potential multi-column use.

```
\begingroup % we start a group to keep the style changes local
\newlength{\tocleftmargin} \setlength{\tocleftmargin}{5cm}
\newlength{\tocrightmargin} \setlength{\tocrightmargin}{1cm}

\etocsetstyle{section} % will pretend to be a Chapter
{\addvspace{1ex}\parfillskip0pt
 \leftskip\tocleftmargin % (already done in title)
 \rightskip\the\tocrightmargin plus 1fil
 \parindent0pt\color{cyan}} % (already done)
{\bfseries\LARGE\upshape\addvspace{1ex}\leavevmode}
{\llap{Chapter\hspace{.75cm}}{\etocnumber}\hspace{.75cm}}{\etocname}
\hfill\makebox[-\tocrightmargin][l]{\makebox[0pt]{\etocpage}}\par}
{}

\etocsetstyle{subsection} % will pretend to be a Section
{}
{\mdseries\large\addvspace{.5ex}\leavevmode}
{\llap{\etocnumber\hspace{.75cm}}{\textit{\etocname}}%
 \hfill\makebox[-\tocrightmargin][l]{\makebox[0pt]{\etocpage}}\par}
{}
```

4. Two examples

```
\etocsettocstyle{\color{cyan}\parindent0pt \leftskip\tocleftmargin
\leavevmode\leaders\hrule height 1pt\hfill\
\huge\textit{My Beautiful Thesis}\par}{\bigskip}

% \color{cyan}\parindent0pt and \leftskip\tocleftmargin were repeated
% in the <start> code of the ‘‘section style’’, for clarity of code.

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

————— *My Beautiful Thesis*

Chapter 1	Initial motivation: nested lists	4
1.1	<i>Limitations intrinsic to the use of environments</i>	4
Chapter 2	Line styles and toc display style	5
2.1	<i>\etocsettocstyle for the toc display</i>	5
2.2	<i>\etocsetstyle for the line styles</i>	5
2.3	<i>Compatibility mode</i>	5
Chapter 3	Arbitrarily many TOCs, and local ones too	6
3.1	<i>Labeling and reusing elsewhere</i>	6
3.2	<i>The hyperref option bookmarksdepth</i>	7
3.3	<i>On manually adding layout commands to the .toc file</i>	7
3.4	<i>Shuffling the levels with \etocsetlevel</i>	8
Chapter 4	Two examples	8
4.1	<i>Again the table of contents of this part</i>	10
Chapter 5	Surprising uses of etoc	10
5.1	<i>The TOC of TOCs</i>	10
5.2	<i>Arbitrary ‘‘Lists Of...’’</i>	13
5.3	<i>A TOC with a fancy layout</i>	14
5.4	<i>Another compatibility mode</i>	15
5.5	<i>The TOC as a tree</i>	17
5.6	<i>The TOC as a molecule</i>	20

5. Surprising uses of **etoc**

As one can see, the code uses the basic commands from T_EX/L^AT_EX for paragraph layouts and an efficient mix of plain T_EX and L^AT_EX syntaxes.

Users not so knowledgeable in the T_EX syntax (for cause of having been exposed only to L^AT_EX “introductions”) have the possibility explained earlier to use nested `enumerate` environments, of course this means mastering another syntax.

However, using as here the T_EX primitive `\par`,⁷ and basic skip registers `\leftskip`, `\rightskip`, `\parfillskip`, ... is by far the surest way to completely understand and master what happens. Any user of L^AT_EX should learn their signification.

4.1. Again the table of contents of this part

Let us now try out some more sophisticated line styles. The display will use the `\etoc-framedstyle` package command, which requires that the produced table of contents fits on a single page. With all the recent additions to the documentation there is not enough room here, so we wrap it up in a figure environment and it will show [on the next page](#).

The actual design is not pre-built in **etoc**; it uses its ‘framed’ style with a background color. The frame borders have been set to have the same color as the one serving as background for the entire thing. This design (with other colors) is in use also for [this other toc](#), and the reader is referred to [its location](#) for the coding used.

5. Surprising uses of **etoc**

Here are some statistics for this section. It contains 6 subsections. The name of its first is [The TOC of TOCs](#) and the corresponding number is [5.1](#). The name of the last subsection is [The TOC as a molecule](#) and its number is [5.6](#).

5.1. The TOC of TOCs

Here is the numbered and linked list of all tables of contents which are displayed within this document: [1](#), [2](#), [3](#), [4](#), [6](#), [7](#), [5](#), [8](#), [9](#), [10](#), [11](#), [12](#), [13](#), [14](#), [15](#), [16](#), [17](#), [18](#), [19](#), [20](#), [21](#), [22](#), [23](#), [24](#), [25](#). And to obtain it here we just wrote:⁸

```
Here is the numbered and linked list of all tables of contents
which are displayed within this document: \tableofcontents.
```

The preparatory work was the following. First, we defined a counter `visibletoc` whose vocation is to get incremented at each displayed toc. **etoc** has its own private counter but it counts all TOCs, even those not displayed because the `tocdepth` value was `-2` or `-3`.

We could have added manually `\refstepcounter{visibletoc}` and `\label` commands at all suitable locations in the document source, and we would then have used here `\ref` commands, but this imposes heavy manual editing of the source.

There is a much better way: there is a hook `\etocaftertitlehook` and we told it to increment the `visibletoc` counter and to write a line to the `.toc` file, in a manner analogous to what sectioning commands such as `chapter`, `section`, or `subsection` do. As

⁷`\par` is redefined by L^AT_EX but this is of no immediate relevance here.

⁸click the ‘7’ in the list to get confirmation... and click on the previous ‘6’ for a surprise...

The [floating TOC 5](#) is responsible for some numbering madness here: the numbers are listed in the order the TOCs appear in the document; but the numbering itself is from the order of the TOCs in the source of this document...

5. Surprising uses of **etoc**

Contents of Part One		
1	Initial motivation: nested lists	4
1.1	Limitations intrinsic to the use of environments	4
2	Line styles and toc display style	5
2.1	<code>\etocsettocstyle</code> for the toc display	5
2.2	<code>\etocsetstyle</code> for the line styles	5
2.3	Compatibility mode	5
3	Arbitrarily many TOCs, and local ones too	6
3.1	Labeling and reusing elsewhere	6
3.2	The hyperref option <i>bookmarksdepth</i>	7
3.3	On manually adding layout commands to the .toc file	7
3.4	Shuffling the levels with <code>\etocsetlevel</code>	8
4	Two examples	8
4.1	Again the table of contents of this part	10
5	Surprising uses of etoc	10
5.1	The TOC of TOCs	10
5.2	Arbitrary “Lists Of...”	13
5.3	A TOC with a fancy layout	14
5.4	Another compatibility mode	15
5.5	The TOC as a tree	17
5.6	The TOC as a molecule	20

`\tableofcontents \ref{toc:partone}` (cf. subsection 4.1 and [this other toc](#))

etoc increments its own private counter even before typesetting the title of a table of contents, this provides (most of the time) a better link destination than any counter manipulated from inside `\etocaftertitlehook` (for which the link would target the area just after the title). So, rather than including `\refstepcounter{visibletoc}` inside `\etocaftertitlehook`, we just put there `\stepcounter{visibletoc}` followed by the command `\etoccontentsline{visibletoc}{\thevisibletoc}`. This **etoc** command `\etoccontentsline{<level_name>}{<name>}` has the same effect as:

`\addcontentsline{toc}{<level_name>}{<name>}`

but its usefulness is to circumvent⁹ the patching for automatic creation of bookmarks done to `\addcontentsline` by the hyperref package, as pdf bookmarks don’t make much sense here (and would elicit a complaint of hyperref that the bookmark level is ‘un-

⁹using `\addtocontents` rather than `\addcontentsline`

5. Surprising uses of `etoc`

known’).

The package provides a starred variant `\etoccontentsline*`, which will allow the creation of bookmarks and has a third mandatory argument which is the Level to be used by these bookmarks.

Finally, the preamble of the document did `\etocsetlevel{visibletoc}{6}`. The level 6 (or anything with a higher number) is ignored, even if `tocdepth` has value 10 for example; this is independently of whether `etoc` uses the document class default line styles or its own line styles, or the ones defined by the user with the `\etocsetstyle` command. So there is no need to worry that something could go wrong.

Then, only here we have set `\etocsetlevel{visibletoc}{0}`. And to display only this kind of entries we assign temporarily to part and chapter level 1 (or anything higher than zero) and set `tocdepth` to the value 0. We also did `\etocsetstyle{visibletoc}{\etocskipfirstprefix}{, }{\etocname}{}` which defines an inline display with the comma as separator. Finally, as `etoc` issues `\par` automatically by default just before typesetting a table of contents, we used the command `\etocinline` (also known as `\etocnopar`) which turns off this behavior.

Here are the implementation details:

```
. . . in the preamble:
\newcounter{visibletoc}
\renewcommand{\etocaftertitlehook}
{
  \stepcounter{visibletoc}\etoccontentsline{visibletoc}{\thevisibletoc}
\etocsetlevel{visibletoc}{6}
. . .
\begin{document}
. . . document body
\subsection{Surprising uses of etoc}
\begin{group}
  \etocinline
  \etocsetlevel{part}{1}
  \etocsetlevel{chapter}{1}
  \etocsetlevel{visibletoc}{0}
  \etocsetstyle{visibletoc}
    {\etocskipfirstprefix}{, }{\color{niceone}\etocname}{}
  \etocsettocstyle{}{} % don't set any title, rules or frame or multicol!
  \setcounter{tocdepth}{0} % display only the 'visibletoc' entries from .toc
\end{group}
```

Here is the numbered and linked list of all tables of contents which are displayed within this document: `\tableofcontents`.

```
\endgroup
```

After `\etocsetstyle{visibletoc}{...}{...}{...}{...}`, all future TOCs (not in compatibility mode) will use the defined style for level 0 (which is normally the level for chapters). To keep these changes strictly local the simplest manner is to put everything inside a group.

Subsection 15.2 gives another use of the shuffling of levels.

5.2. Arbitrary “Lists Of...”

This idea of interverting the levels is very powerful and allows to let **etoc** display lists of arbitrary things contained in the document. All of that still using nothing else than the .toc file! Example: imagine a document with dozens of exercises, perhaps defined as `\newtheorem{exerci}{}[section]`. Let us explain how to instruct **etoc** to display an hyperlinked list of all these exercises. For this we put in the preamble:

```
\newtheorem{exerci}{}[section]
% the exercise number will be recoverable via \etocname: v--here--v
\newcommand*{\exercisetotoc}{\etoccontentsline{exercise}{\theexerci}}
\newenvironment{exercise}{\begin{exerci}\exercisetotoc}{\end{exerci}}
\etocsetlevel{exercise}{6}
```

In this way, `\etocname` will give the exercise number (but `\etocnumber` will be empty). Had we used instead

```
\newcommand*{\exercisetotoc}
{\etoccontentsline{exercise}{\protect\numberline{\theexerci}}}
```

the exercise number would then have been available via `\etocnumber`, and `\etocname` would have been empty. It doesn't matter which one of the two methods is used. The **etoc** command `\etoccontentsline{...}{...}` is provided as a substitute to `\addcontentsline{toc}{...}{...}`: this is to avoid the patching which is done by `hyperref` to `\addcontentsline` in its process of creation of bookmarks. If one wants to authorize `hyperref` to create bookmarks at a specific level $\langle n \rangle$, one can use (here with $\langle n \rangle = 2$) the starred variant `\etoccontentsline*` which has an additional argument:

```
\newcommand{\exercisetotoc}{\etoccontentsline*{exercise}{\theexerci}{2}}
```

This example originates with question 94766 on the T_EX-StackExchange site. The counter `exerci` is already incremented by the `exerci` theorem environment, and provides the correct destination for the link added by package `hyperref`. The command `\exercisetotoc` adds for each exercise a line to the .toc file, corresponding to a fictitious document unit with name 'exercise'. A four-column list, including the sections, can then be typeset with the following code:

```
\setcounter{tocdepth}{2} % sections are at level 1 and will show up
\begingroup
\etocsetlevel{exercise}{2} % but:
\etocsetlevel{chapter}{3} % no chapters
\etocsetlevel{subsection}{3} % no subsections
\etocsetlevel{part}{3} % no parts
\etocsetstyle{exercise}{}{} % \etocname = exercise number
{\noindent\etocname\strut\leaders\etoclineleaders\hfill\etocpage\par}
{\pagebreak[2]\vskip\baselineskip}
\etocsetstyle{section}{}{}
{\noindent\strut{\bfseries\large\etocnumber\hskip.5em\etocname}\par}
\nopagebreak[3]}
\etocruledstyle[4]{\Large\bfseries List of the exercises}
\setlength{\columnseprule}{.4pt}
\tableofcontents
\endgroup
```

5. Surprising uses of *etoc*

In the above, recall that \LaTeX counters are global. The current `tocdepth` value is 2, and if not reset it will prevent `hyperref` to assign bookmarks to sub-subsections (level 3). The global `hyperref` option `bookmarksdepth` can be used to avoid having to systematically reset `tocdepth` after having changed it.

5.3. A TOC with a fancy layout

Another question (numbered 61297) on the \TeX StackExchange site was about using \LaTeX to obtain a table of contents where the sections from a given chapter would be represented by a number range (like 18–22 for a given chapter, 42–49 for another one ... of course to be inserted automatically in the TOC). Here is the result of my effort at using *etoc* for this specific problem. How this was done will be found on the above cited site.

TABLE OF CONTENTS

PART I

				SECTIONS.	PAGE.
Introductory	1—8	5
Concord.					
LESSON.					
1.	Concord of Subject and Verb	9—17	7
2.	Concord of Substantive and Adjective		...	18—22	9
	Concord of Relative and its Antecedent		...	23—25	

PART II

Government.

3.	The Accusative Case	17
	General uses	26—30	
	Particular uses	31—37	
4.	Verbs governing two Accusatives		...	38—41	23
5.	The Causal	42—49	25
6.	The Instrumental Case	27
	General uses	50—54	
	Particular uses	55—59	
7.	The Dative Case	33
	General uses	60—65	
	Particular uses	66—71	
8.	The Ablative Case	39
	General uses	72—75	
	Particular uses	76—86	
9.	The Locative Case	45
	General uses	87—92	

5. Surprising uses of *etoc*

Particular uses 93—100

This is not an image inclusion, the TOC is produced from its original `tex` source inserted in this document after replacement of `part`, `chapter` or `section` with `dummypart`, `dummychapter` and `dummysection` (and there is also a dummy page count). We copied the line styles used in the original and displayed the table of contents following:

```
\etocsetlevel{dummypart}{-1} \etocsetlevel{part}{2}
\etocsetlevel{dummychapter}{0} \etocsetlevel{chapter}{2}
\etocsetlevel{dummysection}{1} \etocsetlevel{section}{2}
\setcounter{tocdepth}{1}
```

Each chapter displays the numbers of only the first and the last sections it contains. A technique for doing this is explained in the subsection 15.4.

5.4. Another compatibility mode

As explained in the section 2.3, the commands `\etocstandardlines` and `\etocstandarddisplaystyle` tell *etoc* to, essentially, act as an observer. The document class layout for the table of contents is then perfectly obeyed. There is no way to customize this standard layout (change fonts, margins, vertical spacings, etc...) from within the package. For this, use some package dedicated to this task; because *etoc* either is (temporarily perhaps) in compatibility mode with no customization on its part possible, or the user has specified the layout in `\etocsetstyle` commands (and `\etocsettocstyle`) and is (supposedly...) in complete control.

Well, there is actually an alternative. It is possible to use the `\etocsetstyle` commands to recreate an artificial compatibility mode, in order to achieve effects like the following, all things being otherwise equal to the document class defaults:

1. get the `hyperref` link to encapsulate only the names, but not the numbers of each entry of the table of contents,
2. use the document class style for chapters and sections, but modify it only for subsections,
3. do either of the above only for some portions of the table of contents.

Here is how to proceed. One puts in the preamble:

```
\makeatletter
\newcommand{\MyStandardTOC}{%
  \begingroup
  \let\savedpartline\l@part
  \let\savedchapterline\l@chapter %% remove if article/scrartcl class
  \let\savedsectionline\l@section
  \let\savedsubsectionline\l@subsection
  % and so on if \subsubsection, etc... is used
  %
  % for the book or article classes:
  \etocsetstyle{part}{}{}
  {\savedpartline{\etocnumber\hspace{1em}\etocname}{\etocpage}}{}%
  % for the scrbook or scrartcl classes:
  \etocsetstyle{part}{}{}
  {\savedpartline{\numberline{\etocnumber}\etocname}{\etocpage}}{}%
```

Contents

