

tocloft 宏包^{*}

作者: Peter Wilson, Herries Press

维护: Will Robertson

翻译: 黄旭华[†]

2020 / 1 / 10

摘要

`tocloft` 宏包提供了控制目录 (table of Contents)、图目录 (list of Figures) 和表目录 (list of Tables) 的排版设计的方法。可以定义新类型的“... 目录”。

该宏包已使用 `tocbibind`、`minitoc`、`ccaption`、`subfigure`、`float`、`fncychap` 和 `hyperref` 等宏包进行了测试。

^{*}此文件 (`tocloft.dtx`) 的版本号为 v2.3j, 最后修订日期为 2020/01/10。

[†]赣南医学院第一附属医院神经科的一名 L^AT_EX 业余爱好者。

目 录

摘要	1
1 介绍	3
1.1 L ^A T _E X 的方法	3
2 tocloft 宏包	9
2.1 宏包选项	9
2.2 更改标题	9
2.3 排版条目	10
2.4 新列表 ...	16
2.5 实验性程序	20
2.6 与其他宏包一起使用	22
3 宏包代码	24
3.1 对 subfigure 宏包的支持	51
3.2 新列表 ...	54
3.3 切换页码	61
3.4 实验性工具	64
索引	70

图形目录

1 目录、图目录、表目录等的条目布局	4
--------------------	---

表格目录

1 缩进和编号宽度	5
-----------	---

1 介绍

在标准类 (standard classes) 中, 目录 (Table of Contents, ToC)、图目录 (List of Figures, LoF) 和表目录 (List of Tables, LoT) 的排版设计 (typographic design) 是固定的, 或者更准确地说, 它被隐藏在类定义 (class definitions) 中。tocloft 宏包为作者提供了修改设计以满足特定文档需求的句柄 (handles)。

宏包的元素 (elements) 是作为 ISO 标准 [Wil96b] 排版的类 (class) 和包 (package) 的一部分开发的。本手册根据 L^AT_EX DOCSTRIP 实用程序 (utility) 的约定 (conventions) 排版, 该实用程序可自动提取 L^AT_EX 宏源文件 (macro source files) [GMS94]。

第 2 节描述了该宏包的使用。该宏包源代码的注释见第 3 节。

该宏包已至少与 tocbibind 宏包 [Wil00]、minitoc 宏包 [Dru99]、ccaption 宏包 [Wil01]、subfigure 宏包 [Coc95] (2.0 版和 2.1 版)、algorithm 宏包 [Wil96a] (依次称为 float 宏包 [Lin95]) 和 fncychap 宏包 [Lin97] 结合进行了测试。它也适用于 hyperref 宏包。请给我任何关于您认为该宏包可以如何改进的意见, 或者关于你如何使用它的任何有趣的例子。¹

1.1 L^AT_EX 的方法

这是对 L^AT_EX 如何处理目录 (Table of Contents, ToC) 的一般描述 (general description)。由于图目录 (List of Figures) 和表目录 (List of Tables) 的处理类似, 我将在不失一般性的情况下讨论 ToC。

`\addcontentsline` 如果文档 (document) 包含 `\tableofcontents` 命令, L^AT_EX 将生成一个 `.toc` 文件。分节命令 (sectioning commands)² 通过调用 L^AT_EX 的

`\addcontentsline{<file>}{<kind>}{<title>}`

命令将条目 (entries) 放入 `.toc` 文件中, 这里的 `<file>` 是文件扩展名 (例如, `toc`), `<kind>` 是条目类型 (例如, `section` 或 `subsection`), `<title>` 是 (已编号的 [numbered]) 标题文本 (title text)。在有数字的情况下, `<title>` 参数以

¹感谢卡斯珀 (Kasper, kgb@dkik.dk)、罗兰 (Rowland, rebecca@astrid.u-net.com)、约翰·福斯特 (John Foster, john@isjf.demon.co.uk)、利·内夫 (Lee Nave, nave@math.washington.edu)、安德鲁·瑟伯 (Andrew Thurber, athurber@emba.uvm.edu) 等的建议。

²对于图形 (figures) 和表格 (tables), 分节命令则是 `\caption` 命令, 该命令填充 (populates) `.lof` 和 `.lot` 文件。

`{\numberline{number} title-text}` 的形式给出。

注意: `hyperref` 宏包不喜欢作者使用 `\addcontentsline`。要使其与 `hyperref` 一起正常工作, 通常必须在 `\addcontentsline` 之前立即放置 `\phantomsection` (在 `hyperref` 宏包中定义的宏)。

`\contentsline` `\addcontentsline` 命令以

`\contentsline{<kind>}{<title>}{<page>}`

的形式向给定文件写入一个条目 (entry), 这里的 `<page>` 是页码 (page number)。对于每个 `<kind>`, L^AT_EX 提供一个命令 `\l@kind{<title>}{<page>}`, 它执行 `\contentsline` 条目 (entry) 的实际排版 (actual typesetting)。

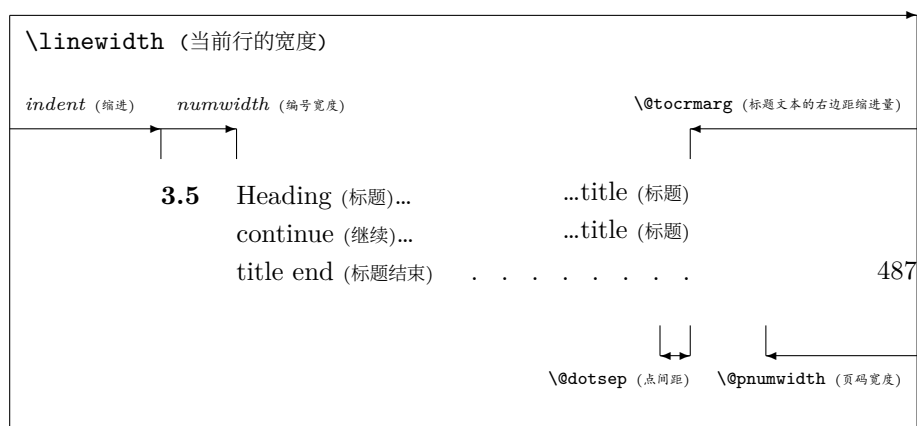


图 1: 目录 (ToC)、图目录 (LoF)、表目录 (LoT) 等的条目布局

排版条目 (typeset entry) 的总体布局 (general layout) 如图 1 所示。排版中使用了三个内部 (internal) L^AT_EX 命令。页码 (page number) 在一个宽度为 `\@pnumwidth` 的盒子右对齐排版 (typeset flushright), 盒子位于右边距 (righthand margin)。如果页码太长而无法放入盒子中, 则页码将突出到右边距 (righthand margin)。标题文本 (title text) 从右边距缩进了 `\@tocrmarg` 给出的量 (amount)。请注意, `\@tocrmarg` 应大于 `\@pnumwidth`。有些条目是在标题文本末尾和右边距缩进 (righthand margin indentation) 之间用点引线 (dotted leader) 排版的。引线 `\@dotsep` 中的点 (dots) 之间的距离 (以数学单位³表示) 由 `\@dotsep` 的值给出。在标准类 (standard classes) 中, 目录 (ToC)、图目录 (LoF)、表目录 (LoT) 等使用相同的值。

³这里是 18mu 到 1em。

这些内部命令 (internal commands) 的标准值 (standard values) 为:

- `\@pnumwidth = 1.55em`
- `\@tocrmarg = 2.55em`
- `\@dotsep = 4.5`

尽管前两个值看起来都是长度 (lengths), 但是可以通过使用 `\renewcommand` 命令来更改这些值。

点引线 (dotted leaders) 不适用于 Part 和 Chapter 的目录条目 (ToC entries) (也不适用于 article 类及其派生物 [derivatives] 中的 Section 条目)。

`\numberline` 每个 `\l@kind` 宏负责设置从左边距 (lefthand margin) 开始的常规缩进 (general *indent*) 和编号宽度 (*numwidth*)。 `\numberline{<number>}` 宏负责在一个宽度为 *numwidth* 的盒子中以左对齐的 (flushleft) 方式排版数字。如果数字对于盒子来说太长, 那么它将突出到 (protrude into) 标题文本 (title text) 中。标题文本从左边距 (lefthand margin) 缩进 (*indent + numwidth*)。也就是说, 标题文本是按宽度为 (`\linewidth - indent - numwidth - \@tocrmarg`) 的块 (block) 来排版的。

表 1: 缩进 (indents) 和编号宽度 (numwidths)(单位: em)

Entry	Level	Chaptered		Otherwise	
		indent	numwidth	indent	numwidth
part	-1	0	—	0	—
chapter	0	0	1.5		
section	1	1.5	2.3	0	1.5
subsection	2	3.8	3.2	1.5	2.3
subsubsection	3	7.0	4.1	3.8	3.2
paragraph	4	10.0	5.0	7.0	4.1
subparagraph	5	12.0	6.0	10.0	5.0
figure/table	(1)	1.5	2.3	1.5	2.3

上面的表 1 列出了缩进 (*indent*) 和编号宽度 (*numwidth*) 的标准值 (standard values)。对于部分 (part) 没有明确的编号宽度 (*numwidth*); 而是在编号 (number) 和标题文本 (title text) 之间留出 1em 的间隙 (gap)。请注意, 对于分节命令

(sectioning command), 值取决于文档类 (document class) 是否提供 `\chapter` 命令。此外, 这让我有些惊讶, 表格 (table) 和图形 (figure) 的条目 (entries) 都是缩进的。

`\@dottedtocline` 大多数 `\l@kind` 命令是根据 `\@dottedtocline` 命令来定义的。此命令有三个参数:

`\@dottedtocline{<secllevel>}{<indent>}{<numwidth>}`.

例如, `\l@section` 命令的一个定义是:

```
\newcommand*{\l@section}{\@dottedtocline{1}{1.5em}{2.3em}}
```

有必要更改条目的默认排版, 那么通常有必要更改这些定义 (但 `tocloft` 包提供了一些句柄 [handles], 可以轻松地更改内容, 而无需知道 L^AT_EX 内部组件)。

您可以使用 `\addcontentsline` 命令向文件中添加 `\contentsline` 命令。

`\addtocontents` L^AT_EX 还提供了 `\addtocontents{<file>}{<text>}` 命令, 该命令将把 `<text>` 插入 `<file>`。您可以使用它向文件中添加额外的 (extra) 文本和/或宏, 以便在文件由 `\tableofcontents` 命令 (或任何其他用于 `<file>` 处理的命令, 例如处理 `.lot` 文件的 `\listoftables` 命令) 排版时进行处理。

当 `\addcontentsline` 和 `\addtocontents` 将其参数写入文件时, 参数中使用的任何脆弱命令 (fragile commands) 都必须受到 `\protected`。

您可以在不使用任何宏包的情况下对 ToC 等布局 (layout) 进行某些调整。例如:

- 如果您的页码 (page numbers) 突出到右边空白处 (righthand margin), 则可以进行如下设置:

```
\renewcommand{\@pnumwidth}{3em} % 设置“页码宽度”为 3em
\renewcommand{\@tocrmarg}{4em} % 设置“标题文本的右边距缩进量”为 4em
```

以使长度 (lengths) 适合您文档。

- 为了在目录 (ToC) 等中有 (节 [sectional]) 标题 (titles), 右对齐并不加连字符 (hyphenation) 排版, 则可以:

```
\renewcommand{\@tocrmarg}{2.55em plus1fil}
```

这里的值 `2.55em` 可以根据您需要的边距空间 (margin space) 进行更改。

- 引线中的点 (dots in the leaders) 可以通过将 `\@dotsep` 增加到一个大值来消除:

```
\renewcommand{\@dotsep}{10000}
```

- 在您的目录 (ToC) 和图形目录 (LoF) 中如果想要有点引线 (dotted leaders), 但在表格目录 (LoT) 中不要有点引线, 则可以:

```
...
\tableofcontents
\makeatletter \renewcommand{\@dotsep}{10000} \makeatother
\listoftables
\makeatletter \renewcommand{\@dotsep}{4.5} \makeatother
\listoffigures
...
```

对于此文档, 我使用此方法将图形目录 (LoF) 的点间距 (dot spacing) 与目录 (ToC) 的点间距加倍。如您所见, 所有点引线 (dot leaders) 具有相同的间距 (spacing) 要好得多。

- 要在部分条目 (entry for a Part) 的下面添加贯穿目录 (ToC) 整个宽度的水平线 (horizontal line), 请执行以下操作:

```
\part{Part title}
\addtocontents{toc}{\protect\mbox{}\protect\hrulefill\par}
```

请注意, 当 `\addtocontents` 和 `\addcontentsline` 都将它们的参数写入文件时, 这意味着它们参数中的任何脆弱命令 (*fragile* commands) 都必须通过在每个脆弱命令前面加上 `\protect` 来保护。上述示例的结果将是 `.toc` 文件中的以下两行 (假设它是第二部分, 位于第 34 页):

```
\contentsline {part}{II\hspace {1em}Part title}{34}
\mbox {} \hrulefill \par
```