```
% identical in book/article/scrbook/scrartcl classes:
\etocsetstyle{chapter}{}{} %% only for book and scrbook
  {\savedchapterline{\numberline{\etocnumber}\etocname}{\etocpage}}{}%
\etocsetstyle{section}{}{}
  {\savedsectionline{\numberline{\etocnumber}\etocname}{\etocpage}}{}%
\etocsetstyle{subsection}{}{}
  {\savedsubsectionline{\numberline{\etocnumber}\etocname}{\etocpage}}{}%
% etc... if further sectioning units are used
% (see the text for what to do with the memoir class)
\etocstandarddisplaystyle % this is for the title, page-marks, etc...
\tableofcontents
\endgroup}
\makeatother
```

Of course if the document has only one table of contents then there is no need to put the commands inside a macro, or even inside a group.¹⁰ With these commands **etoc** will construct a TOC completely identical to what would have been done by one of the document class: `article`, `book`, `scrartcl`, `scrbook`.¹¹ The number and the name of each entry are each separately an `hyperref` link, as is always the case with **etoc**, when not in compatibility mode. Replacing `\etocnumber` with `\etocthenumber` will give a TOC where the numbers are not links anymore, but the names still are. Or one may decide to use `\etocthename` and keep an hyperlinked number with `\etocnumber`.

Here is a subtler example where one only marginally modifies the sections (adding color to the number and removing the `hyperref` link) and keeps the subsections as in the default, *except* for those of one specific section, for which the layout is completely modified:

Contents

1. Initial motivation: nested lists	4
1.1. Limitations intrinsic to the use of environments	4
2. Line styles and toc display style	5
2.1. <code>\etocsettocstyle</code> for the toc display	5
2.2. <code>\etocsetstyle</code> for the line styles	5
2.3. Compatibility mode	5
3. Arbitrarily many TOCs, and local ones too	6
<i>Labeling and reusing elsewhere (3.1)–The <code>hyperref</code> option <code>bookmarksdepth</code> (3.2)–On manually adding layout commands to the <code>.toc</code> file (3.3)–Shuffling the levels with <code>\etocsetlevel</code> (3.4).</i>	

¹⁰and if moreover one just wants to keep the same layout as in the default, one may question why using `etoc...` there is *one* good reason: numbers and names are separately hyperref links, whereas normally there is only one link holding both the number and the name corresponding to one toc entry.

¹¹For the memoir class, one needs a bit more: each of the command `\booknumberline`, `\partnumberline` and `\chapternumberline` will have to be saved with a `\let`, and, one then specifies:

```
\etocsetstyle{chapter}{}{}{\savedchapterline\savedchapternumberline  
{\etocnumber}\etocname}{\etocpage}}}
```

(and analogously for part, respectively book).

4. Two examples	8
4.1. Again the table of contents of this part	10
5. Surprising uses of <code>etoc</code>	10
5.1. The TOC of TOCs	10
5.2. Arbitrary “Lists Of...”	13
5.3. A TOC with a fancy layout	14
5.4. Another compatibility mode	15
5.5. The TOC as a tree	17
5.6. The TOC as a molecule	20

This example only has sections and subsections, and the code used in `\MyStandardTOC` was:

```
\etocsetstyle{section}{%
  {\ifnum\etocthenumber=3
    \etocsetstyle{subsection}
      {\par\nopagebreak\begin{group}
        \leftskip1.5em \rightskip\@tocrmarg \parfillskip\@flushglue
        \parindent 0pt \normalfont\normalsize\rmfamily\itshape
        % \columnseplem
        % \begin{minipage}{\dimexpr\linewidth-\leftskip-\rightskip\relax}%
        % \begin{multicols}{2}%
        \etocskipfirstprefix}
      {\allowbreak\,--\,}
      {\etocname\ \textup{(\etocnumber)}}
      {\par\endgroup}%
      % {\par\end{multicols}\end{minipage}\par\endgroup}%
    \else
      \etocsetstyle{subsection}
        {}{}
        {\savedsubsectionline{\numberline{\etocnumber}\etocname}{\etocpage}}
        {}%
    \fi}
  {\savedsectionline{\numberline{{\color{cyan}\etocthenumber}}\etocname}{\etocpage}}
  {}%
```

Notice the page head-mark added by this standard TOC. Sections and subsections are printed exactly as in the default (except for the subsections of one specific user-chosen section and except for the color of the section numbers), with no need to specify explicitly any length, font or other formatting instructions. But we had to examine the `scrartcl` sources to determine what to use for `\leftskip` and `\rightskip` for our customized subsection entries.

Also, a fancier layout has been commented out.

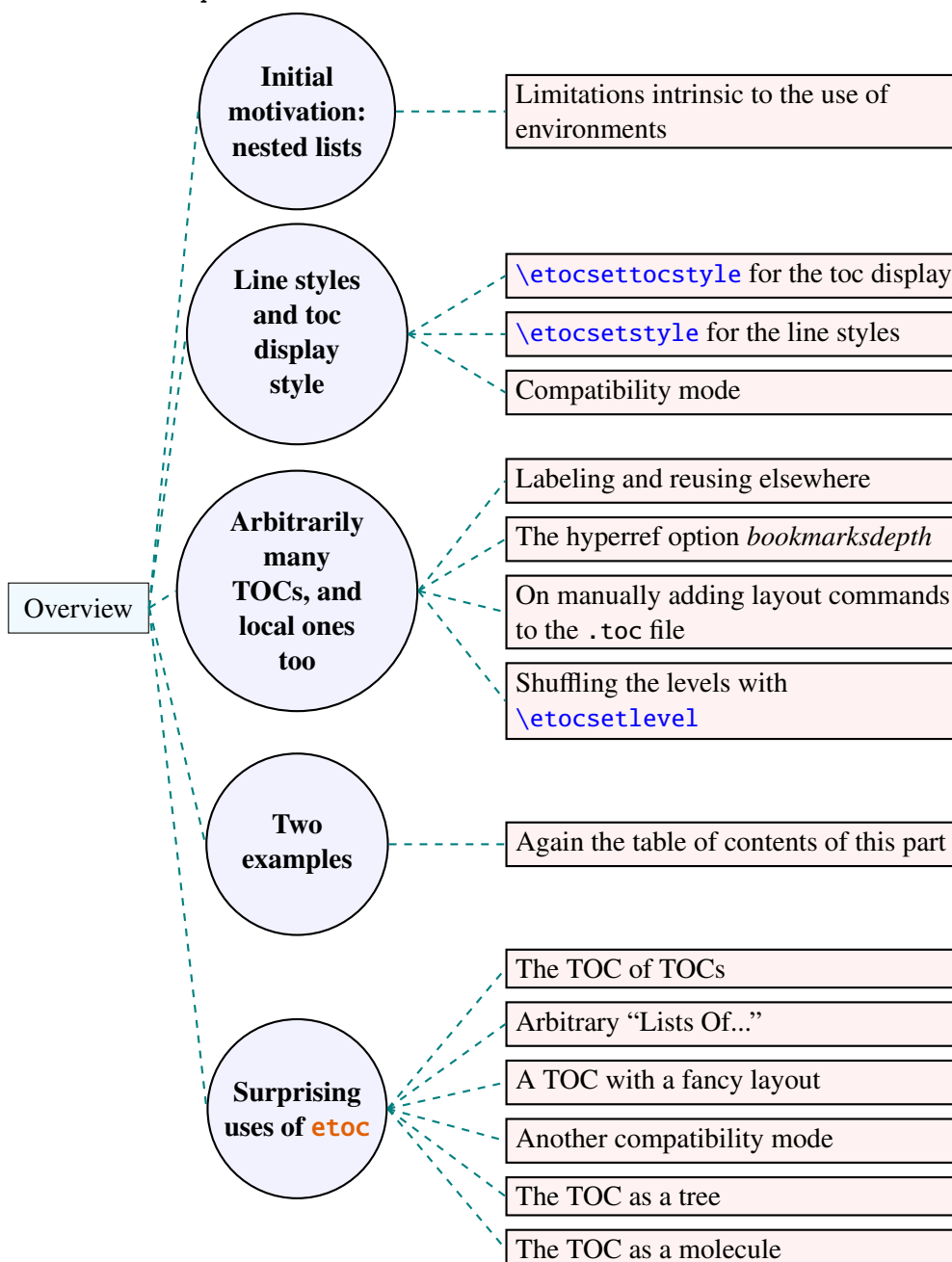
5.5. The TOC as a tree

Using `tikz`¹² and the package `tikz-qtree`¹³ we shall display the table of contents of this part as a tree. The technique (this whole subsection perhaps should have double dangerous-bend signs) is to use the `etoc` modified command `\tableofcontents` not for typesetting,

¹²<http://ctan.org/pkg/pgf>

¹³<http://ctan.org/pkg/tikz-qtree>

but to prepare a macro, or rather here a token list with name `\treetok`, with all the instructions to be executed later. Putting `\etocnumber` or `\etocname` commands in `\treetok` would be of no use: to which number or name would they refer to, in such a delayed execution? Rather the *contents* of `\etocthenumber` or `\etocthename` are added with suitable decoration to `\treetok`: the **etoc** line styles are modified to expand once `\etocthename` (for example) at the time of the execution and then add the outcome to `\treetok` for later execution in a `tikzpicture`.



We told **etoc** to insert the `hyperref` links inside the argument to the `\Tree` command

Contents

(from package `tikz-qtree`).¹⁴ But here in our use of `tikz-qtree` we have been confronted with the problem that the hyperlinks, even if correctly added to the `\treetok` token list, appear in the final document in wrong locations (except for the bottom section and the bottom subsection nodes).¹⁵

Here is the code used, the whole thing being with double dangerous-bend signs, I shall not comment much the actual \LaTeX programming used therein:

```
\newtoks\treetok \newtoks\temptok \newcommand*\qtreenode{}
\newcommand*\appendtotok[2]{% #1=token list, #2=macro, expands once #2
  #1\expandafter\expandafter\expandafter
  {\expandafter\the\expandafter #1#2}}

\newcommand*\PrepareSectionNode{%
  \temptok {\centering\bfseries}%
  \appendtotok\temptok\etocthelinkedname
  \edef\qtreenode{ [. {\noexpand\parbox{2cm}{\the\temptok}}]}%
}
\newcommand*\PrepareSubsectionNode{%
  \temptok {\raggedright}%
  \appendtotok\temptok\etocthelinkedname
  \edef\qtreenode{ [. {\noexpand\parbox{6cm}{\the\temptok}}]}%
}
%% only sections and subsections:
\etocsetstyle{section}
  {\etocskipfirstprefix}
  {\appendtotok\treetok{ }}
  {\PrepareSectionNode \appendtotok\treetok\qtreenode}
  {\appendtotok\treetok{ }}
\etocsetstyle{subsection}
  {\etocskipfirstprefix}
  {\appendtotok\treetok{ }}
  {\PrepareSubsectionNode \appendtotok\treetok\qtreenode}
  {\appendtotok\treetok{ }}

\etocsettocstyle
  {\treetok{\Tree [.Overview]}
  {\global\appendtotok\treetok{ }}}
% See the tikz-qtree (David Chiang) documentation for the Qtree syntax of
% Alexis Dimitriadis (the spaces at various places before the square
% brackets are important).

% The whole effect of \tableofcontents will be to fill the token list
% \treetok according to the syntax expected by tikz-qtree. The opening
% portion of \etocsettocstyle initializes \treetok, and the closing portion
% adds a final square bracket. A \global is needed as etoc always creates a
% group when typesetting a TOC.

% This QTree syntax has the advantage to use square brackets and not braces,
% which eases things quite a bit in a TeX context. The next section explains
% how to create a token list with the original TikZ syntax for trees.
```

¹⁴this uses the macro `\etocthelinkedname` introduced with package version 1.07f. The same effect was achieved in earlier versions in a more complicated manner.

¹⁵`dvipdfmx` complains: **** WARNING **** Annotation out of page boundary. Current page's MediaBox: [0 0 595.276 841.89] Annotation: [94.3997 1200.33 146.512 1209.98]. I read on the internet that there is some problem with `pgf` with regards to hyperlinks, but I don't know if this is the same issue (as this is my first ever use of `TikZ`). [2013/03/01]
Update [2013/03/03]: it *does* work when using the regular `TikZ` syntax for trees, so perhaps there is here some interaction with `tikz-qtree`. See next section.

```
\tableofcontents \ref{toc:partone}
% Time now to display the tree (see tikz-qtree documentation)
\begin{tikzpicture}[grow'=right]
\tikzset{sibling distance=1ex,
  level 1/.style={level distance=4cm},
  level 2/.style={level distance=6cm},
  every level 0 node/.style={draw,fill=cyan!5,inner sep=6pt},
  every level 1 node/.style={circle,draw,thick,fill=blue!5},
  every level 2 node/.style={draw,thick,fill=red!5},
  edge from parent/.style={%
    draw, thick, color=teal,
    edge from parent path=
      {[dashed](\tikzparentnode.east) -- (\tikzchildnode.west)}}
}
\the\treetok
\end{tikzpicture}
```

5.6. The TOC as a molecule

It is also possible to construct a TOC tree obeying the TikZ syntax for trees: this is a more complicated task for the **etoc** line styles for reasons related to the importance of braces in \TeX , they need, when filling up the token list to be always balanced at each step (unfortunately LESLIE LAMPORT's book has no mention whatsoever of token lists, and \LaTeX does not really expect users to know about them; this whole section is thus for advanced users).

The simplest strategy is to allocate a token list (or use a macro) for each level used: we may need a `\parttok`, a `\chaptertok`, a `\sectiontok` and a `\subsectiontok`, to help in the task of filling up the total `\treetok`. As we are interested here in the table of contents of this (or another) document part, only a `\sectiontok` and a `\subsectiontok` will be needed.

And the nice thing is that now, our hyperlinks do work. As mentioned already we can not use the commands `\etocname`, etc..., for delayed execution. We can on the other hand store the current values of `\etocthename`, etc..., but don't get then the `hyperref` links. Earlier versions of this documentation explained how to do things with `\hyperlink` and `\Hy@tocdestname`, but since 1.07f, the package provides `\etocthelinkedname`, etc..., specifically for this kind of task.

```
% \newtoks\treetok
\newtoks\sectiontok \newtoks\subsectiontok \newcommand*{\treenode}{}

\newcommand*{\appendchildtree}[2]{% token list t1 becomes: t1 child {t2}
  \edef\tmp{\the#1 child {\the#2}}%
  #1\expandafter{\tmp}%
}
\newcommand*{\preparetreenode}{%
  \tmptok\expandafter{\etocthelinkednumber}% expanded one time (mandatory)
  \edef\treenode{node {\the\tmptok}}%
}

\setcounter{tocdepth}{2}
\etocsetstyle{section}
{
  \etocskipfirstprefix
  {\appendchildtree\treetok\sectiontok}
  {\preparetreenode \sectiontok\expandafter{\treenode}}
  {\appendchildtree\treetok\sectiontok}
}
```

Contents

```
\etocsetstyle{subsection}
  {\etocskipfirstprefix}
  {\appendchildtree\sectiontok\subsectiontok}
  {\preparetreenode \subsectiontok\expandafter{\treenode}}
  {\appendchildtree\sectiontok\subsectiontok}

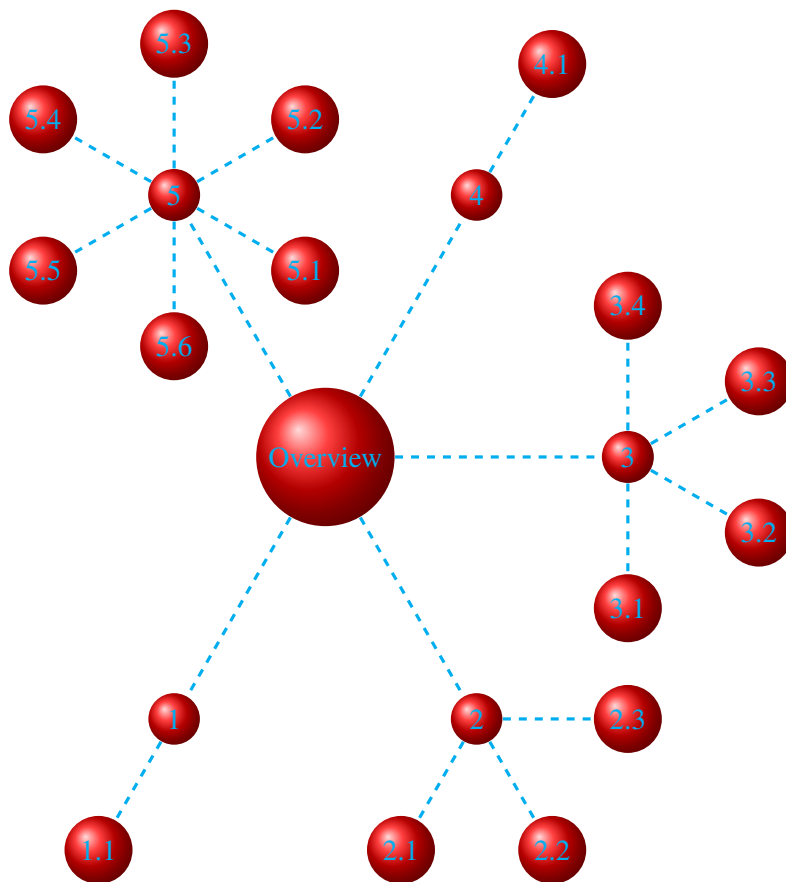
\etocsettocstyle
  {\treetok{\node {\hyperref[part:one]{Overview}}}}
  {\global\appendtotok\treetok{ ;}}

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

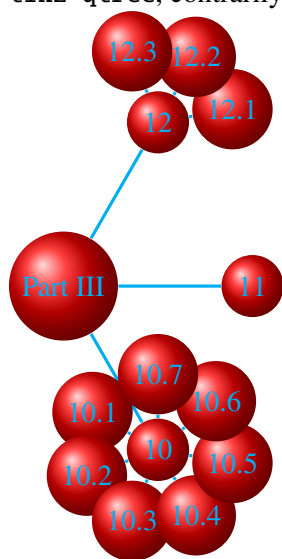
At this stage, `\treetok` has been prepared, and a TikZ picture can be created (it uses the TikZ library `trees` for its “cyclic” grow method):

```
\begin{center}
  \begin{tikzpicture}
    [grow cyclic,
     level 1/.style={level distance=4cm,sibling angle=60},
     level 2/.style={level distance=2cm,sibling angle=60},
     every node/.style={ball color=red,circle,text=cyan},
     edge from parent path={[,dashed,very thick,color=cyan]
                           (\tikzparentnode) --(\tikzchildnode)}]

    \the\treetok
  \end{tikzpicture}
\end{center}
```



This is nice, and especially so as this “molecule TOC” is fully hyperlinked! I don’t know why it goes wrong for the trees constructed with the `\Tree` command of package `tikz-qtree`, contrarily to those coded as here with the original “child” TikZ syntax.