如果未使用 `\protects`, 则第二行将改为:

```
\unhbox \voidb@x \hbox {} \unhbox \voidb@x \leaders \hrule \hfill \kern \z@ \par
```

您可能会在目录 (ToC) 中得到不希望的分页符 (page breaks)。例如, 您可能有一个很长的多行节标题 (long multiline section title), 并且在 ToC 中, 行与行之间有一个分页符。在文档稳定 (stable) 之后, 您可以在文档正文 (body) 的适当位置使用 `\addtocontents` 来调整 ToC 中的分页。例如:

- `\addtocontents{toc}{\protect\newpage}` 强制分页。
- `\addtocontents{toc}{\protect\enlargethispage{2\baselineskip}}` 使页面变长。
- `\addtocontents{toc}{\protect\needspace{2\baselineskip}}` 指定如果页面上没有两条基线 (baselines) 的垂直空间 (vertical space), 则开始一个新页面 (`needspace` 宏包中定义了 `\needspace` 宏)。

请记住, 如果要修改任何包含 @ 符号 (sign) 的命令, 则必须在 `.sty` 文件中执行, 或者如果在文档本身中, 必须用 `\makeatletter` 和 `\makeatother` 包围。例如, 如果你在文档的前言 (preamble) 中修改 `\@dotsep`, 您必须这样做:

```
\makeatletter
\renewcommand{\@dotsep}{9.0}
\makeatother
```


2 tocloft 宏包

tocloft 包提供了指定目录 (Table of Contents, ToC)、图形目录 (List of Figures, LoF) 和表格目录 (List of Tables, LoT) 的排版方法。

`\tableofcontents` ToC、LoF 和 LoT 按照正常的 (normal) L^AT_EX 打印在文档中调用这些命令的位置。然而, 标准的 (standard) L^AT_EX 行为与 tocloft 宏包的行为有一个区别。`\listoffigures` 在具有 `\chapter` 标题 (headings) 的标准 L^AT_EX 类中, ToC、LoF 和 LoT 都显示在新页面上。使用 tocloft 宏包, 它们不必开始新的页面; 如果您想让它们出现在新页面上, 您必须事先发出适当的命令, 例如:

```
...
\clearpage
\tableofcontents
\clearpage
\listoftables
...
```

ToC、LoF 和/或 LoT 的 `\thispagestyle` 页面样式 (page style)

`\tocloftpagestyle` 由 `\tocloftpagestyle{<style>}` 命令 设置, 其中 `<style>` 是可用的页面样式之一。该宏包最初设置为 `\tocloftpagestyle{plain}`。

2.1 宏包选项

该宏包具有以下选项:

`subfigure` 只有当 tocloft 和 subfigure 包一起使用时, 才需要此选项。这两个包可以按任何顺序指定。

`titles` `titles` 选项使用默认的 L^AT_EX 方法对 ToC、LoF 和 LoT 的标题 (titles) 进行排版。这可能是有用的, 例如, 当 tocloft 和 fncychap 宏包一起使用时, “花哨的 (fancy)” 章样式 (chapter styles) 应该用于 ToC 等标题。

如果您使用 `titles` 选项, 可以忽略下一节, 并继续阅读第 2.3 节。

2.2 更改标题

提供了用于控制标题外观 (appearance of the titles) 的命令。按照 L^AT_EX 自定义 (custom), 标题文本 (title texts) 是 `\contentsname`、`\listfigurename` 和 `\listtablename` 命令的值。

为 ToC、LoF 和 LoT 标题排版控件 (title typesetting control) 提供了类似的命令集 (sets of commands)。为了方便 (当然是我的, 也希望是你的), 在下面的描述中, 我将使用 Z 来代表 “toc” 或 “lof” 或 “lot”。例如, `\cftmarkZ` 代表 `\cftmarktoc` 或 `\cftmarklof` 或 `\cftmarklot`。

`\cftmarkZ` 这些宏 (macros) 设置了 ToC、LoF 和 LoT 页面上的短标题 (running heads)。你可能不需要改变这些。

`\cftbeforeZtitleskip` 这些长度控制 (lengths control) 标题前后的垂直间距 (vertical spacing)。您
`\cftafterZtitleskip` 可以使用 `\setlength` 更改它们的默认值。

`\cftZtitlefont` 用于排版 ToC 标题的代码如下所示:

`\cftafterZtitle` `{\cfttoctitlefont \contentsname}{\cftaftertoctitle}\par`

默认情况下, `\cftZtitlefont` 被定义为字体规范 (font specification)(例如, `\Large\bfseries`), 并且 `\cftafterZtitle` 为空。可以 (通过 `\renewcommand`) 更改这些命令进而更改排版, 例如:

- `\renewcommand{\cftZtitlefont}{\hfill\Large\itshape}` 将排版出大斜体 (Large italic) 右对齐 (flushright) 的标题 (title)。
- `\renewcommand{\cftZtitlefont}{\hfill\Large\bfseries}` 与
`\renewcommand{\cftafterZtitle}{\hfill}` 将排版出居中的大粗体标题 (Large bold title)。
- 按下面这样做:

```
\renewcommand{\cftafterZtitle}{%
  \[\baselineskip]\mbox{}\hfill{\normalfont Page}}
```

将在标题 (title) 后面的一行中加上 “Page” 一词。(如果这样做, 则可能需要减少 `\cftafterZtitleskip`)。

- `\renewcommand{\cftafterZtitle}{\thispagestyle{empty}}` 将使页面标题为空 (即, 不会打印页码)。

2.3 排版条目

还提供了命令, 以便能够更好地控制不同类型条目的排版。定义条目的默认布局 (default layout) 的参数作为 `layouts` 宏包的一部分或在 [GMS94, page 34] 中进行了说明, 并在图 1 中重复。

`\Zdepth` `\Zdepth{<number>}` 命令与标准的 `\tocdepth{<number>}` 命令类似，因为它规定，如果新列表 (new listing) 中的条目的编号级别 (numbering level) 大于 `<number>`，则不应该对它们进行排版。默认定义为 `\setcounter{Zdepth}{1}`。这些命令是必需的，例如 `subcaption` 的用户需要这些命令，`subcaption` 将生成对应于 `lofdepth` 和 `lotdepth` 为 2 的子图标题 (subfigure captions) 和子表标题 (subtable captions)。

在默认的 ToC 排版中，只有较小的条目 (minor entries) 在节标题 (sectioning title) 和页码 (page number) 之间有点引线 (dotted leader lines)。`tocloft` 宏包为所有条目提供一般引线 (general leaders)。引线中的“点 (dot)”由 `\cftdot` 的值给出。它的默认定义是 `\newcommand{\cftdot}{.}`，它给出了默认的点引线。通过更改 `\cftdot` 您可以在引线中使用句号 (period) 以外的符号 (symbols)，例如：

```
\renewcommand{\cftdot}{\ensuremath{\ast}}
```

将会生成一个以星号 (asterisks) 作为符号 (symbol) 的引线。

每种条目都可以控制其前导 (见下文) 中的点 (dots) 之间的间隔。但是，为了保持一致性，所有点引线 (dotted leaders) 应使用相同的间距 (same spacing)。宏 `\cftdotsep` 指定默认间距 (default spacing)。它的值是一个数字。但是，如果间隔 (separation) 太大，则实际上没有点 (dots) 会被排版。宏 `\cftnodots` 是一个“很大的 (too large)”的分隔值 (separation value)。

`\cftsetpnumwidth` 页码 (page numbers) 在固定宽度的盒子中排版。`\cftsetpnumwidth{<length>}` 命令可用于更改盒子的宽度 (L^AT_EX 的内部 `\@pnumwidth`)。标题文本 (title texts) `\cftsetrmarg` 将在到达右边距 (righthand margin) 之前结束。`\cftsetrmarg{<length>}` 可用于设置该距离 (L^AT_EX 的内部 `\@tocrmarg`)。请注意，`\cftsetrmarg` 中使用的长度 (length) 应大于 `\cftsetpnumwidth` 中设置的长度。这些值在任何给定文档中都应保持不变。

页码 (page numbers) 在如上所述的盒子中排版。默认情况下，它们是右对齐的 (right-aligned)，当页码在页面上垂直对齐以便它们数字对齐 (digits line up) 时，右对齐是合适的。例如，对于 ToC 条目与其页码之间具有固定宽度 (fixed width) 的设计，左对齐 (left alignment) 可能更合适。这可以通过将 `\cftpnumalign` 宏设置为 `l`、`c` 或 `r` (就像 `\makebox`) 来控制：

```
\renewcommand{\cftpnumalign}{l}
```

`\cftparskip` 通常，ToC 等中的 `\parskip` 为零。这可以通过更改 `\cftparskip` 长度 (length) 进行更改。请注意，`\cftparskip` 的当前值 (current value) 用于目录

(ToC)、图形目录 (LoF) 和表格目录 (LoT)，但如果其中一个或另一个具有不同的值 (这不是一个好主意)，则可以在调用 `\tableofcontents` 或 `\listoffigures` 或 `\listoftables` 之前更改该值。

我们使用 `x` 代表以下内容：

- 用 `part` 代表 `\part` 标题 (titles)
- 用 `chap` 代表 `\chapter` 标题 (titles)
- 用 `sec` 代表 `\section` 标题 (titles)
- 用 `subsec` 代表 `\subsection` 标题 (titles)
- 用 `subsubsec` 代表 `\subsubsection` 标题 (titles)
- 用 `para` 代表 `\paragraph` 标题 (titles)
- 用 `subpara` 代表 `\subparagraph` 标题 (titles)
- 用 `fig` 代表图形 (figure) `\caption` 标题 (titles)
- 用 `subfig` 代表子图形 (subfigure) `\caption` 标题 (titles)
- 用 `tab` 代表表格 (table) `\caption` 标题 (titles)
- 用 `subtab` 代表子表格 (subtable) `\caption` 标题 (titles)

`\cftbeforeXskip` 这将控制条目前的垂直空间 (vertical space)。可以使用 `\setlength` 来更改。

`\cftXindent` 这控制从左边距 (left margin) 缩进条目 (图 1 中的缩进 [*indent*])。可以使用 `\setlength` 来更改。

`\cftXnumwidth` 这控制了排版标题编号 (title numbers) 所允许的空间 (图 1 中的编号宽度 [*numwidth*])。可以使用 `\setlength` 来更改。多行标题 (multiline title) 的第二行和后续行 (subsequent lines) 将缩进此量。

其余的命令与条目排版的细节 (specifics) 有关。这是一个简化的伪代码版本 (pseudo-code version)，用于已编号条目 (numbered entries) 和未编号条目 (unnumbered entries) 的排版。

```
{\cftXfont {\cftXpresnum SNUM\cftXaftersnum\hfil} \cftXaftersnumb TITLE}%
      {\cftXleader}{\cftXpagefont PAGE}\cftXafterpnum\par

{\cftXfont TITLE}{\cftXleader}{\cftXpagefont PAGE}\cftXafterpnum\par
```

其中 `SNUM` 是节编号 (section number), `TITLE` 是标题文本 (title text), `PAGE` 是页码 (page number)。在已编号的条目 (numbered entry) 中, 伪代码 (pseudo-code)

```
{\cftXpresnum SNUM\cftaftersnum\hfil}
```

是在宽度为 `\cftXnumwidth` 的盒子内排版的。

`\cftXfont` 这将控制标题 (及其前面的编号, 如果有) 的外观。可以使用 `\renewcommand` 命令来更改。

通常, 节编号 (section number) 是在宽度为 `\cftXnumwidth` 的盒子内排版的。

`\cftXpresnum` 在盒子中, 首先调用宏 `\cftXpresnum`, 然后是排版编号 (number), 然后是在排版编号之后调用 `\cftXaftersnum` 宏。盒子中的最后一个命令是 `\hfil`, 它用于使盒子中的内容左对齐 (flushleft)。盒子排版后, 在排版标题文本 (title text) 之前调用 `\cftXaftersnumb` 宏。所有这三个都可以通过 `\renewcommand` 进行更改。默认情况下, 它们被定义为什么都不做。

在标准类 (standard classes) 中, `\part` 的 ToC 条目只是排版为编号 (number) 和标题 (title), 后跟页码 (page number), 在排版编号和标题之前调用 `\cftpartpresnum` 宏。由于 \LaTeX idiosyncracies, 如果第三方宏包的行为与默认的内部 \LaTeX 命令不同, `\cftpartpresnum` 在输出中可能会加倍 (doubled)。`tocloft` 宏包包含特定代码, 以防止 `KomaScript` 类和 `titlesec` 宏包出现这种情况; 如果您发现其他宏包也表现出这种错误行为 (mis-behaviour), 请联系维护人员添加进一步的更正。

当使用标准类 (standard class) 时, `\cftpartaftersnum` 和 `\cftpartaftersnumb` 宏无效, 但如果使用非标准类 (non-standard class), 它们可能会起作用。

`\cftXleader` `\cftXleader` 定义标题 (title) 和页码 (page number) 之间的引线 (leader), 它可以通过 `\renewcommand` 进行更改。引线中任何点 (dots) 之间的间距 (spacing)

`\cftXdotsep` 由 `\cftXdotsep` (图 1 中的 `\@dotsep`) 控制。它可以通过 `\renewcommand` 进行更改, 其值必须是数字 (例如 6.6 或 `\cftdotsep`) 或 `\cftnodots` (禁用点 [dots])。间距以数学单位 (*math units*) 表示, 例如 “18mu to 1em”。

`\cftXpagefont` 这定义了用于排版页码 (page number) 的字体。它可以通过 `\renewcommand` 进行更改。

`\cftXafterpnum` 此宏在页码排版后调用。它的默认值是什么都不做。它可以通过 `\renewcommand` 进行更改。

`\cftsetindents` 命令 `\cftsetindents{<entry>}{<indent>}{<numwidth>}` 设置 `<entry>` 的缩进 (`<indent>`) 为长度 `<indent>`, 设置 `<entry>` 的编号宽度 (`<numwidth>`) 为长度 `<numwidth>`。`<entry>` 参数是一个标准条目 (standard entries) (例如, subsection [子节]) 的名称或

用 `tocloft` 宏包定义的条目的名称。例如, `\cftsetindents{figure}{0em}{1.5em}` 将使图形 (figure) 条目左对齐 (left justified)。

通过单独或组合更改 `\cftXfont`、`\cftXaftersnum`、`\cftXaftersnumb`、`\cftXleader` 和 `\cftXafterpnum` 的定义, 可以实现各种效果 (effects)。为了示例起见, 假设我们有以下初始定义 (initial definitions):

```
\newcommand{\cftXfont}{}
\newcommand{\cftXaftersnum}{}
\newcommand{\cftXaftersnumb}{}
\newcommand{\cftXleader}{\cftdotfill{\cftXdotsep}}
\newcommand{\cftXdotsep}{\cftdotsep}
\newcommand{\cftXpagefont}{}
\newcommand{\cftXafterpnum}{}

```

(请注意, 标题 [title]、页眉 [leader] 和页码 [page number] 应使用相同的字体, 以提供一致的外观 [coherent appearance])。

- 要消除引线中的点 (dots in the leader), 请执行以下操作:

```
\renewcommand{\cftXdotsep}{\cftnodots}

```

- 把某物 (如名字) 放在标题 (编号) 之前, 请执行以下操作:

```
\renewcommand{\cftXpresnum}{SOMETHING }

```

- 要在节编号 (section number) 后添加冒号 (colon), 请执行以下操作:

```
\renewcommand{\cftXaftersnum}{:}

```

- 要在标题编号 (title number) 之前添加内容, 在标题编号之后添加冒号 (colon), 将所有内容设置为粗体字体 (bold font), 并在下一行开始标题文本 (title text), 请执行以下操作:

```
\renewcommand{\cftXfont}{\bfseries}
\renewcommand{\cftXleader}{\bfseries\cftdotfill{\cftXdotsep}}
\renewcommand{\cftXpagefont}{\bfseries}
\renewcommand{\cftXpresnum}{SOMETHING }
\renewcommand{\cftXaftersnum}{:}
\renewcommand{\cftXaftersnumb}{\\}

```

如果在编号盒子 (number box) 中除了编号 (number) 之外还要添加文本 (text), 则可能需要增加盒子的宽度, 以便多行标题 (multiline titles) 具有整齐的垂直对齐方式 (neat vertical alignment); 更改盒子宽度通常意味着缩进 (indents) 也需要修改⁴。调整上述示例的盒子宽度 (box width) 的一种可能方法是:

```
\newlength{\mylen}    % a "scratch" length
\settowidth{\mylen}{\bfseries\cftXpresnum\cftXaftersnum} % 额外空格
\addtolength{\cftXnumwidth}{\mylen} % 添加额外空格 (extra space)
```

- 要将节编号 (section numbers) 设置为右对齐 (flushright)⁵, 执行:

```
\setlength{\mylen}{0.5em}    % 在编号末尾需要一些额外的空格
\renewcommand{\cftXpresnum}{\hfill} % 注意两个 “l”
\renewcommand{\cftXaftersnum}{\hspace*{\mylen}}
\addtolength{\cftXnumwidth}{\mylen}
```

在上文中, 盒子中添加的初始 (initial) `\hfill` 将覆盖盒子中的最终 (final) `\hfil`, 从而将所有内容移动到盒子的右端。额外的空格 (extra space) 是为了使数字 (number) 不会直接排版在标题文本的左侧 (left of the title text)。

- 要将条目设置为左对齐 (ragged left)(但这只适用于单行标题 [single line titles]):

```
\renewcommand{\cftXfont}{\hfill\bfseries}
\renewcommand{\cftXleader}{}
```

- 要将页码 (page number) 设置在条目文本 (entry text) 之后而不是右侧边距 (righthand margin):

```
\renewcommand{\cftXleader}{\hspace*{\fill}}
\renewcommand{\cftXafterpnum}{\cftparfillskip}
\renewcommand{\cftpnumalign}{l}
```

默认情况下, `\parfillskip` 值在局部设置为填充 (fill up) 段落的最后一行。只更改 `\cftXleader` 会在标题的最后一行放置讨厌的字间空格 (interword spaces)。`\cftparfillskip` 命令是 `tocloft` 宏包的一部分, 提供该命令只是为了实现上述效果。此外, 这是一个很好的例子, 说明何时适合更改页码盒子 (page number box) 的对齐方式。

⁴Lyndon Dudding(林登·杜丁)(lyndon.dudding@totalise.co.uk) 发现了这一点。

⁵感谢 David Holz(大卫·霍尔茨) (lbda@earthlink.net) 的请求。

- 要删除表格 (table) 和图形 (figure) 标题条目 (caption entries) 之间插入的空格, 请在章 (chapters) 之间:

```
\begingroup
\renewcommand*{\addvspace}[1]{\vspace{1cm}}
\listoftables
\listoffigures
\endgroup
```

`\cftpagenumbersoff` 命令 `\cftpagenumbersoff{⟨entry⟩}` 将删除列表 (listing) 中 `⟨entry⟩` 的页码 (page numbers), 其中 `⟨entry⟩` 是标准类型的条目 (standard kinds of entries) 之一的名称 (例如, `subsection` 或 `figure` — 如果使用了 `subfigure` 则包括 `subfigure` — 等等), 或由 `tocloft` 宏包定义的新条目的名称。

`\cftpagenumberon` 命令 `\cftpagenumberon{⟨entry⟩}` 反转 (reverses) 相应的 `\cftpagenumbersoff` 的效果。

在 `comp.text.tex` 新闻组 (newsgroup) 中出现的一个问题是, 如何在没有页码的情况下获取目录 (ToC) 中的附录清单 (Appendices list) 的标题。这里有一个简单的方法, 假设文档包含章 (chapters)

```
...
\appendix
\addtocontents{toc}{\cftpagenumbersoff{chapter}}
\chapter{First appendix}
```

如果在附录 (appendices) 之后还有其他章标题 (chaptered headings) 要进入目录 (ToC), 那么就有必要执行

```
\addtocontents{toc}{\cftpagenumberon{chapter}}
```

这样类似的工作以恢复目录 (ToC) 中的页码 (page numbering)。

类似地, 如果您使用的是 `subfigure` 宏包, 您可能需要删除子图片标题 (subfigure captions) 的页码, 这可以通过以下方式完成:

```
\cftpagenumbersoff{subfigure}
```

在这一点上, 我把它留给你的独创性 (ingenuity), 使你可以实现其他效果。但是, 如果您想到了更多的示例, 请让我知道是否可能包含在本文档的后续版本中。

2.4 新列表 ...

```
\newlistof \newlistof[⟨within⟩]{⟨entry⟩}{⟨ext⟩}{⟨listofname⟩}
```

命令创建一个新列表 ... (new List of ...), 以及与之配套的各种命令。

第一个必需的参数 $\langle entry \rangle$ 用于定义一个名为 `entry` 的新计数器 (counter)。可以使用可选的 $\langle within \rangle$ 参数, 每次更改 `within` 调用的计数器 (counter) 时, `entry` 都重置为 1。也就是说, 前两个参数等同于调用 `\newcounter{\langle entry \rangle}[\langle within \rangle]`。

下一个参数 $\langle ext \rangle$ 是新列表 (new List) 的文件扩展名 (file extension)。最后一个参数 $\langle listofname \rangle$ 是新列表 (new List) 的标题文本 (text for the heading)。举个例子:

```
\newcommand{\listanswername}{List of Answers}
\newlistof{chapter}{answer}{ans}{\listanswername}
```

将创建一个新的 `answer` 计数器, 该计数器将在每一 `\chapter{...}` 开始时被重置 (reset)。任何答案标题 (answer titles) 都将写入文件 `jobname.ans`, 并且 `\listanswername` 将用作列表标题 (list heading)。创建的命令 `\listofanswer` 可以像 `\listoftables` 或 `\tableofcontents` 命令一样用来生成一个列表 (listing)。由您来指定如何将这些条目放入新的答案列表 (new List of Answers) 中。下面是一个非常简单的示例, 请记住已经创建了一个 `answer` 计数器。

```
\newcommand{\answer}[1]{%
  \refstepcounter{answer}
  \par\noindent\textbf{Answer \theanswer. #1}
  \addcontentsline{ans}{answer}{\protect\numberline{\theanswer}#1}\par}
```

当使用时, 如:

`\answer{Hard}` The `\ldots` 将打印为:

Answer 1. Hard

The ...

如上所述, `\newlistof` 命令创建了几个新命令, 其中大部分您现在应该熟悉了。为方便起见, 假设 `\newlistof{X}{Z}{...}` 已发布, 因此, `X` 是新计数器的名称, 与第 2.3 节中的 `X` 相对应, `Z` 是新文件扩展名, 与第 2.2 节中的 `Z` 相对应。然后, 除其他外, 将提供以下新命令。

这五个命令 `\cftmarkZ`、`\cftbeforeZtitleskip`、`\cftafterZtitleskip`、`\cftZtitlefont` 和 `\cftafterZtitle`, 与第 2.2 节中描述的同名命令类似。命令 `\listofX` `\listofX` 类似于 `\listoftables` 等, 因为它在调用新列表 (new listing) 的位置排版新列表。

`\Zdepth` `\Zdepth{\langle number \rangle}` 命令类似于标准的 `\tocdepth{\langle number \rangle}` 命令, 因为它

指定, 如果新清单 (new listing) 中的条目 (entries) 的编号级别 (numbering level) 大于 $\langle number \rangle$, 则不应该对它们进行排版。默认定义为 `\setcounter{Zdepth}{1}`。

`\newlistentry` `\newlistentry[$\langle within \rangle$]{ $\langle entry \rangle$ }{ $\langle ext \rangle$ }{ $\langle level-1 \rangle$ }`

命令创建用于在列表 (listing) 中排版新条目的新命令。它由 `\newlistof` 命令内部使用, 但可以独立使用。

第一个必需参数 $\langle entry \rangle$ 用于定义名为 `entry` 的新计数器。可以使用可选的 $\langle within \rangle$ 参数, 每次更改 `within` 调用的计数器 (counter) 时, `entry` 都重置为 1。也就是说, 前两个参数等同于调用 `\newcounter{ $\langle entry \rangle$ }[$\langle within \rangle$]`。第二个必需参数 $\langle ext \rangle$, 是条目列表 (entry listing) 的文件扩展名 (file extension)。最后一个参数 $\langle level-1 \rangle$ 是一个数字, 指定清单中条目的编号级别 (numbering level) 减去 1。例如, `\newlistof[chapter]{answer}{ans}{\listanswername}` 命令将调用 `\newlistentry[chapter]{answer}{ans}{0}` 命令。

调用 `\newlistentry` 会创建几个新命令。假设它被称为 `\newlistentry[within]{X}{Z}{N}`, 其中 `X` 和 `Z` 与前面的用法类似, 并且 `N` 是一个整数, 那么以下命令可用。

`\cftbeforeXskip`、`\cftXfont`、`\cftXpresnum`、`\cftXaftersnum`、
`\cftXaftersnumb`、`\cftXleader`、`\cftXdotsep`、`\cftXpagefont` 和 `\cftXafterpnum`,
这组命令与第 2.3 节中描述的同名命令类似。它们的默认值也如前所述。