On the side, the (fully hyperlinked) table of contents of Part III.

```
\etocsettocstyle
{\treetok{\node {\autoref{part:globalcmds}}}}
{\global\appendtotok\treetok{ ;}}
\tableofcontents \ref{toc:globalcmds}
\noindent
\parbox{4cm}{%
\begin{tikzpicture}
[grow cyclic,
level 1/.style={level distance=2.5cm,sibling angle=60},
level 2/.style={level distance=1cm,sibling angle=50},
every node/.style={ball color=red,circle,text=cyan},
edge from parent path={[,very thick,color=cyan]
(\tikzparentnode) --(\tikzchildnode)}]
\the\treetok
\end{tikzpicture}}%
...
```

Part II.

Package commands for line styles

Here are some statistics for this part: it contains 4 sections and 8 subsections. The name of the first section is [The `\etocsetstyle` command](#) and the corresponding number is [6](#). The name of the last section is [Am I also red?](#) and its number is [9](#). The name of the first subsection is [The `\etocname` and `\etocpage` commands](#) and the corresponding number is [6.1](#). The name of the last subsection is [This is a \(pale\) red subsection for illustrative purposes](#) and its number is [8.2](#).

Contents of this second part

6 The `\etocsetstyle` command (page 23)

6.1 The `\etocname` and `\etocpage` commands (p. 23)

6.2 The `\etocskipfirstprefix` command (p. 24)

6.3 The `\etocnumber` command (p. 24)

The `\etocifnumbered` switch.

6.4 The `\etocthenname`, `\etocthenumber`, and `\etocthepage` commands (p. 25)

6.5 The `\etoclink` command (p. 25)

6.6 The `\etocthelinkedname`, `\etocthelinkednumber`, `\etocthelinkedpage` and `\etoc-thelink` commands. (p. 25)

7 The `\etocsetlevel` command (page 26)

8 Scope of commands added to the `.toc` file (page 27)

8.1 Testing the scope (p. 27)

8.2 This is a (pale) red subsection for illustrative purposes (p. 27)

9 Am I also red? (page 27)

6. The `\etocsetstyle` command

6.1. The `\etocname` and `\etocpage` commands

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:¹⁶

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

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

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

These provisory style definitions rely on the automatic numbering generated by the `enumerate` environments but it is much better to use the further command `\etocnumber` inside the item label, which gives the real thing. The improved definitions will thus be explained later.

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*.

¹⁶the 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.

6. The `\etocsetstyle` command

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.

The (robust) commands `\etocname`, `\etocnumber` and `\etocpage` are provided for use inside the `{<prefix>}` and `{<contents>}` parts of the `\etocsetstyle` specification. They represent of course, the name, number, and page number of the corresponding toc entry.¹⁷ If package `hyperref` is active in the document and has added hyperlinks to the TOC data, then these links are kept in the commands `\etocname`, `\etocnumber` and `\etocpage` (this last one will have a link only if `hyperref` was passed option `linktoc=all`.)

6.2. The `\etocskipfirstprefix` command

The chosen subsubsection style also uses the command `\etocskipfirstprefix`, which, if present, *must* be the very last one in the `start` code. It instructs to not use for the first item the specified “prefix” code.

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!

6.3. The `\etocnumber` command

So far, our specifications would use the numbering generated by the `enumerate` environments, but of course we generally want 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}`. Note that `\etocnumber` is a robust command which explains why it can be used inside the label specification without needing an added `\protect`.

6.3.1. 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).¹⁸

¹⁷ up to version 1.07a the package put an `\xspace` in each of `\etocname`, `\etocnumber`, and `\etocpage`, but this wasn't such a great idea and has now been removed.

¹⁸ as seen 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`.

6. The `\etocsetstyle` command

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\colorbox{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}`.

6.4. The `\etocthenname`, `\etocthenumber`, and `\etocthepage` commands

It is sometimes desirable to have access to the name, number and page number without the `hyperref` link data: something similar to the starred variant of the `\ref` command, when package `hyperref` is used. For example one may wish to use the unit or page number in some kind of numeric context, or change its formatting. This is provided by the `\etocthe...` commands.

These commands are not protected, so in moving argument contexts (for example in a label specification) they should be preceded by `\protect`.

6.5. The `\etoclink` command

The command `\etoclink{<linkname>}` can be used in the line style specifications in a manner analogous to the argument-less commands `\etocname`, `\etocnumber` and `\etocpage`. It creates a link (if such a link was added by `hyperref` to the `.toc` file entry) whose destination is the corresponding document unit and whose name is the given argument. Hence `\etoclink{\etocthename}` is like the original `\etocname`. Notice that if `hyperref` was not instructed to put a link in the page number (via its option `linktoc=all`) then `etoc`’s `\etocpage` contains no link either, but `\etoclink{\etocthepage}` does.

The command `\etoclink` is robust.

6.6. The `\etocthelinkedname`, `\etocthelinkednumber`, `\etocthelinkedpage` and `\etocthelink` commands.

This is for advanced uses by advanced users.

The `\etocthename` macro has been mentioned before; using it in instructions such as `\global\let\lastone\etocthename` in an `etoc` line style will define the macro `\lastone` to expand to the last name seen at the corresponding level. But no facilities was previously available to do the same with the link data.

7. The `\etocsetlevel` command

The package now provides `\etocthelinkedname` to do the similar thing, with the link data included. It was used in this documentation when doing the examples with trees.

Also provided with the similar goal: `\etocthelinkednumber`, `\etocthelinkedpage` (which contains a link only if `hyperref` added one to the page number) and `\etocthelink{<linkname>}` which allows to make a link with an arbitrary name.

All these commands are fragile.

7. The `\etocsetlevel` command

As already explained in the section 5, one can inform **etoc** of a level to associate to a given sectioning command with `\etocsetlevel`. For example:

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

Of course, in compatibility mode, it will be assumed here that the macros `\l@cell`, `\l@molecule`, ..., pre-exist. If no table of contents is typeset in compatibility mode, then all that matters is that the line styles have been set. If for example section is at level 1, then there is no need to do a `\etocsetstyle{molecule}` if `\etocsetstyle{section}` has already been done (and it has been done by the package itself in its definition of its own line styles).

The accepted levels run from -2 to 6 inclusive. Anything else is mapped to 6, which is a dummy level, never displayed. The package 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}
```

etoc own custom styles are activated by `\etocdefaultlines`. They are illustrated by the main table of contents of this document.

These level assignments can be modified at anytime: see the section 5 for various applications of this technique. As one further example, let's mention here that the main table of contents of this document was typeset following these instructions:

```
\setcounter{tocdepth}{3}  
\etocdefaultlines % use the package default line styles. At this early stage in  
                  % the document they had not yet been modified by \etocsetstyle  
                  % commands, so \etocclines could have been used, too.  
  
\etocmarkboth\contentsname  
\etocmulticolstyle[1] % one-column display  
  {\noindent\bfseries\Large  
   \leaders\hrule height1pt\hfill  
   \MakeUppercase{Table of Contents}}  
\begingroup % use a group to limit the scope of the
```

8. Scope of commands added to the .toc file

```
\etocsetlevel{subsection}{3}      % subsection level change.  
\etocsetlevel{subsubsection}{4}    % anything > tocdepth=3.  
\tableofcontents \label{toc:main}  
\endgroup
```

In this way, the subsections used the style originally designed for subsubsections, the subsubsections were not printed. Without this modification, the appearance would have been very different: the package line styles were targeted to be employed in documents with many many sub-sub-sections, in a two-column layout, giving thus a more compact output that what is achieved by the default L^AT_EX table of contents. But here, we have few sub-sub-sections and it is more interesting to drop them and print in a visually different manner sections and subsections.

8. Scope of commands added to the .toc file

8.1. Testing the scope

Let us switch to the color red, and also add this command to the .toc file:

```
\color{red!50}                    % changing text color  
\addtocontents{toc}{\string\color{red!50}} % and also in the .toc file
```

8.2. This is a (pale) red subsection for illustrative purposes

Actually, this title here was printed black, due to the way the `scrartcl` class works (it would have been red in the `article` class), but we are more interested in how it looks in the tables of contents: it does appear red in the [main table of contents](#) at the beginning of this document, and also in the [table of contents for this part](#). Both entries obey as expected the `\color{red!50}` command inserted in the .toc file.

But let us now close this subsection and start a section.

9. Am I also red?

The question is about how it appears in the tables of contents: the answer is that, yes it is red in the [main TOC](#), and no it is not red in the [local TOC for this part](#). The reason is that the `<finish>` code for the subsection level closed a group, as it used `\end{enumerate}`.

This illustrates the discussion from subsection 1.1.

The default package line styles do not contain group opening and closing instructions: the influence of a command added to the .toc file will propagate until cancelled by another explicit such command inserted in the .toc file.

```
\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.

Part III.

Package commands for toc display styles

Here are some statistics for this part: it contains 3 sections and 10 subsections. The name of the first section is [Specifying the toc display style](#) and the corresponding number is [10](#). The name of the last section is [Table of contents for this part](#) and its number is [12](#). The name of the first subsection is [The command `\etocruledstyle`](#) and the corresponding number is [10.1](#). The name of the last subsection is [A \(crazy\) inline display](#) and its number is [12.3](#).

```
\setcounter{tocdepth}{-3}
\localtableofcontents \label{toc:globalcmds}
```

10. 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})
```

10.1. 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 `\fboxes` to get frames around them. For the example with the standard formatting we did not use an `\fbox` and got rid of the horizontal rules via:

```
\renewcommand{\etocoprulr}{\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}%
\fbxrule1pt\fbxseplex
\framebox[\linewidth]
{\normalcolor\hss Contents of this second part\hss}}
```

10.2. 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⟩}`

[New with 1.07] An explicit `\par` not being accepted in the `⟨heading⟩` argument (this is actually a restriction originating in the `multicols` environment), an implicit one is automatically added by **etoc** at the end of the argument, as in this example which shows how the main table of contents of this document was configured:

```
\etocmulticolstyle{\noindent\bfseries\Large
\leaders\hrule height1pt\hfill
\MakeUppercase{Table of Contents}}
```

After `\etocmulticolstyle` all future `\tableofcontents` will use the specified style, if not changed in-between. 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⟩}`.

10.2.1. The command `\etoc tocstyle`

`\etoc tocstyle[⟨kind⟩]{⟨number_of_columns⟩}{⟨title⟩}`
`= \etocmulticolstyle[⟨number_of_columns⟩]{\kind*{⟨title⟩}}`

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

10.2.1.1. `\etoc tocstylewithmarks[⟨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 `\etoc tocstylewithmarksnouc` which does not uppercase.

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

10.2.1.3. really?

10.3. The command `\etocruled`

As a shortcut to set the style with `\etocruledstyle` and then 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`.

10.4. 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`.

10.5. Headings, titles, `\etocinnertopsep`

[*modified in v1.07*] There is a slight difference between `\etocmulticolstyle` and `\etocruledstyle` or `\etocframedstyle`. For `\etocmulticolstyle` the mandatory *⟨heading⟩* argument can be something like: `\section*{Table of Contents}`. On the contrary `\etocruledstyle` and `\etocframedstyle` expect an argument “in LR mode” (to use the terminology from the *LaTeX, a document preparation system*). This means that multiline contents arguments to `\etocruledstyle` or `\etocframedstyle` must be enclosed in something like a `\parbox`.

[*new in v1.07*] The command `\etocmulticolstyle` now also accepts horizontal mode material in its mandatory argument *⟨heading⟩*: it internally automatically adds a closing `\par`. So one can use for example `\etocmulticolstyle{Hello World}`. Speaking of `\par`, there is a `multicols` aspect which has nothing to do with **etoc**, the input on the left creates a compilation error:

<pre>\begin{multicols}{2}[hello\par world] someone here? \end{multicols}</pre>	<pre>\let\oldpar\par \begin{multicols}{2}[hello\oldpar world] at least me. \end{multicols}</pre>
--------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------

But the version on the right does not (it disguises `\par` so as to be acceptable). **etoc** provides `\etocoldpar` as a substitute for `\par` (it does `\let\etocoldpar\par` just before the `multicols` environment and automatically adds it to close the heading, before the vertical skip of value `\etocinnertopsep`).¹⁹ The command `\etocoldpar` can also be used explicitly in the mandatory argument to `\etocmulticolstyle`.

An important dimension used by all three of `\etocmulticolstyle`, `\etocruledstyle` and `\etocframedstyle` is `\etocinnertopsep`. It gives the amount of separation between the heading and the start of the contents. Its default value is `2ex` and it is changed by `\renewcommand*{\etocinnertopsep}{⟨new_value⟩}`, not with `\setlength`.

10.6. The command `\etocsettocstyle`

This is a command with two mandatory arguments:

`\etocsettocstyle{⟨before_toc⟩}{⟨after_toc⟩}`

¹⁹this command `\etocoldpar` (= working `\par` in the argument to `\etocmulticolstyle`) is not related to the switch `\etocinline` whose purpose is to tell **etoc** not to do a `\par` before the table of contents.

11. Starred variants of the `\tableofcontents` etc... commands

The `{\before_toc}` part is responsible for typesetting the heading, for example it can be something like `\section*{\contentsname}`.

Generally speaking this heading should leave \TeX in vertical mode when the actual typesetting of the contents will start: the line styles (either from the standard classes or the package default line styles) expect to start in ‘vertical mode’.

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, and start a `multicols` environment in `{\before_toc}` to close it in `{\after_toc}`.

10.7. 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.

11. Starred variants of the `\tableofcontents` etc... commands

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

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

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

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

Independently of the situation with the `memoir` class there is generally speaking a hook macro called `\etocaftertitlehook` which is inhibited by using the starred variants of the displaying commands. Except for the `memoir` class, this hook is initially defined to do nothing. There is also `\etocaftercontentshook`, similarly defined to do nothing. They can be used for some special effects.

12. Table of contents for this part

12.1. Testing the compatibility mode

As a third example we now print the local table of contents for this part. First we will test the compatibility mode.²⁰ The original was invisibly defined with a label at the beginning of this part III.

```
\KOMAOptions{toc=left}
\etocstandarddisplaystyle % necessary for the display to obey toc=left
\etocstandardlines
\setcounter{tocdepth}{3}
\tableofcontents \ref{toc:globalcmds}
```

Contents

10. Specifying the toc display style	28
10.1. The command <code>\etocruledstyle</code>	28
10.2. The command <code>\etocmulticolstyle</code>	29
10.2.1. The command <code>\etoctocstyle</code>	29
10.3. The command <code>\etocruled</code>	29
10.4. The commands <code>\etocframedstyle</code> and <code>\etocframed</code>	30
10.5. Headings, titles, <code>\etocinnertopsep</code>	30
10.6. The command <code>\etocsettocstyle</code>	30
10.7. The compatibility mode <code>\etocstandarddisplaystyle</code>	31
11. Starred variants of the <code>\tableofcontents</code> etc... commands	31
12. Table of contents for this part	32
12.1. Testing the compatibility mode	32
12.2. A framed display	32
12.3. A (crazy) inline display	33

12.2. A framed display

We now opt for a “framed” style, using the package default line styles and some colors added.