`\cftXindent` 和 `\cftXnumwidth` 的默认值是根据 $\langle level-1 \rangle$ 参数 (例如, 在本例中是 `N`) 的值设置的。对于 `N=0`, 设置对应于非分章文档 (non-chaptered documents) 中节 (sections) 的设置, 如表 1 所示。对于 `N=4`, 设置对应于非分章文档 (non-chaptered documents) 中的子段 (subparagraphs) 的设置。而对于 `0<N<4` 的中间值 (intermediate values), 对应于分章文档 (chaptered documents) 中匹配的节划分 (sectional division)。对于 `N<0` 或 `N>4` 的值, 或对于非默认值 (non-default values), 使用 `\cftsetindents` 命令设置值。

`\l@X` `\l@X` 是一个内部命令 (internal command), 用于在排版列表 (listing) 中的条目, 并根据上述 `\cft*X*` 命令进行定义。如果 `\Zdepth` 为 `N` 或更小, 则不会排版条目, 其中 `Z` 是列表的文件扩展名。

`\theX` 命令 `\theX` 打印计数器 `X` 的值。它最初被定义为打印阿拉伯数字 (arabic numerals)。如果使用可选的 $\langle within \rangle$ 参数, 则 `\theX` 定义为: `\renewcommand{\theX}{\thewithin.\arabic{X}}`, 否则, `\theX` 定义为: `\renewcommand{\theX}{\arabic{X}}`。

作为独立使用 `\newlistentry` 的示例, 下面将设置子答案 (sub-answers)。

```
\newlistentry[answer]{subanswer}{1}
\cftsetindents{subanswer}{1.5em}{3.0em}
\renewcommand{\thesubanswer}{\theanswer.\alph{subanswer}}
\newcommand{\subanswer}[1]{%
  \refstepcounter{subanswer}
  \par\textbf{\thesubanswer} #1}
\addcontentsline{ans}{subanswer}{\protect\numberline{\thesubanswer}#1}}
\setcounter{ansdepth}{2}
```

然后:

```
\answer{Harder} The \ldots
\subanswer{Reformulate the problem} 它有助于 \ldots
```

将排版为:

Answer 2. Harder

The ...

2.a) Reformulate the problem 它有助于 ...

默认情况下, 答案条目 (answer entries) 将出现在答案列表 (List of Answers listing) 中 (由 `\listofanswer` 命令排版)。为了显示子答案 (subanswers), 上面使用了 `\setcounter{ansdepth}{2}` 命令。

若要关闭子答案 (subanswers) 的页码, 请执行

```
\cftpagenumbersoff{subanswer}
```

作为 `\newlistentry` 的另一个例子, 假设在 `subparagraph` 之下需要额外的节划分 (sectioning division), 称为子段 (subsubpara)。`\subsubpara` 命令本身可以通过 LaTeX 内核 (kernel) 命令 `\@startsection` 来定义。此外, 还需要定义一个 `\subsubparamark` 宏、一个新的 `subsubpara` 计数器、一个 `\thesubsubpara` 宏和一个 `\l@subsubpara` 宏。如下所示, 使用 `tocloft` 宏包的 `\newlistentry` 可以解决如下所示的大部分问题 (请记住关于带有 `@` 符号的命令的注意事项)。

```
\newcommand{\subsubpara}{\@startsection{subpara} %
  {6} % 级别 (level)
  {\parindent} % 从左边距 (left margin) 缩进
  {3.25ex \@plus1ex \@minus .2ex} % 标题上方间距 (skip above heading)
  {-1em} % 页眉栏标题 (runin heading) 与标题和文本间有 1em
```

```

{\normalfont\normalsize\itshape} % 斜体数字和标题
}
\newlistentry[subparagraph]{subsubpara}{toc}{5}
\cftsetindents{subsubpara}{14.0em}{7.0em}
\newcommand*{\subsubparamark}[1]{ } % 吞噬标题标记 (heading mark)

```

每个 List off...使用一个文件来存储列表条目 (list entries), 这些文件在整个文档处理 (document processing) 过程中必须保持打开状态以便写入。TeX 只有有限数量的文件可以保持打开状态, 这限制了可以使用的列表 (listings) 的数量。对于一个包含目录 (ToC) 但没有其他额外辅助文件 (ancillary files)(例如, 没有索引或参考文献输出文件) 的文档, 包括图形目录 (LoF) 和表格目录 (LoT) 在内的 LoX 的最大数量不能超过 11 个。如果您尝试创建太多的新列表 (new listings), LaTeX 将会返回错误消息:

No room for a new write (译者注: 没有空间以供新的写入)

如果您收到这样的信息, 唯一的办法就是重新设计 (redesign) 您的文档。

tocloft 宏包没有提供简单的方法用于指定新的浮动列表 (Lists of Floats) 或浮动环境 (float environments)。对于这些, 我建议使用 ccaption 宏包 [W101]。

2.5 实验性程序

本节中描述的宏比之前描述的宏更具实验性 (experimental)。

一些老式小说, 甚至一些现代教科书⁶, 都在章标题 (chapter heading) 之后或在目录 (Toc) 中, 或在两个地方, 都有一个简短的章内容提要 (synopsis)。

`\cftchapterprecis` 命令 `\cftchapterprecis{<text>}` 在文档中调用它的位置处打印它的参数, 并将它添加到 .toc 文件, 例如:

```

...
\chapter{} % first chapter
\cftchapterprecis{介绍了我们的英雄; 家谱; 早期。}
...

```

`\cftchapterprecishere` `\cftchapterprecis` 命令调用这两个命令来打印文档 (此处的 `\...here{<text>}`)
`\cftchapterprecistoc` _____

⁶例如, Robert Sedgewick (罗伯特·塞奇威克), 《算法 (Algorithms)》, Addison-Wesley (艾迪生-韦斯利), 1983 年。

命令) 中的文本并将其放入目录 (ToC)(此处的 `\...toc{<text>}` 命令)。如果需要, 可以单独使用。

有时可能需要更改单个条目的全局参数 (global parameters)。例如, 一幅图形 (figure) 可能被放在一本书的最后一页 (封面或封底的内侧), 并需要放在图形目录 (LoF) 中, 其页码设置为 “inside front cover(封面内侧)”。如果 “inside front cover(封面内侧)” 被排版为普通页码 (ordinary page number), 它将突出到页边空白处 (margin)。因此, 需要更改此特定条目 (particular entry) 的参数。

`\cftlocalchange` 命令 `\cftlocalchange{<file>}{<pnumwidth>}{<tocrmarg>}` 将在 `<file>` 中写入一个条目 (entry) 以重置全局参数 (global parameters)。应该在任何特殊条目之后再次调用该命令, 以将参数重置回其通常的值 (usual values)。必须保护参数中使用的任何脆弱命令 (fragile commands)。

`\cftaddtitleline` 命令 `\cftaddtitleline{<file>}{<kind>}{<title>}{<page>}` 将在 `<file>` 中写入一个 `\contentsline` 条目, 作为一个 `<kind>` 条目, 其标题为 `<title>` 而页码为 `<page>`。也就是说, 条目的形式 (form) 是:

```
\contentsline{kind}{title}{page}
```

必须保护参数中使用的任何脆弱命令 (fragile commands)。

`\cftaddnumtitleline` 命令 `\cftaddnumtitleline{<file>}{<kind>}{<num>}{<title>}{<page>}` 类似, 只是它还包含 `<num>` 作为 `\numberline` 的参数。也就是说, 条目的形式 (form) 是:

```
\contentsline{kind}{\numberline{num} title}{page}
```

必须保护参数中使用的任何脆弱命令 (fragile commands)。

作为使用这些命令的示例, 请注意, `\@pnumwidth` 和 `\@tocrmarg` 的默认 L^AT_EX 值分别为 1.55em 和 2.55em, 您可以对扉页 (frontispiece page) 中的图形执行以下操作:

```
...
% 这是没有页码的扉页 (frontispiece page)
% 绘制或导入图形 (没有 \caption)
\cftlocalchange{lof}{4em}{5em} % 使扉页 (frontispiece) 的
% 页码宽度 (pnumwidth) 足够大, 并更改边距 (margin) 以适合
\cftaddtitleline{lof}{figure}{The title}{frontispiece}
\cftlocalchange{lof}{1.55em}{2.55em} % 恢复正常设置
...
```

回想一下, `\caption` 命令将在 `.lof` 文件中放入一个条目 (entry), 这在这里是不需要的。如果需要标题 (caption), 您可以自己制作一个, 或者, 假设您的一般标题 (general captions) 不是太奇特 (exotic), 则可以使用 `ccaption` 宏包中的 `\legend` 命令。如果插图 (illustration) 已编号, 请使用 `\cftaddnumtitleline` 命令而不是 `\cftaddtitleline`。

`\cftZprehook` 实现多栏目录 (multicolumn ToCs) 出人意料地困难, 您能猜到写下面这段话
`\cftZposthook` 的问题是什么吗?

```
\begin{multicols}{2}
\tableofcontents
\end{multicols}
```

在普通 (regular) L^AT_EX 中做到这一点最简单的方法可能是:

```
\RequirePackage{multicol}
\AtBeginDocument{\addtocontents{toc}{\protect\begin{multicols}{2}}}
\AtEndDocument {\addtocontents{toc}{\protect\end {multicols}}}
```

这种写入 `.toc` 文件的方法对于试图控制目录的排版输出 (typesetting output) 是最灵活的。

为使 `tocloft` 稍简单些, 可以使用以下宏: `\cftZprehook` 和 `\cftZposthook`, 其中 `Z` 是 `toc`、`lof`、`lot` 等。如果定义了这些宏, 它们会在目录条目的实际排版之前插入材料 (material), 以此类推。因此, 可以实现多栏目录 (multicolumn ToCs):

```
\RequirePackage{multicol}
\renewcommand\cfttocprehook{\begin{multicols}{2}}
\renewcommand\cfttocposthook{\end{multicols}}
```

2.6 与其他宏包一起使用

`tocloft` 和 `tocbibind` 宏包可以在同一文档中一起使用。`tocbibind` 宏包提供了向目录 (Table of Contents) 添加如参考文献 (bibliography) 和索引 (index) 这样的文档元素 (document elements) 的简单方法。然而, 有一个已知的潜在问题 (known potential problem):

- 如果 `\tocotherhead` 命令的参数不是一个正常的节划分 (sectioning divisions) (即从部分 [part] 到子段 [sub-paragraph]), 比如 `\tocotherhead{clause}`, 那么这几乎肯定会导致问题 (因为 `tocloft` 宏包不知道如何定义相应的 `\l@clause` 命令)。在这种情况下, 您必须自己提供适当的宏。

有些宏包，比如 Anselm Lingnau (安塞姆·林瑙) 的 `float` 宏包，可以创建其他类型的 *List of ...*。`tocloft` 宏包只能最小限度地改变这些包的格式，主要是因为这些宏包彼此独立，而且，在 `float` 宏包的情况下，可以在文档的任何位置动态创建新类型的 `float` 环境及其相关列表 (lists)。排版的某些方面由 `\@cftbsnum`、`\@cftbsnum`、`\@cftasnum` 和 `\@cftasnumb` 命令控制。这些命令等效于前面描述 `\@cftasnum` 的 `\cftXpresnum`、`\cftXaftersnum` 和 `\cftXaftersnumb` 命令。在默认情况下，`\@cftasnumb` 它们被定义为什么也不做，但可能会被重新定义为做某事情。

`tocloft` 宏包和 `minitoc` 宏包有一个不幸的交互 (unfortunate interaction)⁷，幸运的是它可以被修复。在正常情况下，当在分章文档 (chaptered document) 中使用 `minitoc` 时，它会将 `minitocs` 中的节条目 (section entries) 以粗体字体 (bold font) 排版。如果 `tocloft` 与 `Lpackminitoc` 结合使用，则 `minitocs` 节条目 (section entries) 将以普通字体 (normal font) 排版，但页码 (page numbers) 为粗体字体，而目录 (ToC) 中的节条目 (section entries) 则全部为普通字体。

如果您希望 `minitoc` 节条目 (section entries) 全部使用普通字体 (normal font)，一个解决方法是在导言区 (preamble) 放置将下述命令：

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

否则，解决方法就是下面的魔法 (incantation)：

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

要将 ToC 和 `minitoc` 中的节条目 (section entries) 都加粗，请在导言中加入咒语 (incantation)。若要使 `minitoc` 节条目 (section entries) 以粗体显示，而 ToC 条目以普通字体 (normal font) 显示，请将上述魔法放在 `\tableofcontents` 命令和第一个 `\chapter` 命令之间。

一般来说，与其他重新定义的宏包 (而该宏包也能被 `tocloft` 修改) 一起使用可能会有问题。

⁷发现者是林登·杜丁 (lyndon.dudding@totalise.co.uk)。

3 宏包代码

```
1 \langle *usc\rangle
```

为了尽量避免与其他包发生名称冲突 (name clashes), 每个内部名称 (internal name) 都将包含字符串 `@cft`。

`\@cftifundefined` 由于与这个宏包和 `fancyhdr` 在如何检查未定义 (checked for undefinedness) 上会发生的冲突。

```
2 \newcommand\@cftifundefined[1]{%
3   \begingroup\expandafter\expandafter\expandafter\endgroup
4   \expandafter\ifx\csname #1\endcsname\relax
5     \expandafter\@firstoftwo
6   \else
7     \expandafter\@secondoftwo
8   \fi}
```

`\@cftquit` 我们将为 ToC 等使用章 (chapter) 或节 (section) 类型的标题 (type headings), 因

`\ifcfthaschapter` 此我们需要知道文档类 (document class) 支持哪一种。

```
9 \newcommand{\@cftquit}{}
10 \newif\ifcfthaschapter
```

`\ifcftkoma` `koma` 类的默认值与标准类 (standard classes) 不同, 因此我们需要知道是否加载了 `koma` 类。

```
11 \newif\ifcftkoma
12 \@cftkomafalse
13 \@ifclassloaded{scrartcl}{\@cftkomatrue}{}
14 \@ifclassloaded{scrreprt}{\@cftkomatrue}{}
15 \@ifclassloaded{scrbook}{\@cftkomatrue}{}

```

```
\ifcfttitlesec
```

```
16 \newif\ifcfttitlesec
17 \AtBeginDocument{\@ifpackageloaded{titlesec}{\@cfttitlesectrue}{}}
```

如果没有可识别的节分区 (recognised sectional divisions), 则发出警告, 然后跳过宏包代码的其余部分。

```
18 \@cftifundefined{chapter}{%
19   \cfthaschapterfalse

```



```

20 \cftifundefined{section}{%
21   \PackageWarning{tocloft}%
22     {I don't recognize any sectional divisions so I'll do nothing}
23   \renewcommand{\cftquit}{\endinput}
24   }\PackageInfo{tocloft}{The document has section divisions}}
25 }\cftifchaptertrue
26   \PackageInfo{tocloft}{The document has chapter divisions}}

```

也许现在就退出。

```

27 \cftquit

```

如果 `\ifcftchapter` 为 TRUE, 则使用章样式 (chapter style), 否则使用节样式 (section style)。

`\ifcfttobibind` 如果 `tocbibind` 宏包已经加载, 则标志 (flag) 设置为 TRUE。1998/11/15 版本的 `tocbibind` 不一定适用于 `tocloft`。

```

28 \newif\ifcfttobibind
29 \AtBeginDocument{%
30   \ifpackageloaded{tocbibind}{\cfttobibindtrue}{\cfttobibindfalse}
31   \ifcfttobibind
32     \ifpackagelater{tocbibind}{1998/11/16}{-}{%
33       \PackageWarning{tocloft}{%
34         You are using a version of the tocbind package\MessageBreak
35         that is not compatible with tocloft.\MessageBreak
36         The results may be surprising.\MessageBreak
37         Consider installing the current version of tocbind.}}
38   \fi
39 }

```

`\ifcftnctoc` 用于实现 `titles` 选项的布尔值 (boolean)。如果目录 (ToC)、表格目录 (LoT)、图形目录 (LoF) 标题应该使用默认样式 (default styles), 则为 TRUE。

```

40 \newif\ifcftnctoc\cftnctocfalse
41 \DeclareOption{titles}{\cftnctoctrue}
42 %% \ProcessOptions\relax

```

`\ifcftsubfigopt` 用于实现 `subfigure` 选项的布尔值。

```

43 \newif\ifcftsubfigopt\cftsubfigoptfalse
44 \DeclareOption{subfigure}{\cftsubfigopttrue}

```

处理选项。

```
45
46 \ProcessOptions\relax
47
```

`\tocloftpagestyle` 用户级宏 (user-level macro), 用于设置 To C 的第一页的页面样式 (pagestyle) 等。

`\@cftpagestyle` 默认为 plain 页面样式。

```
48 \newcommand{\tocloftpagestyle}[1]{%
49   \def\@cftpagestyle{\thispagestyle{#1}}
50 \tocloftpagestyle{plain}
51
```

`\cftmarktoc` 这三个宏设置了页眉标题 (running heads) 的样式。它们被初始化为默认的外观
`\cftmarklof` (default appearance)。

```
\cftmarklot 52 \newcommand{\cftmarktoc}{%
53   \@mkboth{\MakeUppercase\contentsname}{\MakeUppercase\contentsname}}
54 \newcommand{\cftmarklof}{%
55   \@mkboth{\MakeUppercase\listfigurename}{\MakeUppercase\listfigurename}}
56 \newcommand{\cftmarklot}{%
57   \@mkboth{\MakeUppercase\listtablename}{\MakeUppercase\listtablename}}
58 \if@cftkoma
59   \renewcommand{\cftmarktoc}{%
60     \@mkboth{\contentsname}{\contentsname}}
61   \renewcommand{\cftmarklof}{%
62     \@mkboth{\listfigurename}{\listfigurename}}
63   \renewcommand{\cftmarklot}{%
64     \@mkboth{\listtablename}{\listtablename}}
65 \fi
```

`\@cfttocstart` 在 `\tableofcontents` 命令的开头和结尾执行操作 (perform the actions) 的两个
`\@cfttocfinish` 宏 (和好友)。`\@cfttocstart` 处理分章文档 (chaptered documents), 确保 ToC
排版成单栏 (single column)(原始代码请参见 `classes.dtx`)。ccaption 宏包也提供
这些宏。

```
66 \providecommand{\@cfttocstart}{%
67   \if@cfthaschapter
68     \if@twocolumn
```

```

69      \@restonecoltrue\onecolumn
70      \else
71      \@restonecolfalse
72      \fi
73 \fi}

```

如果需要, `\@cfttocfinish` 重置为两栏排版。

```

74 \providecommand{\@cfttocfinish}{%
75   \if@cfthaschapter
76     \if@restonecol\twocolumn\fi
77 \fi}

```

`\phantomsection` 这是因为 `hyperref` 宏包与 `\addcontentsline` 一起使用。

```

78 \providecommand{\phantomsection}{}
79

```

`\@cftdobibtoc` 如果使用了 `tocbibind` 宏包, 并且它重新定义了 `\tableofcontents`, 我们需要满足这一点。定义的内容 (contents of the definition) 在 `tocbibind` 中定义。

```

80 \newcommand{\@cftdobibtoc}{%
81   \if@dotoc
82     \if@bibchapter
83       \phantomsection
84       \addcontentsline{toc}{chapter}{\contentsname}
85     \else
86       \phantomsection
87       \addcontentsline{toc}{\@tocextra}{\contentsname}
88     \fi
89 \fi}
90

```

`\cftparskip` ToC 等的局部 `\parskip` 设置为长度 (length) `\cftparskip`。

```

91 \newlength{\cftparskip}
92 \setlength{\cftparskip}{0pt}
93

```

`\tableofcontents` 这是默认 `\tableofcontents` 命令的参数化版本 (parameterised version)。每个类 (class) 都有自己的定义, 但我们必须在一个定义中满足所有类, 因此需要进行一些检查 (checks)。在加载所有宏包之后修改定义。

如果已使用 titles 选项, 则不会修改该命令。

```
94 \AtBeginDocument{%
95 \if@cftnctoc
96 % ensure \cftparskip is still set properly
97 \let\OLD@starttoc\@starttoc
98 \renewcommand{\@starttoc}[1]{%
99 \begingroup
100 \parskip=\cftparskip
101 \OLD@starttoc{#1}%
102 \endgroup
103 }
104 \else
105 \renewcommand{\tableofcontents}{%
106 \@cfttocstart
```

确保前面的段落 (previous paragraph) 已经完成。在一个组 (group) 中设置局部段落样式 (local paragraphing style) 并排版标题 (title)。

```
107 \par
108 \begingroup
109 \parindent\z@ \parskip\cftparskip
110 \@cftmaketoctitle
```

如果已使用 tocbibind, 则将 ToC 名称添加到 ToC

```
111 \if@cfttocbibind
112 \@cftdobibtoc
113 \fi
```

最后, 读取 .toc 文件并结束 (finish up)。

```
114 \@starttoc{toc}%
115 \endgroup
116 \@cfttocfinish}
117 \fi
118 }
```

`\@cftmaketoctitle` 此命令排版 ToC 的标题。

```
119 \newcommand{\@cftmaketoctitle}{%
120 \addpenalty\@secpenalty
121 \if@cftthaschapter
```

```

122     \vspace*{\cftbeforetoctitleskip}%
123   \else
124     \vspace{\cftbeforetoctitleskip}%
125   \fi
126   \@cftpagestyle
127   {\interlinepenalty\@M
128    {\cfttoctitlefont\contentsname}{\cftaftertoctitle}%
129    \cftmarktoc
130    \par\nobreak
131    \vskip \cftaftertoctitleskip
132    \@afterheading}}

```

`\cftbeforetoctitleskip` 这两个长度 (lengths) 控制 ToC 标题前后的垂直间距 (vertical spacing)。

```

\cftaftertoctitleskip 133 \newlength{\cftbeforetoctitleskip}
134 \newlength{\cftaftertoctitleskip}

```

它们的值取决于文档是否有章 (chapters)。在分章文档 (chaptered documents) 中, 默认的 ToC 标题被排版为 `\chapter*`, 否则排版为 `\section*`。

```

135 \if@cfthaschapter
136   \setlength{\cftbeforetoctitleskip}{50pt}
137   \setlength{\cftaftertoctitleskip}{40pt}
138 \else
139   \setlength{\cftbeforetoctitleskip}{3.5ex \@plus 1ex \@minus .2ex}
140   \setlength{\cftaftertoctitleskip}{2.3ex \@plus .2ex}
141 \fi

```

`\cfttoctitlefont` ToC 标题按 `\cfttoctitlefont` 给定的样式排版。在排版标题后调用 `\cftaftertoctitle` `\cftaftertoctitle` 宏。这被初始化为不执行任何操作。这两个宏都可以被重新定义以执行其他操作 (例如, 在 `\cfttoctitlefont` 中添加一个 `\hfill` 会使标题右对齐 [flushright])。

```

142 \if@cfthaschapter
143   \newcommand{\cfttoctitlefont}{\normalfont\Huge\bfseries}
144   \if@cfthaschapter\renewcommand{\cfttoctitlefont}{\size@chapter\sectfont}\fi
145 \else
146   \newcommand{\cfttoctitlefont}{\normalfont\Large\bfseries}
147   \if@cfthaschapter\renewcommand{\cfttoctitlefont}{\size@section\sectfont}\fi
148 \fi

```

```
149 \newcommand{\cftaftertoctitle}{}

```

`\cftsetpnumwidth` 用于设置 `\@pnumwidth` 和 `\@tocrmarg` 的用户命令。

```
\cftsetrmarg 150 \newcommand{\cftsetpnumwidth}[1]{\renewcommand{\@pnumwidth}{#1}}
151 \newcommand{\cftsetrmarg}[1]{\renewcommand{\@tocrmarg}{#1}}

```

`\cftpnumalign` 对齐字符串 (作为页码盒子的 `\makebox` 输入)。

```
152 \newcommand{\cftpnumalign}{r}

```

`\cftdot` 在默认 ToC 中, 可以使用点线 (dotted line) 在标题 (title) 和页码 (page number) 之间提供引线 (leader)。该引线的定义隐藏在 `\@dottedtocline` 命令中。
`\cftdotfill{<sep>}` 命令提供引线代码 (leader code) 的参数化版本 (parameterised version), 其中 `<sep>` 是以 `mu` 为单位的点 (dots) 之间的间距 (separation)。引线中用于“点 (dots)”的符号 (symbol) 由 `\cftdot` 的值给出。这些宏也由 `ccaption` 宏包提供。

```
153 \providecommand{\cftdot}{.}
154 \providecommand{\cftdotfill}[1]{%
155   \def\@tempa{#1}%
156   \def\@tempb{\cftnodots}%
157   \ifx\@tempa\@tempb
158     \hfill
159   \else
160     \leaders\hbox{$\m@th\mkern #1 mu\hbox{\cftdot}\mkern #1 mu$}\hfill
161   \fi
162 }

```

`\cftdotsep` `\cftdotsep` 保留默认的点分隔 (default dot separation), 也由 `ccaption` 包提供。如

`\cftnodots` 果 `\cftdotfill` 中的压缩字距 (kerns) 足够大, 则不会打印任何点。`\cftnodots` 应“足够大”。(实际上, `\cftnodots` 现在用作条件分支 [conditional branch] 的标志 [flag], 因此其数值现在不再那么重要。)

```
163 \providecommand{\cftdotsep}{4.5}
164 \newcommand{\cftnodots}{5000}