```
\etocdefaultlines
\begingroup % we use a group to limit the scope of the next commands
\renewcommand{\etoccolumnsep}{2em}
\renewcommand{\etocinnerleftsep}{1.5em}
\renewcommand{\etocinnerrightsep}{1.5em}
% specify a background color for the toc contents
\renewcommand{\etocbkgcolorcmd}{\color{yellow!10}}
%\renewcommand{\etocbkgcolorcmd}{\relax}
```

²⁰the present document uses the `scrartcl` class, and we check here that the `etoc` compatibility mode does respect the customizing done via the class commands.

Contents

```
% set up the top and bottom rules
\renewcommand{\etocprerule}{\hrule height 1pt}
\renewcommand{\etocprerulecolorcmd}{\color{red!25}}
\renewcommand{\etocbottomrule}{\hrule height 1pt}
\renewcommand{\etocbottomrulecolorcmd}{\color{red!25}}
% set up the left and right rules
\renewcommand{\etocleftrule}{\vrule width 5pt}
\renewcommand{\etocrightrule}{\vrule width 5pt}
\renewcommand{\etocleftrulecolorcmd}{\color{red!25}}
\renewcommand{\etocrightrulecolorcmd}{\color{red!25}}
% use \fcolorbox to set up a colored frame for the title
\fbxrule1pt
\etocframedstyle{\normalsize\rmfamily\itshape
  \fcolorbox{red}{white}{\parbox{.8\linewidth}{\centering
    This is a table of contents \a la \etoc, but just for
    the sections and subsections in this part. As it is put
    in a frame, it has to be small enough to fit on the
    current page. It has the label |toc:b|.}}}
% set up a label for future (or earlier...) reference
\setcounter{tocdepth}{3}
\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	
.	10, p. 28
The command <code>\etocruledstyle</code>	
.	10.1, p. 28
The command <code>\etocmulticolstyle</code>	
.	10.2, p. 29
The command <code>\etocdisplaystyle</code> .	
The command <code>\etocruled</code> . . .	
.	10.3, p. 29
The commands <code>\etocframedstyle</code>	
and <code>\etocframed</code> . . .	10.4, p. 30
Headings, titles, <code>\etocinnertop-</code>	
<code>sep</code>	
.	10.5, p. 30

The command <code>\etocsettocstyle</code>	
.	10.6, p. 30
The compatibility mode <code>\etocstandarddisplaystyle</code>	
.	10.7, p. 31
Starred variants of the <code>\tableof-</code>	
<code>contents</code> etc... commands . .	
.	11, p. 31
Table of contents for this part .	
.	12, p. 32
Testing the compatibility mode .	
.	12.1, p. 32
A framed display . . .	12.2, p. 32
A (crazy) inline display	12.3, p. 33

12.3. A (crazy) inline display

Let us finally make some crazy inline display of the table of contents of this entire document. We will typeset the subsections as footnotes... This kind of style is suitable for a

hyperlinked document, not for print!

Here is the inline table of contents. **Overview:** *Initial motivation: nested lists*²¹, *Line styles and toc display style*²², *Arbitrarily many TOCs, and local ones too*²³, *Two examples*²⁴, *Surprising uses of `etoc`*²⁵. **Package commands for line styles:** *The `\etocsetstyle` command*²⁶, *The `\etocsetlevel` command*, *Scope of commands added to the `.toc` file*²⁷, *Am I also red?*. **Package commands for toc display styles:** *Specifying the toc display style*²⁸, *Starred variants of the `\tableofcontents` etc... commands*, *Table of contents for this part*²⁹. **Using and customizing `etoc`:** *Summary of the main style commands*³⁰, *Customizing `etoc`*³¹. **Tips:** *... and tricks*³². **The code:** *Implementation.*

Here is the code which has been used:

```
\begingroup
\setcounter{tocdepth}{3}
\newsavebox{\forsubsections}
\etocsetstyle{part}{\etocskipfirstprefix}{. \upshape}{\bfseries\etocname:~}{\etocname}
\etocsetstyle{section}{\itshape\etocskipfirstprefix}{, }{\mdseries\etocname}{\etocname}
\etocsetstyle{subsection}
{ \begin{lrbox}{\forsubsections}\upshape\etocskipfirstprefix
{; }
{\etocname}
{ \end{lrbox}\footnote{\unhbox\forsubsections.}}
\etocsetstyle{subsubsection}{ (\itshape\etocskipfirstprefix
{, }{\etocname}{\/\upshape})}
\etocsettocstyle{Here is the inline table of contents. }{.\par}
\tableofcontents
\endgroup
```

²¹Limitations intrinsic to the use of environments.

²²`\etocsettocstyle` for the toc display; `\etocsetstyle` for the line styles; Compatibility mode.

²³Labeling and reusing elsewhere; The hyperref option `bookmarksdepth`; On manually adding layout commands to the `.toc` file; Shuffling the levels with `\etocsetlevel`.

²⁴Again the table of contents of this part.

²⁵The TOC of TOCs; Arbitrary “Lists Of...”; A TOC with a fancy layout; Another compatibility mode; The TOC as a tree; The TOC as a molecule.

²⁶The `\etocname` and `\etocpage` commands; The `\etocskipfirstprefix` command; The `\etocnumber` command (*The `\etocifnumbered` switch*); The `\etocthenname`, `\etocthenumber`, and `\etocthepage` commands; The `\etoclink` command; The `\etocthelinkedname`, `\etocthelinkednumber`, `\etocthelinkedpage` and `\etocthelink` commands..

²⁷Testing the scope; **This is a (pale) red subsection for illustrative purposes.**

²⁸The command `\etocruledstyle`; The command `\etocmulticolstyle` (*The command `\etocnocstyle`*); The command `\etocruled`; The commands `\etocframedstyle` and `\etocframed`; Headings, titles, `\etocinnertopsep`; The command `\etocsettocstyle`; The compatibility mode `\etocstandarddisplaystyle`.

²⁹Testing the compatibility mode; A framed display; A (crazy) inline display.

³⁰Setting up local styles; Setting up toc display styles; Displaying tables of contents; Labels and references; The package default line styles: `\etocdefaultlines`; One more example TOC layout.

³¹Customizing the `etoc` pre-defined line styles; Customizing the toc display styles.

³²Hacking framed parboxes; Interverting the levels; All subsections of this document; Displaying statistics; Compatibility with other packages; T_EXnical matters; Errors and catastrophes.

Part IV.

Using and customizing **etoc**

Here are some statistics for this part: it contains 2 sections and 8 subsections. The name of the first section is [Summary of the main style commands](#) and the corresponding number is [13](#). The name of the last section is [Customizing **etoc**](#) and its number is [14](#). The name of the first subsection is [Setting up local styles](#) and the corresponding number is [13.1](#). The name of the last subsection is [Customizing the toc display styles](#) and its number is [14.2](#).

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 13, p. 35	The package default line styles: <code>\etoc-defaultlines</code> 13.5, p. 36
Setting up local styles . . . 13.1, p. 35	One more example TOC layout 13.6, p. 39
Setting up toc display styles 13.2, p. 35	Customizing etoc 14, p. 41
Displaying tables of contents 13.3, p. 36	Customizing the etoc pre-defined line styles 14.1, p. 41
Labels and references . . . 13.4, p. 36	Customizing the toc display styles 14.2, p. 42

13. Summary of the main style commands

13.1. Setting up local styles

```
\etocsetstyle{<levelname>}{<start>}{<prefix>}{<contents>}{<finish>}  
\etocname, \etocnumber, \etocpage, \etocifnumbered{<A>}{<B>}  
\etocthename, \etocthenumber, \etocthepage, \etoclink{<linkname>}
```

13.2. Setting up toc display 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>}
```

13.3. 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

```

13.4. 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 toc display style are the ones currently defined, not the ones from the cloned toc.

13.5. The package default line styles: `\etocdefaultlines`

Activating the use of the package default line styles³³ is done via `\etocdefaultlines`, or `\etococlines` if these styles have not been modified with `\etocsetstyle`. Sections and sub-sections are printed in essentially the same manner, except that the leading for sub-sections is a bit smaller (with document classes lacking a `\chapter` command, the sections are printed in bold typeface; this is the case in the present document). Sub-sub-sections are printed inline, in one paragraph, with no numbers or page numbers. This style was designed and tested with documents having lots of sub-sub-sections, and should be used on a two-column layout: it provides (only in that situation with many sub-sub-sections) a more compact presentation than what is achieved by the L^AT_EX default.³⁴ On the other hand, used with a one-column layout, and with few sub-sub-sections, the style is a bit more spread out vertically than the L^AT_EX default, sub-sections are not visually much different

³³they were written at a very early stage in the development of the package, and version 1.07e has, among other things, modified the previous unsatisfactory use of penalties and vertical spacing commands. This will cause differences to documents having been compiled with earlier versions, apologies for that.

³⁴and there will never be a Part or Chapter entry alone at the bottom of a column or page (except if it has no sub-unit).

from sections (especially for document classes with a `\chapter` command), so the result is less hierarchical in appearance than in the L^AT_EX default.

In this document, for the main table of contents, we did `\etocsetlevel{subsection}{3}` hence the sub-sections were printed with the sub-sub-section inline style.

Let us, to the contrary, typeset now this main table of contents as if the document had been done with a class having the `\chapter` command: we will print sections as chapters, and subsections as sections. We use `\etocsetlevel` for that, and also we need to change the font style of “sections” (which in truth are our subsections) to use not the bold but the medium series; we modify the `\etocfontone` command for that.

```
\etocruledstyle[2]{\normalfont\normalsize\rmfamily\itshape
  \fbox{\parbox{.6\linewidth}{
    \leftskip 0pt plus .5fil
    \rightskip 0pt plus -.5fil
    \parfillskip 0pt plus 1fil This is the global table of
    contents on two columns, using \etoc default line styles, but with
    sections
    being printed as chapters, and subsections as sections.
  }}}
\etocdefaultlines
\setcounter{tocdepth}{1}
\begingroup
\etocsetlevel{section}{0}
\etocsetlevel{subsection}{1}
\renewcommand*{\etocfontone}{\normalfont \normalsize}
\tableofcontents
\endgroup
```

*This is the global table of contents on two columns,
using **etoc** default line styles, but with sections being
printed as chapters, and subsections as sections.*

Part I. Overview

1. Initial motivation: nested lists

Limitations intrinsic to the use of environments 1.1, p. 4

2. Line styles and toc display style

`\etocsettocstyle` for the toc display 2.1, p. 5

`\etocsetstyle` for the line styles 2.2, p. 5

Compatibility mode 2.3, p. 5

3. Arbitrarily many TOCs, and local ones too

Labeling and reusing elsewhere 3.1, p. 6

The hyperref option *bookmarksdepth* 3.2, p. 7

On manually adding layout commands to the .toc file 3.3, p. 7

Shuffling the levels with `\etocsetlevel` 3.4, p. 8

4. Two examples

Again the table of contents of this part 4.1, p. 10

5. Surprising uses of **etoc**

The TOC of TOCs	5.1, p. 10
Arbitrary “Lists Of...”	5.2, p. 13
A TOC with a fancy layout	5.3, p. 14
Another compatibility mode	5.4, p. 15
The TOC as a tree	5.5, p. 17
The TOC as a molecule	5.6, p. 20

Part II. Package commands for line styles

6. The `\etocsetstyle` command

The <code>\etocname</code> and <code>\etocpage</code> commands	6.1, p. 23
The <code>\etocskipfirstprefix</code> command	6.2, p. 24
The <code>\etocnumber</code> command	6.3, p. 24
The <code>\etocthenname</code> , <code>\etocthenumber</code> , and <code>\etocthepage</code> commands	6.4, p. 25
The <code>\etoclink</code> command	6.5, p. 25
The <code>\etocthelinkedname</code> , <code>\etocthelinkednumber</code> , <code>\etocthelinkedpage</code> and <code>\etocthelink</code> commands.	6.6, p. 25

7. The `\etocsetlevel` command

8. Scope of commands added to the `.toc` file

Testing the scope	8.1, p. 27
This is a (pale) red subsection for illustrative purposes	8.2, p. 27

9. Am I also red?

Part III. Package commands for toc display styles

10. Specifying the toc display style

The command <code>\etocruledstyle</code>	10.1, p. 28
The command <code>\etocmulticolstyle</code>	10.2, p. 29
The command <code>\etocruled</code>	10.3, p. 29
The commands <code>\etocframedstyle</code> and <code>\etocframed</code>	10.4, p. 30
Headings, titles, <code>\etocinnertopsep</code>	10.5, p. 30
The command <code>\etocsettocstyle</code>	10.6, p. 30
The compatibility mode <code>\etocstandard-displaystyle</code>	10.7, p. 31

11. Starred variants of the `\table-ofcontents` etc... commands

12. Table of contents for this part

Testing the compatibility mode	12.1, p. 32
A framed display	12.2, p. 32
A (crazy) inline display	12.3, p. 33

Part IV. Using and customizing **etoc**

13. Summary of the main style commands

Setting up local styles	13.1, p. 35
Setting up toc display styles	13.2, p. 35
Displaying tables of contents	13.3, p. 36
Labels and references	13.4, p. 36

13. Summary of the main style commands

The package default line styles: `\etoc-defaultlines` 13.5, p. 36
 One more example TOC layout 13.6, p. 39

14. Customizing `etoc`

Customizing the `etoc` pre-defined line styles 14.1, p. 41
 Customizing the toc display styles 14.2, p. 42

Interverting the levels . . . 15.2, p. 44
 All subsections of this document 15.3, p. 45
 Displaying statistics . . . 15.4, p. 45
 Compatibility with other packages 15.5, p. 47
 T_EXnical matters 15.6, p. 48
 Errors and catastrophes . . 15.7, p. 48

Part V. Tips

15. ... and tricks

Hacking framed parboxes . . . 15.1, p. 44

Part VI. The code

16. Implementation

13.6. One more example TOC layout

I got motivated by a question³⁵ I saw on the T_EX StackExchange site. I copied the color RGB specifications from an answer which had been provided to the question. The `\etoc-framedstyle` puts the title on the top rule in a centered position. This is not very convenient for this example so we included the title as part of the `<start>` code at section level, to get it *inside* the frame.

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

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

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

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

\newcommand\shiftedwhiterule[2]{%
  \hbox to \linewidth{\color{white}%
```

³⁵tex.stackexchange.com/questions/83184

13. Summary of the main style commands

```

\hskip#1\leaders\vrule height1pt\hfil}\nointerlineskip\vskip#2}

\etocsetstyle{subsubsection}{\etocskipfirstprefix}
{\shiftdwhiterule{\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}
{\shiftdwhiterule{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
\fbboxsep0pt % \colorbox uses \fbboxsep 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}
\endgroup

```

The coding is a bit involved³⁶ as it does not use any additional package. Also, it was written at some early stage and I have not revised it since.

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.

³⁶and reveals the author's preference for the T_EX syntax...

I am a twin of that other TOC (click me!)		
10	Specifying the toc display style	28
10.1	The command <code>\etocruledstyle</code>	28
10.2	The command <code>\etocmulticolstyle</code>	29
10.2.1	The command <code>\etocstyle</code>	29
10.3	The command <code>\etocruled</code>	29
10.4	The commands <code>\etocframedstyle</code> and <code>\etocframed</code> . . .	30
10.5	Headings, titles, <code>\etocinnertopsep</code>	30
10.6	The command <code>\etocsettocstyle</code>	30
10.7	The compatibility mode <code>\etocstandarddisplaystyle</code> . . .	31
11	Starred variants of the <code>\tableofcontents</code> etc... commands	31
12	Table of contents for this part	32
12.1	Testing the compatibility mode	32
12.2	A framed display	32
12.3	A (crazy) inline display	33

14. Customizing **etoc**

14.1. Customizing the **etoc** pre-defined 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 \large \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}
\newcommand*\etocsepone{1.5ex plus .3ex minus .3ex}
%%\newcommand*\etocseptwo{1ex plus .15ex minus .15ex} % modified in 1.07e
\newcommand*\etocseptwo{.5ex plus .1ex minus .1ex}
\newcommand*\etocseptthree{.25ex plus .05ex minus .05ex}

\newcommand*\etocminustwoleftmargin{1.5em plus 0.5fil}
\newcommand*\etocminustworightmargin{1.5em plus -0.5fil}

```

14. Customizing *etoc*

```
\newcommand*\etocminusoneleftmargin{1em}
\newcommand*\etocminusonerightmargin{1em}

\newcommand*\etocbaselinespreadminusstwo{1}
\newcommand*\etocbaselinespreadminusone{1}
\newcommand*\etocbaselinespreadzero{1}
\newcommand*\etocbaselinespreadone{1}
\newcommand*\etocbaselinespreadtwo{1}
\newcommand*\etocbaselinespreadthree{.9}
\newcommand*\etocclineleaders
  {\hbox{\normalfont\normalsize\hbox to 2ex {\hss.\hss}}}
\newcommand*\etocabbrevpagename{p.~}
\newcommand*\etocpartname{\partname} % utilisateurs de frenchb: attention
                                       % car donne "partie" sans majuscule.
\newcommand*\etocbookname{Book} % to be modified according to language
```

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*.³⁷

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

Using the command `\etocsetstyle`, be it in the preamble or in the body of the document, has the secondary effect of switching off the compatibility mode.

14.2. Customizing the toc display styles

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

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

\newcommand*\etoccolumnsep{2em}
\newcommand*\etocmulticolsep{0ex}
\newcommand*\etocmulticolpretolerance{-1}
\newcommand*\etocmulticoltolerance{200}
\newcommand*\etocdefaultnbcol{2}
\newcommand*\etocinnertopsep{2ex}
\newcommand*\etocoprul{\hrule}
\newcommand*\etocoprulcolorcmd{\relax}
```

% for the framed style only:

```
\newcommand*\etocinnerleftsep{2em}
\newcommand*\etocinnerrightsep{2em}
```

³⁷with 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.