```

现在来看一下关于 ToC 条目排版的更棘手的问题。

`.toc` (也是 `.lof` 和 `.lot`) 文件是由 `\contentsline{<kind>}{<title>}{<page>}` 命令组成的列表 (list), 其中 `<kind>` 是标题的类型 (kind of heading) (例如 `part` 或 `section` 或 `figure`)。 `<title>` 是标题文本 (title text) (包括编号 [number])。 `<page>` 是

页码 (page number)。通过调用 `\addcontentsline{<file>}{<kind>}{<title>}` 命令将条目 (entries) 插入文件, 其中 `<file>` 是文件扩展名 (例如 `toc`、`lot`), 其他参数与 `\contentsline` 命令相同。任意内容也可以通过 `\addtocontents{<file>}{<text>}` 命令放入文件中。`\contentsline` 条目的排版由 `\l@kind` 形式 (form) 的命令执行。分节命令 (sectioning commands) 和标题命令 (captioning commands) 调用 `\addcontentsline` 将标题 (titles) 插入 `.toc` 等文件中。

为了方便起见, 通常不可能将标题 (title) 和其编号 (number) 的排版分开处理, 因为两者都绑定到 (bundled into) `\contentsline` 中的 `<title>` 参数。如果适当修改了 `\contentsline` 命令, 则可以单独处理它们。如果这样做了, 那么 `\addtocontentsline` 命令也需要修改, 同样也需要修改分节命令 (sectioning commands) 和标题命令 (captioning commands)。这当然是可能的, 但如果任何其他宏包也修改了分节命令或标题命令, 就会产生问题, 并且有几个宏包可以这样做。

如上所述, 对于除 Part 条目外的所有条目, 节编号 (sectional number) 都是通过 `\numberline` 命令进行排版的。我们可以利用这个事实。

我已经决定不触碰 `\contentsline` 宏, 而是在它存在的情况下使用它。也就是说, 我将修改 `\l@kind` 命令。本质上, 我的新定义由 `\@dottedtocline` 代码的内联版本 (inlined versions) 组成。

`\cftparfillskip` `\l@kind` 命令 (局部 [locally]) 修改 `\parfillskip` 的值。`\cftparfillskip` 是 *TeXbook* `\parfillskip` 默认定义的副本。

```
165 \newcommand{\cftparfillskip}{\parfillskip=0pt plus1fil}
```

`\numberline` `\numberline{<secnum>}` 命令的目的是在一个宽度为 `\@tempdima` 的盒子中左对齐排版 `<secnum>`。我重新定义了它, 添加了三个额外的参数 (additional parameters), 即 `\@cftbsnum`、`\@cftasnum` 和 `\@cftasnumb` (原始定义见 `ltsect.dtx`)。

```
166 \renewcommand{\numberline}[1]{%
```

```
167   \hb@xt@\@tempdima{\@cftbsnum #1\@cftasnum\hfil}\@cftasnumb}
```

`\@cftbsnum` 最初, 这些命令未定义, 但是在 `\l@...` 命令中 `\let` 使用适当的命令, 但是必须 `\@cftasnum` 定义它们, 以防意外调用 `\numberline`, 例如通过使用 `float` 宏包⁸。

```
\@cftasnumb 168 \newcommand{\@cftbsnum}{}%
```

```
169 \newcommand{\@cftasnum}{}%
```

⁸这个 bug 是 Andrew Thurber (安德鲁·瑟伯) 在同时使用 `tocloft` 宏和 `algorithm` 宏包时发现的。

```
170 \newcommand{\@cftasnumb}{}
```

`\l@part \l@part{<title>}{<page>}` 排版 part 标题 (heading) 的 ToC 条目。它是默认 `\if@cftdopart` `\l@part` 的参数化副本 (parameterised copy)(参见 `classes.dtx` 以获取原始定义, 并获取 `\l@section` 的以下代码及大部分代码的解释)。默认情况下, 部分 (Parts)(和章 [Chapters]) 没有点引线 (dotted leaders)。此包为所有条目提供了点引线。

```
171 \newif\if@cftdopart
172 \newif\if@cfthaspart
173 \@cftifundefined{part}{\@cfthaspartfalse}{\@cfthasparttrue}
174 \if@cfthaspart
175 \renewcommand*{\l@part}[2]{%
176   \@cftdopartfalse
177   \ifnum \c@tocdepth >-2\relax
178     \if@cfthaschapter
179       \@cftdoparttrue
180     \fi
181     \ifnum \c@tocdepth >\m@ne
182       \if@cfthaschapter\else
183         \@cftdoparttrue
184       \fi
185     \fi
186   \fi
187   \if@cftdopart
188     \if@cfthaschapter
189       \addpenalty{-\@highpenalty}%
190     \else
191       \addpenalty\@secpenalty
192     \fi
193     \addvspace{\cftbeforepartskip}%
194     \begingroup
195       {\leftskip \cftpartindent\relax
196        \rightskip \@tocrmarg
197        \parfillskip -\rightskip
198        \parindent \cftpartindent\relax\@afterindenttrue
199        \interlinepenalty\@M
```



```

200      \leavevmode
201      \@tempdima \cftpnumwidth\relax
202      \let\@cftbsnum \cftpnum
203      \let\@cftasnum \cftpnum
204      \let\@cftasnumb \cftpnumb
205      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip

```

在默认的 L^AT_EX 中，一部分 ToC 条目在写入时没有 `\numberline`，因此需要手动插入“presnum”。在 Koma-Script (科马脚本) 和 titlesec (可能还有其他地方—请让我知晓!) 中，然而，情况并非如此。

```

206      {\cftpnumfont \if@cfkoma\else\if@cftitlesec\else\cftpnum\fi\fi #1}%
207      \cftpnumfillnum{#2}}
208      \nobreak
209      \if@cfthaschapter
210        \global\@nobreaktrue
211        \everypar{\global\@nobreakfalse\everypar{}}%
212      \else
213        \if@compatibility
214          \global\@nobreaktrue
215          \everypar{\global\@nobreakfalse\everypar{}}%
216        \fi
217      \fi
218    \endgroup
219  \fi}
220 \fi

```

`\cftbeforepartskip` 这些是用于控制部分条目 (Part entries) 排版的用户命令 (user commands)。它们

`\cftpnumwidth` 经过了初始化，以提供标准外观 (standard appearance)。

```

\cftpnumfont 221 \if@cfthaspart
\cftpnumpresnum 222 \newlength{\cftbeforepartskip}
\cftpnumaftersnum 223 \setlength{\cftbeforepartskip}{2.25em \@plus\p@}
\cftpnumaftersnumb 224 \newlength{\cftpnumwidth}
\cftpnumleader 225 \setlength{\cftpnumwidth}{0em}
\cftpnumdotsep 226 \newcommand{\cftpnumfont}{\large\bfseries}
\cftpnumpagefont 227 \newcommand{\cftpnumpresnum}{}
\cftpnumafterpnum 228 \newcommand{\cftpnumaftersnum}{}
\cftpnumindent 229 \newcommand{\cftpnumaftersnumb}{}
\cftpnumfillnum

```

```

230 \newcommand{\cftpартleader}{\large\bfseries\cftdotfill{\cftpартdotsep}}
231 \newcommand{\cftpартdotsep}{\cftnodots}
232 \newcommand{\cftpартpagefont}{\large\bfseries}
233 \newcommand{\cftpартafterpnum}{\}
234 \newlength{\cftpартindent}
235 \setlength{\cftpартindent}{0em}
236 \newcommand{\cftpартfillnum}[1]{%
237   {\cftpартleader}%
238   {\makebox[\@pnumwidth][\cftpnumalign]{
239     \cftpартpagefont #1\cftpартafterpnum\par}%
240   }

```

koma 类使用一些不同的设置。

```

241 \if@cfkoma
242   \setlength{\cftpартnumwidth}{2em}
243   \renewcommand{\cftpартfont}{\sectfont\large}
244   \renewcommand{\cftpартpagefont}{\sectfont\large}
245 \fi
246 \fi

```

`\l@chapter \l@chapter{<title>}{<page>}` 排版章标题 (chapter heading) 的 ToC 条目。它是默认 `\l@chapter` 的参数化副本 (parameterised copy)(参见 `classes.dtx` 获取原始定义 [original definition])。这仅适用于分章文档 (chaptered documents)。

```

247 \if@cfthaschapter
248 \renewcommand*{\l@chapter}[2]{%
249   \ifnum \c@tocdepth >\m@ne
250     \addpenalty{-\@highpenalty}%
251     \vskip \cftbeforechapskip
252     {\leftskip \cftchapindent\relax
253       \rightskip \@tocrmarg
254       \parfillskip -\rightskip
255       \parindent \cftchapindent\relax\@afterindenttrue
256       \interlinepenalty\@M
257       \leavevmode
258       \@tempdima \cftchapnumwidth\relax
259       \let\@cftbsnum \cftchappresnum
260       \let\@cftasnum \cftchapaftersnum

```

```

261      \let\@cftasnumb \cftchapaftersnumb
262      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
263      {\cftchapfont #1}\nobreak
264      \cftchapfillnum{#2}}%
265      \fi}%
266 \fi

```

`\cftbeforechapskip` 这些是控制章条目 (Chapter entries) 排版的用户命令。它们经过了初始化，以提供标准外观 (standard appearance)。

```

\cftchapnumwidth 267 \if@cftchapter
\cftchapfont 268 \newlength{\cftbeforechapskip}
\cftchappresnum 269 \setlength{\cftbeforechapskip}{1.0em \@plus\p@}
\cftchapaftersnum 270 \newlength{\cftchapindent}
\cftchapaftersnumb 271 \setlength{\cftchapindent}{0em}
\cftchapleader 272 \newlength{\cftchapnumwidth}
\cftchapdotsep 273 \setlength{\cftchapnumwidth}{1.5em}
\cftchappagefont 274 \newcommand{\cftchapfont}{\bfseries}
\cftchapafterpnum 275 \newcommand{\cftchappresnum}{\bfseries}
\cftchapfillnum 276 \newcommand{\cftchapaftersnum}{\bfseries}
277 \newcommand{\cftchapaftersnumb}{\bfseries}
278 \newcommand{\cftchapleader}{\bfseries\cftdotfill{\cftchapdotsep}}
279 \newcommand{\cftchapdotsep}{\cftnodots}
280 \newcommand{\cftchappagefont}{\bfseries}
281 \newcommand{\cftchapafterpnum}{\bfseries}
282 \newcommand{\cftchapfillnum}[1]{%
283   {\cftchapleader}\nobreak
284   \makebox[\@pnumwidth][\cftpnumalign]{
285     \cftchappagefont #1}\cftchapafterpnum\par
286   }

```

`koma` 类有不同的章设置 (chapter settings)。

```

287 \if@cftkoma
288 \renewcommand{\cftchapfont}{\sectfont}
289 \fi
290 \fi
291

```

`\l@section` `\l@section{<title>}{<page>}` 排版节标题 (section heading) 的 ToC 条目。它是

默认 `\l@section` 的参数化副本 (parameterised copy)(请参见 `classes.dtx` 以获取原始定义)。

```

292 \renewcommand*{\l@section}[2]{%
293   \ifnum \c@tocdepth >\z@
294     \if@cfthaschapter
295       \vskip \cftbeforesecskip
296     \else
297       \addpenalty\@secpenalty
298       \addvspace{\cftbeforesecskip}
299     \fi
300     {\leftskip \cftsecindent\relax
301      \rightskip \@tocrmarg
302      \parfillskip -\rightskip
303      \parindent \cftsecindent\relax\@afterindenttrue
304      \interlinepenalty\@M
305      \leavevmode
306      \@tempdima \cftsecnumwidth\relax
307      \let\@cftbsnum \cftsecpresnum
308      \let\@cftasnum \cftsecaftersnum
309      \let\@cftasnumb \cftsecaftersnumb
310      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
311      {\cftsecfont #1}\nobreak
312      \cftsecfillnum{#2}}%
313   \fi}

```

`\cftbeforesecskip` 这些是用户命令，用于控制节条目 (Section entries) 的排版。它们经过了初始化，
`\cftsecindent` 以提供标准外观 (standard appearance)。

```

\cftsecnumwidth 314 \newlength{\cftbeforesecskip}
\cftsecfont      315 \newlength{\cftsecindent}
\cftsecpresnum   316 \newlength{\cftsecnumwidth}
\cftsecaftersnum 317 \newcommand{\cftsecpresnum}{ }
\cftsecaftersnumb 318 \newcommand{\cftsecaftersnum}{ }
\cftsecaftersnumb 319 \newcommand{\cftsecaftersnumb}{ }
\cftsecleader     320 \if@cfthaschapter
\cftsecdotsep     321 \setlength{\cftbeforesecskip}{\z@ \@plus.2\p@}
\cftsecpagefont   322 \setlength{\cftsecindent}{1.5em}
\cftsecafterpnum
\cftsecfillnum

```

```

323 \setlength{\cftsecnumwidth}{2.3em}
324 \newcommand{\cftsecfont}{\normalfont}
325 \newcommand{\cftsecleader}{\normalfont\cftdotfill{\cftsecdotsep}}
326 \newcommand{\cftsecdotsep}{\cftdotsep}
327 \newcommand{\cftsecpagefont}{\normalfont}
328 \else
329 \setlength{\cftbeforesecskip}{1.0em \@plus\p@}
330 \setlength{\cftsecindent}{0em}
331 \setlength{\cftsecnumwidth}{1.5em}
332 \newcommand{\cftsecfont}{\bfseries}
333 \newcommand{\cftsecleader}{\bfseries\cftdotfill{\cftsecdotsep}}
334 \newcommand{\cftsecdotsep}{\cftnodots}
335 \newcommand{\cftsecpagefont}{\bfseries}
336 \fi
337 \newcommand{\cftsecafterpnum}{}
338 \newcommand{\cftsecfillnum}[1]{%
339   {\cftsecleader}\nobreak
340   \makebox[\@pnumwidth][\cftpnumalign]{
341     \cftsecpagefont #1}\cftsecafterpnum\par
342 }

```

`\l@section` `\l@section{<title>}{<page>}` 排版子节标题 (subsection heading) 的 ToC 条目。它是默认 `\l@section` 的参数化副本 (parameterised copy)(请参见 `classes.dtx` 以获取原始定义)。

```
343 \renewcommand*{\l@section}[2]{%
```

只有当条目在 `tocdepth` 内时, 才可以对其进行排版。

```
344 \ifnum \c@tocdepth >\@ne
```

Add some vertical space.

```
345 \vskip \cftbeforesubsecskip
```

创建一个组 (group), 使段落更改 (paragraphing changes) 保持为局部 (local)。将 `\leftskip` 设置为条目的缩进 (entry's indentation)。

```
346 {\leftskip \cftsubsecindent\relax
```

将 `\rightskip` 设置为 `\@tocrmarg`, 为页码留出空间。

```
347 \rightskip \@tocrmarg
```

确保填写条目的最后一行。如果将 `\parfillskip` 设置为负数, 则可以防止任何溢出盒子的消息 (overfull box messages)。

```
348      \parfillskip -\rightskip
```

将段落缩进 (paragraph indent) 设置为条目的缩进 (entry's indentation)。

```
349      \parindent \cftsubsecindent\relax\@afterindenttrue
```

尽量避免多行条目 (multiple line entry) 中的行之间出现中断 (breaks)。

```
350      \interlinepenalty\@M
```

确保我们已经是左垂直模式 (left vertical mode)。

```
351      \leavevmode
```

我们的 `\numberline` 版本要求数字盒子 (number box) 的宽度以 `\@tempdima` 为单位, 并定义了三个宏 `\@cftbsnum`、`\@cftasnum` 和 `\@cftasnumb`。我们将所有这些设置为该条目的值。

```
352      \@tempdima \cftsubsecnumwidth\relax
```

```
353      \let\@cftbsnum \cftsubsecpresnum
```

```
354      \let\@cftasnum \cftsubsecaftersnum
```

```
355      \let\@cftasnumb \cftsubsecaftersnumb
```

安排在当前缩进处 (current indent) 设置标题的 (节编号和) 第一行, 并进一步缩进任何其他行。

```
356      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
```

打印 (编号和) 标题, 禁止任何断行 (breaking)。

```
357      {\cftsubsecfont #1}\nobreak
```

打印引线 (leader) 和页码, 然后关闭组 (group)。

```
358      \cftsubsecfillnum{#2}}%
```

```
359      \fi}
```

`\cftbeforesubsecskip` 这些是控制子节条目 (Sub-section entries) 排版的用户命令。它们经过了初始化, `\cftsubsecindent` 以提供标准外观 (standard appearance)。

```
\cftsubsecnumwidth 360 \newlength{\cftbeforesubsecskip}
```

```
\cftsubsecfont 361 \setlength{\cftbeforesubsecskip}{\z@ \@plus.2\p@}
```

```
\cftsubsecpresnum 362 \newlength{\cftsubsecindent}
```

```
\cftsubsecaftersnum 363 \newlength{\cftsubsecnumwidth}
```

```
\cftsubsecaftersnumb 364 \if@cfthaschapter
```

```
\cftsubsecleader 365 \setlength{\cftsubsecindent}{3.8em}
```

```
\cftsubsecdotsep
```

```
\cftsubsecpagefont
```

```
\cftsubsecafterpnum
```

```

366 \setlength{\cftsubsecnumwidth}{3.2em}
367 \else
368 \setlength{\cftsubsecindent}{1.5em}
369 \setlength{\cftsubsecnumwidth}{2.3em}
370 \fi
371 \newcommand{\cftsubsecfont}{\normalfont}
372 \newcommand{\cftsubsecpresnum}{}
373 \newcommand{\cftsubsecaftersnum}{}
374 \newcommand{\cftsubsecaftersnumb}{}
375 \newcommand{\cftsubsecleader}{\normalfont\cftdotfill{\cftsubsecdotsep}}
376 \newcommand{\cftsubsecdotsep}{\cftdotsep}
377 \newcommand{\cftsubsecpagefont}{\normalfont}
378 \newcommand{\cftsubsecafterpnum}{}

```

`\cftsubsecfillnum` `\cftsubsecfillnum{<page>}` 排版小节条目 (subsection entry) 的引线 (leader) 和页码 (<page> number)。首先打印引线 (leader), 然后, 没有断行 (no break), 在宽度为 `\@pnumwidth` 的盒子中设置页码 (page number), 不要忘了结束段落 (finish the paragraph)。

```

379 \newcommand{\cftsubsecfillnum}[1]{%
380   {\cftsubsecleader}\nobreak
381   \makebox[\@pnumwidth][\cftpnumalign]{
382     \cftsubsecpagefont #1}\cftsubsecafterpnum\par
383 }

```

`\l@subsubsection` `\l@subsubsection{<title>}{<page>}` 排版子子节标题 (subsubsection heading) 的 ToC 条目。它是默认 `\l@subsubsection` 的参数化副本 (parameterised copy)(请参见 `classes.dtx` 以获取原始定义)。

```

384 \renewcommand*{\l@subsubsection}[2]{%
385   \ifnum \c@tocdepth >\tw@
386     \vskip \cftbeforesubsubsecskip
387     {\leftskip \cftsubsubsecindent\relax
388      \rightskip \@tocrmarg
389      \parfillskip -\rightskip
390      \parindent \cftsubsubsecindent\relax\@afterindenttrue
391      \interlinepenalty\@M
392      \leavevmode

```

```

393      \@tempdima \cftsubsubsecnumwidth\relax
394      \let\@cftbsnum \cftsubsubsecpresnum
395      \let\@cftasnum \cftsubsubsecaftersnum
396      \let\@cftasnumb \cftsubsubsecaftersnumb
397      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
398      {\cftsubsubsecfont #1}\nobreak
399      \cftsubsubsecfillnum{#2}}%
400  \fi}

```

`\cftbeforesubsubsecskip` 这些是控制子子节条目 (Sub-sub-section entries) 排版的用户命令。它们经过了初始化, 以提供标准外观 (standard appearance)。

```

\cftsubsubsecnumwidth 401 \newlength{\cftbeforesubsubsecskip}
\cftsubsubsecfont      402 \setlength{\cftbeforesubsubsecskip}{\z@ \@plus.2\p@}
\cftsubsubsecpresnum   403 \newlength{\cftsubsubsecindent}
\cftsubsubsecaftersnum 404 \newlength{\cftsubsubsecnumwidth}
\cftsubsubsecaftersnumb 405 \if@cfthaschapter
\cftsubsubsecleader     406 \setlength{\cftsubsubsecindent}{7.0em}
\cftsubsubsecdotsep     407 \setlength{\cftsubsubsecnumwidth}{4.1em}
\cftsubsubsecpagefont   408 \else
\cftsubsubsecafterpnum  409 \setlength{\cftsubsubsecindent}{3.8em}
\cftsubsubsecfillnum    410 \setlength{\cftsubsubsecnumwidth}{3.2em}
\cftsubsubsecfillnum    411 \fi
                        412 \newcommand{\cftsubsubsecfont}{\normalfont}
                        413 \newcommand{\cftsubsubsecpresnum}{\cftsubsubsecfont}
                        414 \newcommand{\cftsubsubsecaftersnum}{\cftsubsubsecfont}
                        415 \newcommand{\cftsubsubsecaftersnumb}{\cftsubsubsecfont}
                        416 \newcommand{\cftsubsubsecleader}{\normalfont\cftdotfill{\cftsubsubsecdotsep}}
                        417 \newcommand{\cftsubsubsecdotsep}{\cftdotsep}
                        418 \newcommand{\cftsubsubsecpagefont}{\normalfont}
                        419 \newcommand{\cftsubsubsecafterpnum}{\cftsubsubsecfont}
                        420 \newcommand{\cftsubsubsecfillnum}[1]{%
                        421   {\cftsubsubsecleader}\nobreak
                        422   \makebox[\@pnumwidth][\cftpnumalign]{
                        423     \cftsubsubsecpagefont #1}\cftsubsubsecafterpnum\par
                        424 }

```

`\l@paragraph \l@paragraph{<title>}{<page>}` 排版段落标题 (paragraph heading) 的 ToC 条目。

它是默认 `\l@paragraph` 的参数化副本 (parameterised copy)(请参见 `classes.dtx` 以获取原始定义)。

```

425 \renewcommand*{\l@paragraph}[2]{%
426   \ifnum \c@tocdepth >3\relax
427     \vskip \cftbeforeparaskip
428     {\leftskip \cftparaindent\relax
429       \rightskip \@tocrmarg
430       \parfillskip -\rightskip
431       \parindent \cftparaindent\relax\@afterindenttrue
432       \interlinepenalty\@M
433       \leavevmode
434       \@tempdima \cftparanumwidth\relax
435       \let\@cftbsnum \cftparapresnum
436       \let\@cftasnum \cftparaaftersnum
437       \let\@cftasnumb \cftparaaftersnumb
438       \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
439       {\cftparafont #1}\nobreak
440       \cftparafillnum{#2}}}%
441 \fi}

```

`\cftbeforeparaskip` 这些是控制段落条目 (Paragraph entries) 排版的用户命令。它们经过了初始化，
`\cftparaindent` 以提供标准外观 (standard appearance)。

```

\cftparanumwidth 442 \newlength{\cftbeforeparaskip}
\cftparafont      443 \setlength{\cftbeforeparaskip}{\z@ \@plus.2\p@}
\cftparapresnum   444 \newlength{\cftparaindent}
\cftparaaftersnum 445 \newlength{\cftparanumwidth}
\cftparaaftersnumb 446 \if@cfthaschapter
\cftparaleader     447 \setlength{\cftparaindent}{10em}
\cftparadotsep     448 \setlength{\cftparanumwidth}{5em}
\cftparapagefont   449 \else
\cftparaafterpnum  450 \setlength{\cftparaindent}{7.0em}
\cftparafillnum    451 \setlength{\cftparanumwidth}{4.1em}
\cftparafillnum    452 \fi
\cftparafont       453 \newcommand{\cftparafont}{\normalfont}
\cftparapresnum    454 \newcommand{\cftparapresnum}{\}
\cftparaaftersnum  455 \newcommand{\cftparaaftersnum}{\}

```

```

456 \newcommand{\cftparaaftersnumb}{\}
457 \newcommand{\cftparaleader}{\normalfont\cftdotfill{\cftparadotsep}}
458 \newcommand{\cftparadotsep}{\cftdotsep}
459 \newcommand{\cftparapagefont}{\normalfont}
460 \newcommand{\cftparaafterpnum}{\}
461 \newcommand{\cftparafillnum}[1]{%
462   {\cftparaleader}\nobreak
463   \makebox[\@pnumwidth][\cftpnumalign]{
464     \cftparapagefont #1}\cftparaafterpnum\par
465 }

```

`\l@subparagraph` `\l@subparagraph{<title>}{<page>}` 为子段标题 (subparagraph heading) 排版 ToC 条目。它是默认 `\l@subparagraph` 的参数化副本 (parameterised copy)(请参见 `classes.dtx` 以获取原始定义)。

```

466 \renewcommand*{\l@subparagraph}[2]{%
467   \ifnum \c@tocdepth >4\relax
468     \vskip \cftbeforesubparaskip
469     {\leftskip \cftsubparaindent\relax
470       \rightskip \@tocrmarg
471       \parfillskip -\rightskip
472       \parindent \cftsubparaindent\relax\@afterindenttrue
473       \interlinepenalty\@M
474       \leavevmode
475       \@tempdima \cftsubparanumwidth\relax
476       \let\@cftbsnum \cftsubparapresnum
477       \let\@cftasnum \cftsubparaaftersnum
478       \let\@cftasnumb \cftsubparaaftersnumb
479       \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
480       {\cftsubparafont #1}\nobreak
481       \cftsubparafillnum{#2}}%
482   \fi}