```

\newcommand*\etocinnerbottomsep{3.5ex}

\newcommand*\etoclefttrule{\vrule}
\newcommand*\etocrightrule{\vrule}
\newcommand*\etocbottomrule{\hrule}
\newcommand*\etoclefttrulecolorcmd{\relax}
\newcommand*\etocrightrulecolorcmd{\relax}
\newcommand*\etocbottomrulecolorcmd{\relax}

\newcommand*\etocbkgcolorcmd{\relax}

% hooks
\newcommand\etocframedmphook{\relax}

```

The `\etocframedmphook` is positioned immediately after the beginning of a minipage environment where the contents of the framed TOC are typeset.

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

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

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

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

The top and bottom rules do not have to be rules and can be horizontal *leaders* (of a specified height) in the general $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 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 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ syntax for rules is accepted here.

Part V.

Tips

Here are some statistics for this part: it contains 1 section and 7 subsections. The name of the first section is [... and tricks](#) and the corresponding number is [15](#). The name of the last section is [... and tricks](#) and its number is [15](#). The name of the first subsection is [Hacking framed parboxes](#) and the corresponding number is [15.1](#). The name of the last subsection is [Errors and catastrophes](#) and its number is [15.7](#).

15. ... and tricks

15.1. Hacking framed parboxes

```
\renewcommand\etocdisplayrule{\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 command <code>\etocsettocstyle</code>
10, p. 28	10.6, p. 30
The command <code>\etocruledstyle</code>	The compatibility mode <code>\etocstandard-</code>
10.1, p. 28	<code>displaystyle</code>
The command <code>\etocmulticolstyle</code>	10.7, p. 31
10.2, p. 29	Starred variants of the <code>\tableofcon-</code>
The command <code>\etocdisplaystyle</code> .	tents etc... commands
	11, p. 31
The command <code>\etocruled</code> 10.3, p. 29	Table of contents for this part
The commands <code>\etocframedstyle</code> and	12, p. 32
<code>\etocframed</code>	Testing the compatibility mode
10.4, p. 30	12.1, p. 32
Headings, titles, <code>\etocinnertopsep</code>	A framed display
10.5, p. 30	12.2, p. 32
	A (crazy) inline display
	12.3, p. 33

15.2. Interverting the levels

Let us display and count all subsections occurring in this document (see section 5 for other uses of this technique):

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

15.3. All subsections of this document

i. Limitations intrinsic to the use of environments (1.1, p. 4)	xvii. The <code>\etocskipfirstprefix</code> command (6.2, p. 24)	xxxii. A framed display (12.2, p. 32)
ii. <code>\etocsettocstyle</code> for the toc display (2.1, p. 5)	xviii. The <code>\etocnumber</code> command (6.3, p. 24)	xxxiii. A (crazy) inline display (12.3, p. 33)
iii. <code>\etocsetstyle</code> for the line styles (2.2, p. 5)	xix. The <code>\etocthename</code> , <code>\etocthenumber</code> , and <code>\etocthepage</code> commands (6.4, p. 25)	xxxiv. Setting up local styles (13.1, p. 35)
iv. Compatibility mode (2.3, p. 5)	xx. The <code>\etoclink</code> command (6.5, p. 25)	xxxv. Setting up toc display styles (13.2, p. 35)
v. Labeling and reusing elsewhere (3.1, p. 6)	xxi. The <code>\etocthelinkedname</code> , <code>\etocthelinkednumber</code> , <code>\etocthelinkedpage</code> and <code>\etocthelink</code> commands. (6.6, p. 25)	xxxvi. Displaying tables of contents (13.3, p. 36)
vi. The hyperref option <code>bookmarksdepth</code> (3.2, p. 7)	xxii. Testing the scope (8.1, p. 27)	xxxvii. Labels and references (13.4, p. 36)
vii. On manually adding layout commands to the .toc file (3.3, p. 7)	xxiii. This is a (pale) red subsection for illustrative purposes (8.2, p. 27)	xxxviii. The package default line styles: <code>\etocdefaultlines</code> (13.5, p. 36)
viii. Shuffling the levels with <code>\etocsetlevel</code> (3.4, p. 8)	xxiv. The command <code>\etocruledstyle</code> (10.1, p. 28)	xxxix. One more example TOC layout (13.6, p. 39)
ix. Again the table of contents of this part (4.1, p. 10)	xxv. The command <code>\etocmulticolstyle</code> (10.2, p. 29)	xl. Customizing the <code>etoc</code> pre-defined line styles (14.1, p. 41)
x. The TOC of TOCs (5.1, p. 10)	xxvi. The command <code>\etocruled</code> (10.3, p. 29)	xli. Customizing the toc display styles (14.2, p. 42)
xi. Arbitrary “Lists Of...” (5.2, p. 13)	xxvii. The commands <code>\etocframedstyle</code> and <code>\etocframed</code> (10.4, p. 30)	xlii. Hacking framed par-boxes (15.1, p. 44)
xii. A TOC with a fancy layout (5.3, p. 14)	xxviii. Headings, titles, <code>\etocinnertopsep</code> (10.5, p. 30)	xlili. Interverting the levels (15.2, p. 44)
xiii. Another compatibility mode (5.4, p. 15)	xxix. The command <code>\etocsettocstyle</code> (10.6, p. 30)	xliv. All subsections of this document (15.3, p. 45)
xiv. The TOC as a tree (5.5, p. 17)	xxx. The compatibility mode <code>\etocstandarddisplaystyle</code> (10.7, p. 31)	xlvi. Displaying statistics (15.4, p. 45)
xv. The TOC as a molecule (5.6, p. 20)	xxxi. Testing the compatibility mode (12.1, p. 32)	xlvi. Compatibility with other packages (15.5, p. 47)
xvi. The <code>\etocname</code> and <code>\etocpage</code> commands (6.1, p. 23)		xlvi. T _E Xnical matters (15.6, p. 48)
		xlvi. Errors and catastrophes (15.7, p. 48)

15.4. Displaying statistics

Each part of this document starts with a paragraph telling how many sections and subsections it has. Well, each one of this paragraph is a table of contents! We designed a macro `\thispartstats` to do that. It uses “storage” boxes to keep the information about the first and last section or subsection. Using boxes is the simplest manner to encapsulate the

List of all subsections

hyperref link for later use (whether there is one or none). However, one cannot modify then the font or the color (using the \TeX primitive `\setbox` rather than the \LaTeX `\sbox` would allow to change the color of the un-boxed saved box). If such a need arises, one must switch from boxes to macros, and store the hyperref data for later use as was done in the code presented in the subsection 5.6. We did this for the first paragraph of the section 5.

But first, the coding of `\thispartstats`:

```
\newsavebox\firstnamei \newsavebox\firstnumberi
\newsavebox\lastnamei \newsavebox\lastnumberi
\newsavebox\firstnameii \newsavebox\firstnumberii
\newsavebox\lastnameii \newsavebox\lastnumberii
\newcounter{mycounti} \newcounter{mycountii}
\newcommand*{\thispartstatsauxi}{} \newcommand*{\thispartstatsauxii}{}
\newcommand*{\oldtocdepth}{}
\newcommand*{\thispartstats}{%
  \edef\oldtocdepth{\arabic{tocdepth}}%
  \setcounter{tocdepth}{2}%
  \setcounter{mycounti}{0}%
  \setcounter{mycountii}{0}%
  \def\thispartstatsauxi{%
    \sbox{\firstnamei}{\color{cyan}\etocname}%
    \sbox{\firstnumberi}{\color{cyan}\etocnumber}%
    \def\thispartstatsauxi{}}%
  \def\thispartstatsauxii{%
    \sbox{\firstnameii}{\color{cyan}\etocname}%
    \sbox{\firstnumberii}{\color{cyan}\etocnumber}%
    \def\thispartstatsauxii{}}%
  \begingroup
  \etocsetstyle{subsection} {} {}
  {\thispartstatsauxii
   \stepcounter{mycountii}%
   \sbox{\lastnameii}{\color{teal}\etocname}%
   \sbox{\lastnumberii}{\color{teal}\etocnumber}} {}%
  \etocsetstyle{section} {} {}
  {\thispartstatsauxi
   \stepcounter{mycounti}%
   \sbox{\lastnamei}{\color{teal}\etocname}%
   \sbox{\lastnumberi}{\color{teal}\etocnumber}}
  {Here are some statistics for this part: it contains \arabic{mycounti}
  section\ifnum\value{mycounti}>1 s\fi} and \arabic{mycountii}
  subsection\ifnum\value{mycountii}>1 s\fi. The name of the first section is
  \unhbox\firstnamei} and the corresponding number is \unhbox\firstnumberi.
  The name of the last section is \unhbox\lastnamei} and its number is
  \unhbox\lastnumberi. The name of the first subsection is \unhbox\firstnameii}
  and the corresponding number is \unhbox\firstnumberii. The name of the last
  subsection is \unhbox\lastnameii} and its number is \unhbox\lastnumberii.}%
  \etocinline % don't do a \par automatically (but this is not used, actually).
  \etocsettocstyle {}{}
  \localtableofcontents % to be used at the top level of a Part.
  \endgroup
  \setcounter{tocdepth}{\oldtocdepth}%
}
```

And now, the variant which was used for section 5, with macros rather than boxes:

```
\makeatletter
```

List of all subsections

```
\newcommand*\firstsubname {} \newcommand*\lastsubname {}
\newcommand*\firstsubnumber {} \newcommand*\lastsubnumber {}
\newcommand*\thissectionstatsaux{}

\newcommand*\thissectionstats{%
  \edef\oldtocdepth{\arabic{tocdepth}}%
  \setcounter{tocdepth}{2}%
  \setcounter{mycounti}{0}%
  \def\thissectionstatsaux{% ou plus simple si on ne veut pas le lien.
    \toks@{\expandafter{\etocthename}}%
    \edef\firstsubname{\noexpand\hyperlink{\Hy@tocdestname}{\the\toks@}}%
    \toks@{\expandafter{\etocthenumber}}%
    \edef\firstsubnumber{\noexpand\hyperlink{\Hy@tocdestname}{\the\toks@}}%
    \def\thissectionstatsaux{}}
  \begingroup
  \etocsetstyle{subsection} {} {}
  {\thissectionstatsaux
    \stepcounter{mycounti}%
    \toks@{\expandafter{\etocthename}}%
    \edef\lastsubname{\noexpand\hyperlink{\Hy@tocdestname}{\the\toks@}}%
    \toks@{\expandafter{\etocthenumber}}%
    \edef\lastsubnumber{\noexpand\hyperlink{\Hy@tocdestname}{\the\toks@}}%
  }
  {Here are some statistics for this section. It contains \arabic{mycounti}
  subsections. The name of its first is \emph{\color{cyan}\firstsubname}}
  and the corresponding number is {\color{cyan}\firstsubnumber}. The name of
  the last subsection is \emph{\color{teal}\lastsubname}} and its number is
  {\color{teal}\lastsubnumber}.}% (with boxes, \emph would have done nothing)
  \etocsettocstyle {}{}
  \etocinline
  \localtableofcontents
  \endgroup
  \setcounter{tocdepth}{\oldtocdepth}%
}
\makeatother
```

15.5. Compatibility with other packages

etoc loads the package `multicol`.³⁸ It is `hyperref` aware and hopefully `hyperref` compatible! It doesn't matter whether **etoc** or `hyperref` is loaded first.

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.

etoc can not really cohabit with packages modifying the `\tableofcontents` command: some sort of truce can be achieved if **etoc** is loaded last, hence is the winner.

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.

etoc expects the document sectioning units to write their data into the file having extension `.toc`, in the form of lines containing the `\contentsline` command and its arguments. The macros `\etocname`, `\etocnumber`, and `\etocpage` contain the `hyperref`

³⁸ up to version 1.07a it also used package `xspace`, but this has been removed from later versions.

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). For example, the tables of contents of the present document are all fully linked.³⁹

It is possible to customize (using package `tocloft` for example) throughout the document the macros `\l@section`, `\l@subsection` ... and the effect will be seen in the next table of contents typeset by `etoc` in compatibility mode.

It is possible to use simultaneously `etoc` and `tableof`⁴⁰. For the advanced uses such as what is done in the subsection 5.6 it is important to know that `tableof` adds one level of grouping inside the `.toc` file itself. So when one needs to make some information `\global`, one can not wait to be at the level of the second argument of `\etocsettocstyle`, as `tableof` will already have closed the group then. The `\global` things must be done at the latest in the *finish* part of the top (or last) sectioning level used. This only applies of course to `\tableofcontents` or `\localtableofcontents` following the `\nexttocwithtags{required}{forbidden}` command from `tableof`.

And when the commands `\tableof` or `\tablenotof` of package `tableof` are used, they typeset the table of contents according to the document class defaults: to benefit from the `etoc` styles, it is mandatory to use either `\tableofcontents`, or `\localtableofcontents` or one of the other `etoc` commands, and `tableof` (v1.1) will influence the outcome only if `\nexttocwithtags{required}{forbidden}` was added before the table of contents typesetting command.

15.6. T_EXnical matters

The `\etocname`, `\etocnumber`, `\etocpage` commands are protected against premature expansion. They contain suitable `hyperref` links if package `hyperref` is loaded and active for the TOC. The commands `\etoclink` and `\etocifnumbered` are also protected against premature expansion.

On the other hand `\etocthenname`, `\etocthenumber`, `\etocthepage` do not represent `hyperref` links, and are *not* protected against expansion.

The commands such as `\etocsetstyle`, `\etocsetlevel`, `\etocsettocstyle`, `\etocmulticolstyle`, `\etocruledstyle`, `\etocframedstyle` obey L^AT_EX's groups. All TOCs are typeset inside groups.

15.7. Errors and catastrophes

After using `\etocsetstyle` for one level, the remaining uncustomized levels use the `etoc` default styles (those which are activated by `\etocdefaultlines`). One has to make sure that all levels needed for the next table of contents are mutually compatible: in particular the `etoc` default styles expect to start in “vertical mode”.

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 (but this should never be) these instructions should be done in such a way that they can be activated or deactivated easily from the document source, as need be.

³⁹except the Qtree TOC ...

⁴⁰<http://www.ctan.org/pkg/tableof>

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).

This is the documentation as of 2013/04/06, printed from the source file with the time stamp “06-04-2013 at 13:48:01 CEST”. The package version is v1.07f, of 2013/03/07. See the source for copyright and license information.

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.

16. Implementation

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

```
1 \ProvidesPackage{etoc}
2 [2013/03/07 v1.07f Completely customisable TOCs (jfb)]
3 \NeedsTeXFormat{LaTeX2e}
4 \RequirePackage{multicol}
5 %% \RequirePackage{xspace} %% REMOVED (1.07b)
6 \DeclareOption*{\PackageWarning{etoc}{Option ‘\CurrentOption’ is unknown.}}
7 \ProcessOptions\relax

placeholder for comments

8 \newtoks\Etoc@toctoks
9 \def\Etoc@par{\par}
10 \newcommand*{\etocinline}{\def\Etoc@par{}}
11 \let\etocnopar\etocinline
12 \newif\ifEtoc@jj % book
13 \newif\ifEtoc@j % part
14 \newif\ifEtoc@ % chapter
15 \newif\ifEtoc@i % section
16 \newif\ifEtoc@ii % subsection
17 \newif\ifEtoc@iii % subsubsection
18 \newif\ifEtoc@iv % paragraph
19 \newif\ifEtoc@v % subparagraph
20 \newif\ifEtoc@number
21 \newif\ifEtoc@hyperref
22 \newif\ifEtoc@parskip % 1.07d
23 \newif\ifEtoc@tocwithid
24 \newif\ifEtoc@standard
25 \newif\ifEtoc@part
```

16. Implementation

placeholder for comments

```

26 \newif\ifEtoc@localtoc
27 \newif\ifEtoc@skipthisone
28 \newif\ifEtoc@stoptoc
29 \newif\ifEtoc@notactive
30 \newcounter{etoc@tocid}
31 \newif\ifEtoc@mustclosegroup
32 \def\etoc@{\etoc@} % Delimiter in \Etoc@getnb etc.. macros. This def added 1.07f

```

placeholder for comments

```

33 \@ifclassloaded{memoir}{\def\Etoc@minf{-\thr@@}{\def\Etoc@minf{-\tw@}}
34 \def\Etoc@@minustwo@@{-\tw@}
35 \let\Etoc@@minusone@@\m@ne
36 \chardef\Etoc@@zero@@ 0
37 \let\Etoc@@one@@ \@ne
38 \let\Etoc@@two@@ \tw@
39 \let\Etoc@@three@@ \thr@@
40 \chardef\Etoc@@four@@ 4
41 \chardef\Etoc@@five@@ 5
42 \chardef\Etoc@@six@@ 6
43 \let\Etoc@localtop\Etoc@@minustwo@@
44 \def\Etoc@@minustwo@{minustwo}
45 \def\Etoc@@minusone@{minusone}
46 \def\Etoc@@zero@{zero}
47 \def\Etoc@@one@{one}
48 \def\Etoc@@two@{two}
49 \def\Etoc@@three@{three}
50 \def\Etoc@@four@{four}
51 \def\Etoc@@five@{five}
52 %\def\Etoc@@six@{six}

```

placeholder for comments

```

53 \def\Etoc@levellist{}
54 \def\Etoc@newlevel#1{%
55   \def\Etoc@levellist@elt{\noexpand\Etoc@levellist@elt\noexpand}%
56   \edef\Etoc@levellist{\Etoc@levellist\Etoc@levellist@elt#1}%
57 \def\etocsetlevel#1#2{%
58   \expandafter\Etoc@newlevel\csname l@#1\endcsname
59   \ifcase#2\relax
60     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@zero@@
61     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@zero@
62   \or
63     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@one@@
64     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@one@
65   \or
66     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@two@@
67     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@two@
68   \or
69     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@three@@
70     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@three@
71   \or
72     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@four@@
73     \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@four@

```

16. Implementation

```

74 \or
75 \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@five@@
76 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@five@
77 \or
78 \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@six@@
79 \else
80 \ifnum#2=\m@ne
81 \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@minusone@@
82 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@minusone@
83 \else
84 \ifnum#2=-\tw@
85 \expandafter\let \csname Etoc@#1@@\endcsname\Etoc@@minustwo@@
86 \expandafter\let \csname Etoc@#1@\endcsname\Etoc@@minustwo@
87 \else
88 \PackageWarning{etoc}
89 {unexpected value ‘#2’ in \string\etocsetlevel.^^J}%
90 Should be -2,-1, 0, 1, 2, 3, 4, 5, or 6. Set to 6 (=ignored)}%
91 \expandafter\let\csname Etoc@#1@@\endcsname\Etoc@@six@@
92 \fi\fi\fi}
93 \etocsetlevel{book}{-2}
94 \etocsetlevel{part}{-1}
95 \etocsetlevel{chapter}{0}
96 \etocsetlevel{section}{1}
97 \etocsetlevel{subsection}{2}
98 \etocsetlevel{subsubsection}{3}
99 \etocsetlevel{paragraph}{4}
100 \etocsetlevel{subparagraph}{5}
placeholder for comments
101 \def\Etoc@setflags #1{%
102 \ifcase #1\relax
103 \global\Etoc@vfalse
104 \global\Etoc@ivfalse
105 \global\Etoc@iiifalse
106 \global\Etoc@iifalse
107 \global\Etoc@ifalse
108 \global\Etoc@true
109 \or
110 \global\Etoc@vfalse
111 \global\Etoc@ivfalse
112 \global\Etoc@iiifalse
113 \global\Etoc@iifalse
114 \global\Etoc@itrue
115 \or
116 \global\Etoc@vfalse
117 \global\Etoc@ivfalse
118 \global\Etoc@iiifalse
119 \global\Etoc@iitrue
120 \or
121 \global\Etoc@vfalse
122 \global\Etoc@ivfalse
123 \global\Etoc@iiitrue
124 \or

```

16. Implementation

```

125     \global\Etoc@vfalse
126     \global\Etoc@ivtrue
127 \or
128     \global\Etoc@vtrue
129 \else
130     \ifnum#1=\m@ne
131         \global\Etoc@vfalse
132         \global\Etoc@ivfalse
133         \global\Etoc@iiifalse
134         \global\Etoc@iifalse
135         \global\Etoc@ifalse
136         \global\Etoc@false
137         \global\Etoc@jtrue
138     \else
139         \global\Etoc@vfalse
140         \global\Etoc@ivfalse
141         \global\Etoc@iiifalse
142         \global\Etoc@iifalse
143         \global\Etoc@ifalse
144         \global\Etoc@false
145         \global\Etoc@jfalse
146         \global\Etoc@jjtrue
147     \fi
148 \fi}

placeholder for comments

149 \AtBeginDocument{%
150 \ifpackageloaded{parskip}{\Etoc@parskiptrue}{}%
151 \ifpackageloaded{hyperref}{\Etoc@hyperreftrue
152                                     \def\Etoc@et@hop#1#2#3#4#5{#1{#3}{#4}{#5}#2}%
153                                     \long\def\Etoc@gobblesixorfive#1#2#3#4#5#6{}}
154                                     {\def\Etoc@et@hop#1#2#3#4{#1{#3}{#4}#2}%
155                                     \long\def\Etoc@gobblesixorfive#1#2#3#4#5{}}}%
156 }

placeholder for comments

157 \def\Etoc@swa#1{%
158     \Etoc@et@hop
159     {\Etoc@savedcontentsline{#1}}
160     {\Etoc@prefix\Etoc@contents}}
161 \def\Etoc@swb#1{%
162     \Etoc@et@hop
163     {\Etoc@savedcontentsline{#1}}
164     {\Etoc@contents}}
165 \let\etocskipfirstprefix\@thirdofthree

placeholder for comments

166 \def\Etoc@etoccontentsline#1{%
167     \global\expandafter\let\expandafter\Etoc@tmp\csname Etoc@#1@@\endcsname
168     \Etoc@skipthisonefalse
169     \let\Etoc@next\Etoc@gobblesixorfive
170     \ifnum\Etoc@tmp=\Etoc@@six@@
171         \Etoc@skipthisonetrue
172     \else

```

16. Implementation

```

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

```

16. Implementation

```

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

[2013/03/07]:

Up to 1.06 **etoc** defined only `\etocname`, `\etocnumber` and `\etocpage`. The `hyperref` added data is recycled in the simplest manner, prefixing it with `\leavevmode`. The included `\Hy@tocdestname` is left unexpanded. Due to the initial (enormously incredible and opposite to the credo of separating the content from the form) \LaTeX flaw of mixing (Parts having their own format, not to mention other classes) in the `.toc` file the number and the name we have to spend some time with delimited macros to dis-entangle this, and reconstruct the possible `hyperref` data. Note that if the page number is not hyperlinked, `\etocpage` does *not* add the link found possibly in the name.

Then 1.07 added `\etocthename`, `\etocthenumber`, `\etocthepage` which are left fragile and do not have the links data, and `\etoclink{<linkname>}` which is robust and reconstructs an arbitrarily named link. A need (for things like building up a token list to be used in a `tikzpicture`) arose later to have some form of the link which could be saved by a simple command like one can do `\global\let\lastname\etocthename`, and avoid having to manipulate `\Hy@tocdestname`. So 1.07f adds `\etocthelinkedname`, `\etoc-`

16. Implementation

`thelinkednumber`, `\etocthelinkedpage`, `\etocthelink`: they use `\hyperlink` with an expanded `\Hy@tocdestname`.

One could now define `\etocname`, etc ... to be the robust versions of `\etocthelinkedname`, etc ..., but the original definitions are kept by sentimentalism. 1.07f also adds `\leavevmode` to `\etoclink` which should have been done earlier, as it was included in `\etocname` etc...

attention, `\@namedef{A}{B}` and not `\@namedef{A} {B}` !! on the other hand this gives a simple way to insert a space as the first token in the paramaters. For `\Etoc@again` (which appears later in the code), a `\@firstofone` construct is however the simplest of all.

```

260 \def\Etoc@lxyz #1#2{%
261     \@namedef {etoclink }{\leavevmode}%           fall-back
262     \def\etocthelink      {}%                     fall-back
263     \@namedef {etocname }{\leavevmode #1}% fall-back (perhaps linked)
264     \def\etocthename      {#1}% (if link, will be removed later)
265     \def\etocthelinkedname {#1}% will probably get redefined
266     \@namedef {etocpage }{\leavevmode #2}% (perhaps linked)
267     \Etoc@getthepage #2\etoc@ % defines also \etocthelinkedpage (and \etoclink)
268     \Etoc@getnb #1\relax\relax\etoc@ % gets number *and* name, and \etoclink
269     \ifEtoc@number\else
270         \ifEtoc@part
271             \Etoc@getit #1\hspace\relax\etoc@ % additional job for parts
272         \fi
273     \fi}

placeholder for comments
274 \def\Etoc@getthepage #1{%
275     \let\Etoc@next\Etoc@getthepage@nohyp
276     \ifEtoc@hyperref\ifx #1\hyper@linkstart
277         \let\Etoc@next\Etoc@getthepage@hyp
278     \fi\fi
279     \Etoc@next #1%
280 }
281 \def\Etoc@getthepage@nohyp #1\etoc@ {%
282     \def\etocthepage      {#1}%
283     \def\etocthelinkedpage {#1}%
284 }
285 \def\Etoc@getthepage@hyp #1#2#3#4#5\etoc@ {%
286     \@namedef{etoclink } ##1{\leavevmode #1{#2}{#3}{##1}{#5}%
287     \edef\etocthelink ##1{\noexpand\hyperlink {#3}{##1}}%
288     \def\etocthepage {#4}%
289     \toks@ {#4}%
290     \edef\etocthelinkedpage {\noexpand\hyperlink {#3}{\the\toks@}}%
291 }
292 %
293 \def\Etoc@getnb #1{%
294     \let\Etoc@next\Etoc@getnb@nohyp
295     \ifEtoc@hyperref\ifx #1\hyper@linkstart
296         \let\Etoc@next\Etoc@getnb@hyp
297     \fi\fi
298     \Etoc@next #1%
299 }
300 %

```

16. Implementation

```

301 \def\Etoc@getit #1{%
302   \let\Etoc@next\Etoc@getit@nohyp
303   \ifEtoc@hyperref\ifx #1\hyper@linkstart
304     \let\Etoc@next\Etoc@getit@hyp
305   \fi\fi
306   \Etoc@next #1%
307 }

placeholder for comments
308 \def\Etoc@getnb@nohyp #1#2#3\etoc@ {%
309   \def\Etoc@getname ##1\relax\relax\etoc@ {%
310     \@namedef {etocname }{\leavevmode ##1}%
311     \def\etocthename      {##1}%
312     \def\etocthelinkedname {##1}%
313   }%
314   \ifx #1\numberline
315     \@namedef {etocnumber }{\leavevmode #2}%
316     \def\etocthenumber      {#2}%
317     \def\etocthelinkednumber {#2}%
318     \Etoc@numbertrue
319     \Etoc@getname #3\etoc@
320   \else % then \etocthename and \etocthelinkedname already defined
321     \@namedef {etocnumber }{\leavevmode}%
322     \def\etocthenumber      {}%
323     \def\etocthelinkednumber {}%
324     \Etoc@numberfalse
325   \fi
326 }

placeholder for comments
327 \def\Etoc@getnb@hyp #1#2#3#4#5#6\etoc@ {%
328   \def\Etoc@getname ##1\relax\relax\etoc@ {%
329     \@namedef {etocname }{\leavevmode #1{#2}{#3}{##1}#5}%
330     \def\etocthename {##1}%
331     \toks@           {##1}%
332     \edef\etocthelinkedname {\noexpand\hyperlink {#3}{\the\toks@}}%
333   }%
334   \def\Etoc@getnbr ##1##2##3\etoc@ {%
335     \ifx ##1\numberline
336       \@namedef {etocnumber }{\leavevmode #1{#2}{#3}{##2}#5}%
337       \def\etocthenumber {##2}%
338       \toks@             {##2}%
339       \edef\etocthelinkednumber {\noexpand\hyperlink {#3}{\the\toks@}}%
340       \Etoc@numbertrue
341       \Etoc@getname ##3\etoc@
342     \else
343       \@namedef {etocnumber }{\leavevmode}%
344       \def\etocthenumber      {}%
345       \def\etocthelinkednumber {}%
346       \Etoc@numberfalse
347       \def\etocthename {#4}%
348       \toks@           {#4}%
349       \edef\etocthelinkedname {\noexpand\hyperlink {#3}{\the\toks@}}%

```

16. Implementation

```

350         \fi
351     }%
352 \namedef {etoclink }##1{\leavevmode #1{#2}{#3}{##1}#5}%
353 \edef\etocthelink ##1{\noexpand\hyperlink {#3}{##1}}%
354 \Etoc@getnbr #4\relax\relax\etoc@
355 }

placeholder for comments
356 \def\Etoc@getit@nohyp #1\hspace#2#3\etoc@ {%
357     \def\Etoc@getname ##1\hspace\relax\etoc@ {%
358         \namedef {etocname }{\leavevmode ##1}%
359         \def\etocthename {##1}%
360         \def\etocthelinkedname {##1}%
361     }%
362 \ifx\relax#2\else
363     \namedef {etocnumber }{\leavevmode #1}%
364     \def\etocthenumber {#1}%
365     \def\etocthelinkednumber {#1}%
366     \Etoc@numbertrue
367     \Etoc@getname #3\etoc@
368 \fi
369 }

placeholder for comments
370 \def\Etoc@getit@hyp #1#2#3#4#5#6\etoc@ {%
371     \def\Etoc@getname ##1\hspace\relax\etoc@ {%
372         \namedef {etocname }{\leavevmode #1{#2}{#3}{##1}#5}%
373         \def\etocthename {##1}%
374         \toks@ {##1}%
375         \edef\etocthelinkedname {\noexpand\hyperlink {#3}{\the\toks@}}%
376     }%
377     \def\Etoc@getnbr ##1\hspace##2##3\etoc@ {%
378         \ifx\relax##2\else
379             \namedef {etocnumber }{\leavevmode #1{#2}{#3}{##1}#5}%
380             \def\etocthenumber {##1}%
381             \toks@ {##1}%
382             \edef\etocthelinkednumber {\noexpand\hyperlink {#3}{\the\toks@}}%
383             \Etoc@numbertrue
384             \Etoc@getname ##3\etoc@
385         \fi
386     }%
387     \Etoc@getnbr #4\hspace\relax\etoc@
388 }

placeholder for comments
389 \newcommand*\etocthename {}
390 \newcommand*\etocthenumber {}
391 \newcommand*\etocthepage {}
392 \newcommand*\etocthelinkedname {}
393 \newcommand*\etocthelinkednumber {}
394 \newcommand*\etocthelinkedpage {}
395 \newcommand*\etocthelink {}
396 \DeclareRobustCommand*\etocname {} {}
397 \DeclareRobustCommand*\etocnumber {} {}

```

16. Implementation

```

398 \DeclareRobustCommand*\etocpage{} {}
399 \DeclareRobustCommand*\etoclink{} {}
400 \DeclareRobustCommand*\etocifnumbered{}
401 {\ifEtoc@number\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}

placeholder for comments
402 \def\Etoc@readtoc#1{%
403   \ifeof #1
404     \let\Etoc@nextread\@gobble
405     \global\Etoc@toctoks=\expandafter{\the\Etoc@toctoks}%
406   \else
407     \let\Etoc@nextread\Etoc@readtoc
408     \read #1 to \Etoc@buffer
409     \Etoc@toctoks=\expandafter\expandafter\expandafter
410       {\expandafter\the\expandafter\Etoc@toctoks\Etoc@buffer}%
411   \fi
412   \Etoc@nextread{#1}%
413 }
414 \IfFileExists{\jobname .toc}
415   {\endlinechar=-1
416     \makeatletter
417     \newread\Etoc@tf
418     \openin\Etoc@tf\@filef@und
419     \Etoc@readtoc\Etoc@tf
420     \closein\Etoc@tf}
421   {\typeout{No file \jobname .toc.}}

placeholder for comments
422 \def\Etoc@openouttoc{% formerly \Etoc@starttoc
423 %% 1.07d: parskip and \@nobreakfalse stuff moved to \Etoc@tableofcontents
424 \ifEtoc@hyperref
425   \ifx\hyper@last\@undefined
426     \IfFileExists{\jobname .toc}
427       {\Hy@WarningNoLine
428         {old toc file detected, not used; run LaTeX again (cheers from etoc)}}%
429       \global\Etoc@toctoks={}%
430     }
431   {}%
432 \fi
433 \fi
434 \if@filesw
435   \newwrite \tf@toc
436   \immediate \openout \tf@toc \jobname .toc\relax
437 \fi
438 \gdef\Etoc@openouttoc{}% 1.07d, rather than using a boolean
439 }

placeholder for comments
440 \def\Etoc@toctoc{%
441   \global\Etoc@vfalse
442   \global\Etoc@ivfalse
443   \global\Etoc@iiiifalse
444   \global\Etoc@iifalse
445   \global\Etoc@ifalse

```

16. Implementation

```

446 \global\Etoc@false
447 \global\Etoc@jfalse
448 \global\Etoc@jjfalse
449 \ifEtoc@standard
450 \etoc@setstyle{@minustwo}{}{}{}{}%
451 \etoc@setstyle{@minusone}{}{}{}{}%
452 \etoc@setstyle{@zero}{}{}{}{}%
453 \etoc@setstyle{@one}{}{}{}{}%
454 \etoc@setstyle{@two}{}{}{}{}%
455 \etoc@setstyle{@three}{}{}{}{}%
456 \etoc@setstyle{@four}{}{}{}{}%
457 \etoc@setstyle{@five}{}{}{}{}%
458 \else
459 \def\Etoc@levellist@elt##1{\let##1\Etoc@lxyz}%
460 \Etoc@levellist
461 \let\booknumberline\numberline
462 \let\partnumberline\numberline
463 \let\chapternumberline\numberline
464 \fi
465 \the\Etoc@toctoks
466 \ifEtoc@notactive\else
467 \ifEtoc@v \Etoc@end@five\fi
468 \ifEtoc@iv \Etoc@end@four\fi
469 \ifEtoc@iii \Etoc@end@three\fi
470 \ifEtoc@ii \Etoc@end@two\fi
471 \ifEtoc@i \Etoc@end@one\fi
472 \ifEtoc@ \Etoc@end@zero\fi
473 \ifEtoc@j \Etoc@end@minusone\fi
474 \ifEtoc@jj \Etoc@end@minustwo\fi
475 \fi}

placeholder for comments

476 \def\etoc@@startlocaltoc#1#2{%
477 \let\Etoc@next\relax
478 \ifEtoc@localtoc
479 \ifEtoc@notactive
480 \ifnum #1=#2\relax
481 \ifEtoc@jj \let\Etoc@localtop\Etoc@@minusone@@ \fi
482 \ifEtoc@j \let\Etoc@localtop\Etoc@@zero@@ \fi
483 \ifEtoc@ \let\Etoc@localtop\Etoc@@one@@ \fi
484 \ifEtoc@i \let\Etoc@localtop\Etoc@@two@@ \fi
485 \ifEtoc@ii \let\Etoc@localtop\Etoc@@three@@ \fi
486 \ifEtoc@iii \let\Etoc@localtop\Etoc@@four@@ \fi
487 \ifEtoc@iv \let\Etoc@localtop\Etoc@@five@@ \fi
488 \ifEtoc@v \let\Etoc@localtop\Etoc@@six@@ \fi
489 \def\Etoc@next{\global\Etoc@notactivefalse
490 \global\Etoc@vfalse
491 \global\Etoc@ivfalse
492 \global\Etoc@iiifalse
493 \global\Etoc@iifalse
494 \global\Etoc@ifalse
495 \global\Etoc@false
496 \global\Etoc@jfalse

```

16. Implementation

```

497 \global\Etoc@jjfalse}%
498 \fi
499 \fi\fi
500 \Etoc@next}
501 \let\etoc@startlocaltoc\@gobble

placeholder for comments
502 \def\Etoc@localtableofcontents#1{%
503 \edef\Etoc@tmp{#1}%
504 \ifnum\Etoc@tmp<\@ne
505 \PackageWarning{etoc}
506 {Unknown toc id: run LaTeX to get references right}%
507 \leavevmode --unknown etoc ref: run latex again--\par
508 \let\Etoc@next\@gobble\else\let\Etoc@next\@firstofone\fi
509 \Etoc@next
510 {\edef\etoc@startlocaltoc##1{%
511 \noexpand\etoc@@startlocaltoc{##1}{#1}}
512 \Etoc@localtoctrue
513 \let\Etoc@localtop\Etoc@minustwo@@
514 \global\Etoc@stoptocfalse
515 \global\Etoc@notactivetrue
516 \Etoc@tableofcontents}%
517 \endgroup\ifEtoc@mustclosegroup\endgroup\fi}

[2013/03/07]: I discover a \@namedef trick to construct the \Etoc@again space delimited macro:
\@namedef {Etoc@again} {...stuff...}
Original version was (copied from analogous stuff in source2e):
{\def\1{\Etoc@again}\expandafter\gdef\1 {...stuff...}}
and in the end (now that I think about it) I simply use \@firstofone.

518 \def\Etoc@getrefno #1#2\etoc@ {#1}
519 \def\Etoc@getref #1{\@ifundefined{r@#1}{0}{\expandafter\expandafter\expandafter
520 \Etoc@getrefno\csname r@#1\endcsname\relax\etoc@}}
521 \def\Etoc@ref#1{\Etoc@localtableofcontents{\Etoc@getref{#1}}}
522 \def\Etoc@label#1{\label{#1}\futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
523 \@firstofone{\def\Etoc@again} {\futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}

placeholder for comments
524 \def\Etoc@t@bleofcontents{%
525 \ifx\Etoc@nexttoken\label
526 \def\Etoc@next{\expandafter\Etoc@label\@gobble}\else
527 \ifx\Etoc@nexttoken\@sptoken
528 \let\Etoc@next\Etoc@again\else
529 \ifEtoc@tocwithid
530 \def\Etoc@next{\Etoc@localtableofcontents{\c@etoc@tocid}}%
531 \else
532 \ifx\Etoc@nexttoken\ref
533 \def\Etoc@next{\expandafter\Etoc@ref\@gobble}%
534 \else
535 \def\Etoc@next{\Etoc@localtocfalse
536 \global\Etoc@notactivefalse
537 \Etoc@tableofcontents
538 \endgroup

```

16. Implementation

```

539 \ifEtoc@mustclosegroup\endgroup\fi}%
540 \fi
541 \fi\fi\fi\Etoc@next}
placeholder for comments
542 \def\table@fcontents{%
543 \refstepcounter{etoc@tocid}%
544 \Etoc@tocwithidfalse
545 \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
546 \def\localtable@fcontents{%
547 \refstepcounter{etoc@tocid}%
548 \addtocontents{toc}
549 {\string\etoc@startlocaltoc{\arabic{etoc@tocid}}}%
550 \Etoc@tocwithidtrue
551 \futurelet\Etoc@nexttoken\Etoc@t@bleofcontents}
placeholder for comments
552 \newcommand*\etocaftertitlehook{}
553 \newcommand*\etocaftercontentshook{}
554 \renewcommand*\tableofcontents{%
555 \Etoc@openouttoc
556 \Etoc@par
557 \begingroup % closed in \Etoc@t@bleofcontents or \Etoc@localtableofcontents
558 \def\etoc@startlocaltoc##1{\etoc@startlocaltoc{##1}{\c@etoc@tocid}}%
559 \@ifstar
560 {\def\Etoc@aftertitlehook{}\table@fcontents}
561 {\let\Etoc@aftertitlehook\etocaftertitlehook\table@fcontents}}
562 \newcommand*\localtableofcontents{%
563 \Etoc@openouttoc
564 \Etoc@par
565 \begingroup % closed in \Etoc@t@bleofcontents or \Etoc@localtableofcontents
566 \@ifstar
567 {\def\Etoc@aftertitlehook{}\localtable@fcontents}
568 {\let\Etoc@aftertitlehook\etocaftertitlehook\localtable@fcontents}}
placeholder for comments
569 \newcommand\etocsettocstyle[2]{%
570 \def\Etoc@tableofcontents
571 {\ifnum\c@tocdepth>\Etoc@minf
572 \let\Etoc@@next\@firstofone\else
573 \let\Etoc@@next\@gobble
574 \fi
575 \Etoc@@next{#1\ifEtoc@parskip\parskip\z@skip\fi %1.07d
576 \Etoc@aftertitlehook
577 \let\Etoc@savedcontentsline\contentsline
578 \let\contentsline\Etoc@etoccontentsline
579 \Etoc@toctoc
580 \let\Etoc@@next\relax
581 \ifEtoc@tocwithid\else
582 \ifEtoc@localtoc
583 \ifEtoc@notactive
584 \def\Etoc@@next{\Etoc@localtocfalse
585 \global\Etoc@notactivefalse
586 \Etoc@toctoc}%

```

16. Implementation

```

587         \fi\fi\fi
588         \Etoc@@next
589         \etocaftercontentshook
590         #2\@nobreakfalse}}} % 1.07d: \@nobreakfalse moved here

placeholder for comments
591 \newcommand*\etocsetstyle{\Etoc@standardfalse\etoc@setstyle}
592 \long\def\etoc@setstyle#1#2#3#4#5{%
593 \long\expandafter\def
594     \csname Etoc@begin@\csname Etoc@#1@\endcsname\endcsname {#2}%
595 \long\expandafter\def
596     \csname Etoc@prefix@\csname Etoc@#1@\endcsname\endcsname {#3}%
597 \long\expandafter\def
598     \csname Etoc@contents@\csname Etoc@#1@\endcsname\endcsname {#4}%
599 \long\expandafter\def
600     \csname Etoc@end@\csname Etoc@#1@\endcsname\endcsname {#5}}

placeholder for comments
601 \newcommand*\etocfontminustwo {\normalfont \LARGE \bfseries}
602 \newcommand*\etocfontminusone {\normalfont \large \bfseries}
603 \newcommand*\etocfontzero {\normalfont \large \bfseries}
604 \newcommand*\etocfontone {\normalfont \normalsize \bfseries}
605 \newcommand*\etocfonttwo {\normalfont \normalsize}
606 \newcommand*\etocfontthree {\normalfont \footnotesize}

placeholder for comments
607 \newcommand*\etocsepminustwo {4ex \@plus .5ex \@minus .5ex}
608 \newcommand*\etocsepminusone {4ex \@plus .5ex \@minus .5ex}
609 \newcommand*\etocsepzero {2.5ex \@plus .4ex \@minus .4ex}
610 \newcommand*\etocsepone {1.5ex \@plus .3ex \@minus .3ex}
611 %%\newcommand*\etocseptwo{1ex \@plus .15ex \@minus .15ex} % modified in 1.07e
612 \newcommand*\etocseptwo {1.5ex \@plus .1ex \@minus .1ex}
613 \newcommand*\etocseptthree {1.25ex \@plus .05ex \@minus .05ex}

placeholder for comments
614 \newcommand*\etocbaselinespreadminustwo {1}
615 \newcommand*\etocbaselinespreadminusone {1}
616 \newcommand*\etocbaselinespreadzero {1}
617 \newcommand*\etocbaselinespreadone {1}
618 \newcommand*\etocbaselinespreadtwo {1}
619 \newcommand*\etocbaselinespreadthree {.9}

placeholder for comments
620 \newcommand*\etocminustwoleftmargin {1.5em plus 0.5fil}
621 \newcommand*\etocminustworightmargin {1.5em plus -0.5fil}
622 \newcommand*\etocminusoneleftmargin {1em}
623 \newcommand*\etocminusonerightmargin {1em}
624 \newcommand*\etoclineleaders
625     {\hbox{\normalfont\normalsize\hb@xt@2ex {\hss.\hss}}}
626 \newcommand*\etocabbrevpagename {p.~}
627 \newcommand*\etocpartname {\partname}
628 \newcommand*\etocbookname {Book}

placeholder for comments The macro \etocdefaultlines was initially called \etoclines. Now \etoclines just does \Etoc@standardfalse.

```

16. Implementation

```
629 \def\etocdefaultlines{\Etoc@standardfalse
```

Version 1.07e has rewritten entirely the stuff related to penalties and `\addvspace`, as this was not satisfactory in the earlier versions, which were written at a early stage in the development of the package.

```
630 %% 'book' in memoir class:
```

```
631 \etoc@setstyle{@minustwo}
632 {\addpenalty\@M\etocskipfirstprefix}
633 {\addpenalty\@secpenalty}
634 {\begingroup
635   \etocfontminustwo
636   \addvspace{\etocsepminustwo}%
637   \parindent \z@
638   \leftskip \etocminustwoleftmargin
639   \rightskip \etocminustworightmargin
640   \parfillskip \@flushglue
641   \vbox{\etocifnumbered{\etocbookname\enspace\etocnumber:\quad}}{\etocname
642     \baselineskip\etocbaselinespreadminustwo\baselineskip
643     \par}%
644   \addpenalty\@M\addvspace{\etocsepminusone}%
645   \endgroup}
646 {}%
```

```
647 %% 'part':
```

```
648 \etoc@setstyle{@minusone}
649 {\addpenalty\@M\etocskipfirstprefix}
650 {\addpenalty\@secpenalty}
651 {\begingroup
652   \etocfontminusone
653   \addvspace{\etocsepminusone}%
654   \parindent \z@
655   \leftskip \etocminusoneleftmargin
656   \rightskip \etocminusonerightmargin
657   \parfillskip \@flushglue
658   \vbox{\etocifnumbered{\etocpartname\enspace\etocnumber.\quad}}{\etocname
659     \baselineskip\etocbaselinespreadminusone\baselineskip
660     \par}%
661   \addpenalty\@M\addvspace{\etocsepzero}%
662   \endgroup}
663 {}%
```

```
664 %% 'chapter':
```

```
665 \etoc@setstyle{@zero}
666 {\addpenalty\@M\etocskipfirstprefix}
667 {\addpenalty\@itempenalty}
668 {\begingroup
669   \etocfontzero
670   \addvspace{\etocsepzero}%
671   \parindent \z@ \parfillskip \@flushglue
672   \vbox{\etocifnumbered{\etocnumber.\enspace}}{\etocname
673     \baselineskip\etocbaselinespreadzero\baselineskip
674     \par}%
675   \endgroup}
676 {\addpenalty{-\@highpenalty}\addvspace{\etocsepminusone}}%
677 %% 'section':
```

16. Implementation

```

678 \etoc@setstyle{@one}
679 {\addpenalty\@M\etocskipfirstprefix}
680 {\addpenalty\@itempenalty}
681 {\begingroup
682   \etocfontone
683   \addvspace{\etocseppone}%
684   \parindent \z@ \parfillskip \z@
685   \setbox\z@\vbox{\parfillskip\@flushglue
686                     \etocname\par
687                     \setbox\tw@\lastbox
688                     \global\setbox\@ne\hbox{\unhbox\tw@ \ }}%
689   \dimen\z@=\wd\@ne
690   \setbox\z@=\etoclineleaders
691   \advance\dimen\z@\wd\z@
692   \etocifnumbered
693     {\setbox\tw@\hbox{\etocnumber, \etocabbrevpagename\etocpage}}
694     {\setbox\tw@\hbox{\etocabbrevpagename\etocpage}}%
695   \advance\dimen\z@\wd\tw@
696   \ifdim\dimen\z@ < \linewidth
697     \vbox{\etocname~%
698           \leaders\box\z@\hfil\box\tw@
699           \baselineskip\etocbaselinespreadone\baselineskip
700           \par}
701   \else
702     \vbox{\etocname~%
703           \leaders\copy\z@\hfil\break
704           \hbox{} \leaders\box\z@\hfil\box\tw@
705           \baselineskip\etocbaselinespreadone\baselineskip
706           \par}
707   \fi
708   \endgroup}
709 {\addpenalty\@secpenalty\addvspace{\etocsepzero}}%
710 %% 'subsection':
711 \etoc@setstyle{@two}
712 {\addpenalty\@medpenalty\etocskipfirstprefix}
713 {\addpenalty\@itempenalty}
714 {\begingroup
715   \etocfonttwo
716   \addvspace{\etocseptwo}%
717   \parindent \z@ \parfillskip \z@
718   \setbox\z@\vbox{\parfillskip\@flushglue
719                     \etocname\par\setbox\tw@\lastbox
720                     \global\setbox\@ne\hbox{\unhbox\tw@}}%
721   \dimen\z@=\wd\@ne
722   \setbox\z@=\etoclineleaders
723   \advance\dimen\z@\wd\z@
724   \etocifnumbered
725     {\setbox\tw@\hbox{\etocnumber, \etocabbrevpagename\etocpage}}
726     {\setbox\tw@\hbox{\etocabbrevpagename\etocpage}}%
727   \advance\dimen\z@\wd\tw@
728   \ifdim\dimen\z@ < \linewidth
729     \vbox{\etocname~%

```

16. Implementation

```

730          \leaders\box\z@\hfil\box\tw@
731          \baselineskip\etocbaselinespreadtwo\baselineskip
732          \par}
733 \else
734   \vbox{\etocname~%
735         \leaders\copy\z@\hfil\break
736         \hbox{\leaders\box\z@\hfil\box\tw@
737               \baselineskip\etocbaselinespreadtwo\baselineskip
738               \par}
739   \fi
740 \endgroup}
741 {\addpenalty\@secpenalty\addvspace{\etocseppone}}%
742 %% 'subsubsection':
743 \etoc@setstyle{@three}
744 {\addpenalty\@M
745  \etocfontthree
746  \vspace{\etocsepthree}%
747  \noindent
748  \etocskipfirstprefix}
749 {\allowbreak\,--\,}
750 {\etocname}
751 {.\hfil
752  \begingroup
753   \baselineskip\etocbaselinespreadthree\baselineskip
754   \par
755   \endgroup
756   \addpenalty{-\@highpenalty}}%

placeholder for comments
757 \etoc@setstyle{@four}{\{\}\{\}\{\}\{\}\}%
758 \etoc@setstyle{@five}{\{\}\{\}\{\}\{\}\{\}\}%
759 }

placeholder for comments
760 \newcommand*\etocabovetocskip{3.5ex \@plus 1ex \@minus .2ex}
761 \newcommand*\etocbelowtocskip{3.5ex \@plus 1ex \@minus .2ex}
762 \newcommand*\etoccolumnsep{2em}
763 \newcommand*\etocmulticolsep{0ex}
764 \newcommand*\etocmulticolpretolerance{-1}
765 \newcommand*\etocmulticoltolerance{200}
766 \newcommand*\etocdefaultnbcol{2}
767 \newcommand*\etocinnertopsep{2ex}

placeholder for comments
768 \newcommand\etocmulticolstyle[2][\etocdefaultnbcol]{%
769 \etocsettocstyle
770  {\let\etocoldpar\par
771   \addvspace{\etocabovetocskip}%
772   \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
773   \else \let\Etoc@next\@secondoftwo\fi
774   \Etoc@next}%
775  \multicolpretolerance\etocmulticolpretolerance
776  \multicoltolerance\etocmulticoltolerance
777  \setlength{\columnsep}{\etoccolumnsep}%

```

16. Implementation

```

778 \setlength{\multicolsep}{\etocmulticolsep}%
779 \begin{multicols}{#1}[#2\etocoldpar\addvspace{\etocinnertopsep}]
780 % 2013/01/29: erroneous \etocsepminusone at last replaced by \etocinnertopsep
781 % and definition of \etocoldpar added as multicols chokes on \par as part of #2
782 {#2\par\addvspace{\etocinnertopsep}%
783 \pretolerance\etocmulticolpretolerance
784 \tolerance\etocmulticoltolerance}}
785 {\ifnum #1>\@ne\let\Etoc@next\@firstofone
786 \else \let\Etoc@next\@gobble\fi
787 \Etoc@next\end{multicols}}%
788 \addvspace{\etocbelowtocskip}}

placeholder for comments

789 \newcommand*\etocinnerbottomsep{3.5ex}
790 \newcommand*\etocinnerleftsep{2em}
791 \newcommand*\etocinnerrightsep{2em}
792 \newcommand*\etoctoprule{\hrule}
793 \newcommand*\etocleftrule{\vrule}
794 \newcommand*\etocrightrule{\vrule}
795 \newcommand*\etocbottomrule{\hrule}
796 \newcommand*\etoctoprulecolorcmd{\relax}
797 \newcommand*\etocbottomrulecolorcmd{\relax}
798 \newcommand*\etocleftrulecolorcmd{\relax}
799 \newcommand*\etocrightrulecolorcmd{\relax}

placeholder for comments

800 \def\etoc@ruledheading #1{%
801 \hb@xt@\linewidth{\color@begingroup
802 \hss #1\hss\hskip-\linewidth
803 \etoctoprulecolorcmd\leaders\etoctoprule\hss
804 \phantom{#1}%
805 \leaders\etoctoprule\hss\color@endgroup}%
806 \nointerlineskip\vskip\etocinnertopsep}

placeholder for comments

807 \newcommand*\etocruledstyle[2][\etocdefaultnbcol]{%
808 \etocsettocstyle
809 {\addvspace{\etocabovetocskip}%
810 \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
811 \else \let\Etoc@next\@secondoftwo\fi
812 \Etoc@next
813 {\multicolpretolerance\etocmulticolpretolerance
814 \multicoltolerance\etocmulticoltolerance
815 \setlength{\columnsep}{\etoccolumnsep}%
816 \setlength{\multicolsep}{\etocmulticolsep}%
817 \begin{multicols}{#1}[\etoc@ruledheading{#2}]}
818 {\etoc@ruledheading{#2}\nobreak
819 \pretolerance\etocmulticolpretolerance
820 \tolerance\etocmulticoltolerance}}
821 {\ifnum #1>\@ne\let\Etoc@next\@firstofone
822 \else \let\Etoc@next\@gobble\fi
823 \Etoc@next\end{multicols}}%
824 \addvspace{\etocbelowtocskip}}

```

16. Implementation

placeholder for comments

```

825 \newcommand\etocframedmphook{\relax}
826 \newcommand*\etocbkgcolorcmd{\relax}
827 \def\Etoc@relax{\relax}
828 \newbox\etoc@framed@titlebox
829 \newbox\etoc@framed@contentsbox
830 \newcommand*\etocframedstyle[2][\etocdefaultnbcoll]{%
831 \etocsettocstyle{%
832   \addvspace{\etocabovetocskip}%
833   \sbox\z@{\#2}%
834   \dimen\z@\dp\z@
835   \ifdim\wd\z@<\linewidth \dp\z@\z@ \else \dimen\z@\z@ \fi
836 \setbox\etoc@framed@titlebox=\hb@xt@\linewidth{\color@begingroup
837   \hss
838   \ifx\etocbkgcolorcmd\Etoc@relax\else
839     \sbox\tw@{\color{white}%
840       \vrule\@width\wd\z@\@height\ht\z@\@depth\dimen\z@}%
841     \ifdim\wd\z@<\linewidth \dp\tw@\z@\fi
842     \box\tw@
843     \hskip-\wd\z@
844   \fi
845   \copy\z@
846   \hss
847   \hskip-\linewidth
848   \etoctoprulecolorcmd\leaders\etoctoprule\hss%
849   \hskip\wd\z@
850   \etoctoprulecolorcmd\leaders\etoctoprule\hss\color@endgroup}%
851 \setbox\z@\hbox{\etocleftrule\etocrightrule}%
852 \dimen\tw@\linewidth\advance\dimen\tw@-\wd\z@
853 \advance\dimen\tw@-\etocinnerleftsep
854 \advance\dimen\tw@-\etocinnerrightsep
855 \setbox\etoc@framed@contentsbox=\vbox\bgroup
856 \hsize\dimen\tw@
857 \kern\dimen\z@
858 \vskip\etocinnertopsep
859 \hbox\bgroup
860 \begin{minipage}{\hsize}%
861 \etocframedmphook
862 \ifnum #1>\@ne\let\Etoc@next\@firstoftwo
863 \else \let\Etoc@next\@secondoftwo\fi
864 \Etoc@next
865 {\multicolpretolerance\etocmulticolpretolerance
866 \multicoltolerance\etocmulticoltolerance
867 \setlength{\columnsep}{\etoccolumnsep}%
868 \setlength{\multicolsep}{\etocmulticolsep}%
869 \begin{multicols}{#1}}
870 {\pretolerance\etocmulticolpretolerance
871 \tolerance\etocmulticoltolerance}}
872 {\ifnum #1>\@ne\let\Etoc@next\@firstofone
873 \else \let\Etoc@next\@gobble\fi
874 \Etoc@next\end{multicols}\unskip}%
875 \end{minipage}%

```

16. Implementation

```

876 \egroup
877 \vskip\etocinnerbottomsep
878 \egroup
879 \vbox{\hsize\linewidth
880   \ifx\etocbkgcolorcmd\Etoc@relax\else
881     \kern\ht\etoc@framed@titlebox
882     \kern\dp\etoc@framed@titlebox
883     \hb@xt@\linewidth{\color@begingroup
884       \etocleftrulecolorcmd\etocleftrule
885       \etocbkgcolorcmd
886       \leaders\vrule
887         \@height\ht\etoc@framed@contentsbox
888         \@depth\dp\etoc@framed@contentsbox
889       \hss
890       \etocrightrulecolorcmd\etocrightrule
891       \color@endgroup}\nointerlineskip
892       \vskip-\dp\etoc@framed@contentsbox
893       \vskip-\ht\etoc@framed@contentsbox
894       \vskip-\dp\etoc@framed@titlebox
895       \vskip-\ht\etoc@framed@titlebox
896     \fi
897   \box\etoc@framed@titlebox\nointerlineskip
898   \hb@xt@\linewidth{\color@begingroup
899     {\etocleftrulecolorcmd\etocleftrule}%
900     \hss\box\etoc@framed@contentsbox\hss
901     \etocrightrulecolorcmd\etocrightrule\color@endgroup}
902   \nointerlineskip
903   \vskip\ht\etoc@framed@contentsbox
904   \vskip\dp\etoc@framed@contentsbox
905   \hb@xt@\linewidth{\color@begingroup\etocbottomrulecolorcmd
906     \leaders\etocbottomrule\hss\color@endgroup}}
907   \addvspace{\etocbelowtocskip}}

placeholder for comments

908 \newcommand\etoc@multicoltoc[2][\etocdefaultnbcol]{%
909   \etocmulticolstyle[#1]{#2}%
910   \tableofcontents}
911 \newcommand\etoc@multicoltoctoci[2][\etocdefaultnbcol]{%
912   \etocmulticolstyle[#1]{#2}%
913   \tableofcontents*}
914 \newcommand\etoc@local@multicoltoc[2][\etocdefaultnbcol]{%
915   \etocmulticolstyle[#1]{#2}%
916   \localtableofcontents}
917 \newcommand\etoc@local@multicoltoctoci[2][\etocdefaultnbcol]{%
918   \etocmulticolstyle[#1]{#2}%
919   \localtableofcontents*}

placeholder for comments

920 \newcommand*\etoc@ruledtoc[2][\etocdefaultnbcol]{%
921   \etocruledstyle[#1]{#2}%
922   \tableofcontents}
923 \newcommand*\etoc@ruledtoctoci[2][\etocdefaultnbcol]{%
924   \etocruledstyle[#1]{#2}%

```

16. Implementation

```
925 \tableofcontents*}
926 \newcommand*\etoc@local@ruledtoc[2][\etocdefaultnbcol]{%
927 \etocruledstyle[#1]{#2}%
928 \localtableofcontents}
929 \newcommand*\etoc@local@ruledtoci[2][\etocdefaultnbcol]{%
930 \etocruledstyle[#1]{#2}%
931 \localtableofcontents*}

placeholder for comments

932 \newcommand*\etoc@framedtoc[2][\etocdefaultnbcol]{%
933 \etocframedstyle[#1]{#2}%
934 \tableofcontents}
935 \newcommand*\etoc@framedtoci[2][\etocdefaultnbcol]{%
936 \etocframedstyle[#1]{#2}%
937 \tableofcontents*}
938 \newcommand*\etoc@local@framedtoc[2][\etocdefaultnbcol]{%
939 \etocframedstyle[#1]{#2}%
940 \localtableofcontents}
941 \newcommand*\etoc@local@framedtoci[2][\etocdefaultnbcol]{%
942 \etocframedstyle[#1]{#2}%
943 \localtableofcontents*}

placeholder for comments

944 \def\etocmulticol{\begingroup
945 \Etoc@mustclosegrouptrue
946 \@ifstar
947 {\etoc@multicoltoctoci}
948 {\etoc@multicoltoctoc}}
949 \def\etocruled{\begingroup
950 \Etoc@mustclosegrouptrue
951 \@ifstar
952 {\etoc@ruledtoci}
953 {\etoc@ruledtoc}}
954 \def\etocframed{\begingroup
955 \Etoc@mustclosegrouptrue
956 \@ifstar
957 {\etoc@framedtoci}
958 {\etoc@framedtoc}}
959 \def\etoclocalmulticol{\begingroup
960 \Etoc@mustclosegrouptrue
961 \@ifstar
962 {\etoc@local@multicoltoctoci}
963 {\etoc@local@multicoltoctoc}}
964 \def\etoclocalruled{\begingroup
965 \Etoc@mustclosegrouptrue
966 \@ifstar
967 {\etoc@local@ruledtoci}
968 {\etoc@local@ruledtoc}}
969 \def\etoclocalframed{\begingroup
970 \Etoc@mustclosegrouptrue
971 \@ifstar
972 {\etoc@local@framedtoci}
973 {\etoc@local@framedtoc}}
```

16. Implementation

```

placeholder for comments
974 \def\etocarticlestyle{%
975   \etocsettocstyle
976   {\section *{\contentsname
977     \@mkboth {\MakeUppercase \contentsname}
978     {\MakeUppercase \contentsname}}}
979   {}}
980 \def\etocarticlestylenomarks{%
981   \etocsettocstyle
982   {\section *{\contentsname}}
983   {}}
placeholder for comments
984 \def\etocbookstyle{%
985   \etocsettocstyle
986   {\if@twocolumn \@restonecoltrue \onecolumn \else \@restonecolfalse \fi
987     \chapter *{\contentsname
988       \@mkboth {\MakeUppercase \contentsname}
989       {\MakeUppercase \contentsname}}}
990   {\if@restonecol \twocolumn \fi}}
991 \def\etocbookstylenomarks{%
992   \etocsettocstyle
993   {\if@twocolumn \@restonecoltrue \onecolumn \else \@restonecolfalse \fi
994     \chapter *{\contentsname}}
995   {\if@restonecol \twocolumn \fi}}
996 \let\etocreportstyle\etocbookstyle
997 \let\etocreportstylenomarks\etocbookstylenomarks
998 \def\etocmemoirtoc\tocfmt #1#2{%
999   \def\Etoc@addsuitablecontentsline{\addcontentsline {toc}{#1}{#2}}%
1000   \renewcommand*\etocaftertitlehook{%
1001     \ifmem@em@starred@listof
1002     \else\phantomsection\aftergroup\Etoc@addsuitablecontentsline\fi}}
1003 \def\etocmemoirstyle{%
1004   \etocsettocstyle
1005   {\ensureonecol \par \begingroup \@nameuse {@tocmaketitle}
1006     \Etoc@aftertitlehook\let\Etoc@aftertitlehook\relax
1007     \parskip \cftparskip \@nameuse {cfttocbeforelisthook}}
1008   {\@nameuse {cfttocafterlisthook}\endgroup\restorefromonecol}}
placeholder for comments
1009 \def\etocscrartclstyle{%
1010   \etocsettocstyle
1011   {\let\if@dynlist\if@tocleft
1012     \iftocfeature {toc}{onecolumn}
1013     {\iftocfeature {toc}{leveldown}
1014       {}}
1015     {\if@twocolumn \aftergroup \twocolumn \onecolumn \fi }}
1016   {}}%
1017   \tocbasic@listhead {\listoftocname}%
1018   \begingroup \expandafter \expandafter \expandafter
1019   \endgroup \expandafter
1020   \ifx
1021   \csname microtypesetup\endcsname \relax

```

16. Implementation

```

1022     \else
1023         \iftocfeature {toc}{noprotrusion}{}
1024         {\microtypesetup {protrusion=false}%
1025         \PackageInfo {tocbasic}%
1026         {character protrusion at toc deactivated}}%
1027     \fi
1028     \setlength {\parskip }{\z@ }%
1029     \setlength {\parindent }{\z@ }%
1030     \setlength {\parfillskip }{\z@ \@plus 1fil}%
1031     \csname tocbasic@@before@hook\endcsname
1032     \csname tb@toc@before@hook\endcsname}
1033     {\csname tb@toc@after@hook\endcsname
1034     \csname tocbasic@@after@hook\endcsname}}
1035 \let\etocscrbookstyle\etocscrartclstyle
1036 \let\etocscrreprtstyle\etocscrartclstyle

placeholder for comments
1037 \newcommand*\etocstandarddisplaystyle{\etocarticlestyle}
1038 \newcommand*\etocmarkboth[1]{%
1039     \@mkboth{\MakeUppercase{#1}}{\MakeUppercase{#1}}}
1040 \newcommand*\etocmarkbothnouc[1]{\@mkboth{#1}{#1}}
1041 \newcommand\etocstyle[3][section]{\etocmulticolstyle[#2]%
1042     {\csname #1\endcsname *{#3}}}
1043 \newcommand\etocstylewithmarks[4][section]{\etocmulticolstyle[#2]%
1044     {\csname #1\endcsname *{#3}\etocmarkboth{#4}}}}
1045 \newcommand\etocstylewithmarksnouc[4][section]{\etocmulticolstyle[#2]%
1046     {\csname #1\endcsname *{#3}\etocmarkbothnouc{#4}}}}

placeholder for comments
1047 \def\Etoc@redefetocstyle#1{%
1048     \renewcommand\etocstylewithmarks[4][#1]
1049     {\etocmulticolstyle[##2]%
1050         {\csname ##1\endcsname *{##3}\etocmarkboth{##4}}}}
1051     \renewcommand\etocstylewithmarksnouc[4][#1]
1052     {\etocmulticolstyle[##2]%
1053         {\csname ##1\endcsname *{##3}\etocmarkbothnouc{##4}}}}
1054     \renewcommand\etocstyle[3][#1]{%
1055         \etocmulticolstyle[##2]{\csname ##1\endcsname *{##3}}}}
1056 \@ifclassloaded{scrartcl}
1057     {\renewcommand*\etocstandarddisplaystyle{\etocscrartclstyle}}{}
1058 \@ifclassloaded{book}
1059     {\renewcommand*\etocfontone{\normalfont\normalsize}
1060     \renewcommand*\etocstandarddisplaystyle{\etocbookstyle}
1061     \Etoc@redefetocstyle{chapter}}{}
1062 \@ifclassloaded{report}
1063     {\renewcommand*\etocfontone{\normalfont\normalsize}
1064     \renewcommand*\etocstandarddisplaystyle{\etocreportstyle}
1065     \Etoc@redefetocstyle{chapter}}{}
1066 \@ifclassloaded{scrbook}
1067     {\renewcommand*\etocfontone{\normalfont\normalsize}
1068     \renewcommand*\etocstandarddisplaystyle{\etocscrbookstyle}
1069     \Etoc@redefetocstyle{chapter}}{}
1070 \@ifclassloaded{scrreprt}

```

16. Implementation

```
1071 {\renewcommand*\etocfontone{\normalfont\normalsize}
1072 \renewcommand*\etocstandarddisplaystyle{\etocscrreprtstyle}
1073 \Etoc@redefetocstyle{chapter}}{}
1074 \@ifclassloaded{memoir}
1075 {\renewcommand*\etocfontone{\normalfont\normalsize}
1076 \etocmemoirtotocfmt{chapter}{\contentsname}%
1077 \renewcommand*\etocstandarddisplaystyle{\etocmemoirstyle}
1078 \Etoc@redefetocstyle{chapter}}{}

placeholder for comments
1079 \def\Etoc@addtocontents #1#2%
1080 {\ifEtoc@hyperref
1081 \addtocontents {toc}{\protect\contentsline
1082 {#1}{#2}%
1083 {\thepage }{\@currentHref }}%
1084 \else
1085 \addtocontents {toc}{\protect\contentsline
1086 {#1}{#2}{\thepage }}%
1087 \fi}
1088 \def\Etoc@addcontentsline@ #1#2#3%
1089 {\@namedef{toclevel@#1}{#3}%
1090 \addcontentsline {toc}{#1}{#2}}
1091 \DeclareRobustCommand*\etoccontentsline{
1092 {\@ifstar{\Etoc@addcontentsline@}{\Etoc@addtocontents}}
1093 \newcommand*\etocstandardlines{\Etoc@standardtrue}
1094 \newcommand*\etoclines{\Etoc@standardfalse} % 1.07b
1095 \etocdefaultlines % for initialization
1096 \etocstandardlines
1097 \etocstandarddisplaystyle
1098 \endinput
```