```

`\cftbeforesubparaskip` 这些是控制子段落条目 (Sub-paragraph entries) 排版的用户命令。它们经过了初始化, 以提供标准外观 (standard appearance)。

```

\cftsubparanumwidth 483 \newlength{\cftbeforesubparaskip}
\cftsubparafont      484 \setlength{\cftbeforesubparaskip}{\z@ \@plus.2\p@}
\cftsubparapresnum   485 \newlength{\cftsubparaindent}
\cftsubparaaftersnum 486 \newlength{\cftsubparanumwidth}
\cftsubparaaftersnumb 487 \if@cfthaschapter
\cftsubparaleader     488 \setlength{\cftsubparaindent}{12em}
\cftsubparadotsep     489 \setlength{\cftsubparanumwidth}{6em}
\cftsubparapagefont   490 \else
\cftsubparaafterpnum  491 \setlength{\cftsubparaindent}{10em}
\cftsubparafillnum    492 \setlength{\cftsubparanumwidth}{5em}
\cftsubparafillnum    493 \fi
\cftsubparafont       494 \newcommand{\cftsubparafont}{\normalfont}
\cftsubparapresnum    495 \newcommand{\cftsubparapresnum}{}
\cftsubparaaftersnum  496 \newcommand{\cftsubparaaftersnum}{}
\cftsubparaaftersnumb 497 \newcommand{\cftsubparaaftersnumb}{}
\cftsubparaleader     498 \newcommand{\cftsubparaleader}{\normalfont\cftdotfill{\cftsubparadotsep}}
\cftsubparadotsep     499 \newcommand{\cftsubparadotsep}{\cftdotsep}
\cftsubparapagefont   500 \newcommand{\cftsubparapagefont}{\normalfont}
\cftsubparaafterpnum  501 \newcommand{\cftsubparaafterpnum}{}
\cftsubparafillnum    502 \newcommand{\cftsubparafillnum}[1]{%
\cftsubparaleader}\nobreak
\makebox[\@pnumwidth][\cftpnumalign]{
\cftsubparapagefont #1}\cftsubparaafterpnum\par
506 }
```

`\@cftdobiblof` 如果使用了 `toctibind` 宏包, 并且它重新定义了 `\listoffigures`, 我们需要满足这一点 (cater for that)。定义的内容 (contents of the definition) 在 `toctibind` 中定义。

```

507 \newcommand{\@cftdobiblof}{%
508 \if@dotoclof
509 \if@bibchapter
510 \phantomsection
511 \addcontentsline{toc}{chapter}{\listfigurename}
512 \else
```

```

513     \phantomsection
514     \addcontentsline{toc}{\@tocextra}{\listfigurename}
515     \fi
516     \fi}
517

```

`\listoffigures` 这是默认 `\listoffigures` 命令的参数化版本 (parameterised version)。更改 (changes) 将推迟到加载所有的宏包之后。每个类 (class) 都有自己的定义, 但我们必须在一个定义中满足所有类, 因此需要进行一些检查。首先, 对多栏 (multicolumns) 执行默认检查。(如果使用 `titles` 选项, 则不执行任何操作)。

```

518 \AtBeginDocument{%
519 \if@cftnctoc\else
520 \renewcommand{\listoffigures}{%
521   \cfttocstart

```

确保前面的段落 (previous paragraph) 已经结束 (finished)。在组内 (Within a group) 设置局部段落样式 (local paragraphing style)。排版标题, 然后执行 `.lof` 文件的内容。

```

522   \par
523   \begingroup
524     \parindent\z@ \parskip\cftparskip
525     \@cftmakeloftitle
526     \if@cfttocbibind
527       \cftdobiblof
528     \fi
529     \@starttoc{lof}%
530   \endgroup

```

Finally, restore any multicolumn typesetting.

```

531   \@cfttocfinish}%
532 \fi
533 }
534

```

`\@cftmakeloftitle` 此命令排版图形目录 (LoF) 的标题 (title)。

```

535 \newcommand{\@cftmakeloftitle}{%
536   \addpenalty\@secpenalty
537   \if@cfthaschapter

```

```

538     \vspace*{\cftbeforelofttitleskip}
539   \else
540     \vspace{\cftbeforelofttitleskip}
541   \fi
542   \@cftpagestyle
543   {\interlinepenalty\@M
544    {\cftlofttitlefont\listfigurename}{\cftafterlofttitle}
545    \cftmarklof
546    \par\nobreak
547    \vskip \cftafterlofttitleskip
548    \@afterheading}}
549

```

`\cftbeforelofttitleskip` 这两个长度控制图形目录 (LoF) 标题前后的垂直间距 (vertical spacing)。

```

\cftafterlofttitleskip 550 \newlength{\cftbeforelofttitleskip}
551 \newlength{\cftafterlofttitleskip}

```

它们的值取决于文档是否有章 (chapters)。在分章文档 (chaptered documents) 中，默认的图形目录 (LoF) 标题被排版为 `\chapter*`，而非分章文档将图形目录 (LoF) 标题被排版为 `\section*`。

```

552 \if@cfthaschapter
553   \setlength{\cftbeforelofttitleskip}{50pt}
554   \setlength{\cftafterlofttitleskip}{40pt}
555 \else
556   \setlength{\cftbeforelofttitleskip}{3.5ex \@plus 1ex \@minus .2ex}
557   \setlength{\cftafterlofttitleskip}{2.3ex \@plus .2ex}
558 \fi

```

`\cftlofttitlefont` 图形目录 (LoF) 标题按 `\cftlofttitlefont` 给出的样式排版。宏 `\cftafterlofttitle` 在排版标题后被调用。这被初始化为不执行任何操作。这两个宏都可以被重新定义以执行其他操作 (例如，在 `\cftlofttitlefont` 中添加 `\hfill` 会使标题右对齐 [flushright])。

```

559 \if@cfthaschapter
560   \newcommand{\cftlofttitlefont}{\normalfont\Huge\bfseries}
561   \if@cfthaschapter\renewcommand{\cftlofttitlefont}{\size@chapter\sectfont}\fi
562 \else
563   \newcommand{\cftlofttitlefont}{\normalfont\Large\bfseries}

```

```

564 \if@ctkoma\renewcommand{\cftloftitlefont}{\size@section\sectfont}\fi
565 \fi
566 \newcommand{\cftafterloftitle}{}
567

```

`\l@figure` `\l@figure{<title>}{<page>}` 排版图形标题 (figure caption heading) 的 LoF 条目。它是默认 `\l@figure` 的参数化副本 (parameterised copy)(请参见 `classes.dtx` 以获取原始定义)。

```

568 \renewcommand*{\l@figure}[2]{%
569   \ifnum \c@lofdepth >\z@
570     \vskip \cftbeforefigskip
571     {\leftskip \cftfigindent\relax
572      \rightskip \@tocrmarg
573      \parfillskip -\rightskip
574      \parindent \cftfigindent\relax\@afterindenttrue
575      \interlinepenalty\@M
576      \leavevmode
577      \@tempdima \cftfignumwidth\relax
578      \let\@cftbsnum \cftfigpresnum
579      \let\@cftasnum \cftfigaftersnum
580      \let\@cftasnumb \cftfigaftersnumb
581      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
582      {\cftfigfont #1}\nobreak
583      \cftfigfillnum{#2}}%
584   \fi
585 }

```

`\cftbeforefigskip` 这些是用户命令，用于控制图标题条目 (Figure caption entries) 的排版。它们经

`\cftfigindent` 过了初始化，以提供标准外观 (standard appearance)。

```

\cftfignumwidth 586 \newlength{\cftbeforefigskip}
\cftfigfont 587 \setlength{\cftbeforefigskip}{\z@ \@plus.2\p@}
\cftfigpresnum 588 \newlength{\cftfigindent}
\cftfigaftersnum 589 \setlength{\cftfigindent}{1.5em}
\cftfigaftersnumb 590 \newlength{\cftfignumwidth}
\cftfigleader 591 \setlength{\cftfignumwidth}{2.3em}
\cftfigdotsep 592 \newcommand{\cftfigfont}{\normalfont}
\cftfigpagefont 593 \newcommand{\cftfigpresnum}{}
\cftfigafterpnum
\cftfigfillnum

```

```

594 \newcommand{\cftfigaftersnum}{ }
595 \newcommand{\cftfigaftersnumb}{ }
596 \newcommand{\cftfigleader}{\normalfont\cftdotfill{\cftfigdotsep}}
597 \newcommand{\cftfigdotsep}{\cftdotsep}
598 \newcommand{\cftfigpagefont}{\normalfont}
599 \newcommand{\cftfigafterpnum}{ }
600 \newcommand{\cftfigfillnum}[1]{%
601   {\cftfigleader}\nobreak
602   \makebox[\@pnumwidth][\cftpnumalign]{\cftfigpagefont #1}
603   \cftfigafterpnum\par
604 }

```

lofdepth 计数器 **lofdepth** 和 **lotdepth** 由 **subfigure** 宏包定义。如果未使用该宏包，请在 **lotdepth** 此处定义它们。

```

605 \if@cftsubfigopt\else
606   \newcounter{lofdepth}\setcounter{lofdepth}{1}
607   \newcounter{lotdepth}\setcounter{lotdepth}{1}
608 \fi
609

```

\cftdobiblot 如果使用了 **tocbibind** 宏包，并且它重新定义了 **\listoftables**，我们需要满足这一点 (cater for that)。定义的内容 (contents of the definition) 在 **tocbibind** 中定义。

```

610 \newcommand{\cftdobiblot}{%
611   \if@dotoclot
612     \if@bibchapter
613       \phantomsection
614       \addcontentsline{toc}{chapter}{\listtablename}
615     \else
616       \phantomsection
617       \addcontentsline{toc}{\@tocextra}{\listtablename}
618     \fi
619   \fi}
620

```

\listoftables 这是默认 **\listoftables** 命令的参数化版本 (parameterised version)。更改 (changes) 将推迟到加载所有宏包之后。每个类 (class) 都有自己的定义，但我们

必须在一个定义中满足所有类，因此需要进行一些检查 (checks)。首先，对多栏 (multicolumns) 执行默认检查 (default checks)。(如果已使用 `titles` 选项，则不执行任何操作)。

```
621 \AtBeginDocument{%
622 \if@cftnctoc\else
623 \renewcommand{\listoftables}{%
624 \@cfttocstart
```

确保前面的段落 (previous paragraph) 已经结束。在组内 (Within a group) 设置局部段落样式 (local paragraphing style)。排版标题，然后执行 `.lot` 文件的内容。

```
625 \par
626 \begingroup
627 \parindent\z@ \parskip\cftparskip
628 \@cftmakelottitle
629 \if@cfttocbibind
630 \cftdobiblot
631 \fi
632 \@starttoc{lot}%
633 \endgroup
```

最后，恢复多列排版 (multicolumn typesetting)。

```
634 \@cfttocfinish}%
635 \fi
636 }
637
```

`\@cftmakelottitle` 此命令排版表格目录 (LoT) 的标题。

```
638 \newcommand{\@cftmakelottitle}{%
639 \addpenalty\@secpenalty
640 \if@cftaschapter
641 \vspace*{\cftbeforelottitleskip}
642 \else
643 \vspace{\cftbeforelottitleskip}
644 \fi
645 \@cftpagestyle
646 {\interlinepenalty\@M
647 {\cftlottitlefont\listtablename}{\cftafterlottitle}
```



```

648 \cftmarklot
649 \par\nobreak
650 \vskip \cftafterlottitleskip
651 \@afterheading}}
652

```

`\cftbeforelottitleskip` 这两个长度控制表格目录 (LoT) 标题前后的垂直间距 (vertical spacing)。

```

\cftafterlottitleskip 653 \newlength{\cftbeforelottitleskip}
654 \newlength{\cftafterlottitleskip}

```

它们的值取决于文档是否有章 (chapters)。在分节文档 (chaptered documents) 中, 默认的表格目录 (LoT) 标题被排版为 `\chapter*`, 而在非分章文档中, 表格目录 (LoT) 标题被排版为 `\section*`。

```

655 \if@cfthaschapter
656 \setlength{\cftbeforelottitleskip}{50pt}
657 \setlength{\cftafterlottitleskip}{40pt}
658 \else
659 \setlength{\cftbeforelottitleskip}{3.5ex \@plus 1ex \@minus .2ex}
660 \setlength{\cftafterlottitleskip}{2.3ex \@plus .2ex}
661 \fi

```

`\cftlottitlefont` 表格目录 (LoT) 标题按 `\cftlottitlefont` 给出的样式排版。宏 `\cftafterlottitle` 在排版标题后被调用。这被初始化为不执行任何操作。这两个宏都可以被重新定义以执行其他操作 (例如, 在 `\cftlottitlefont` 中添加 `\hfill` 会使标题右对齐)。

```

662 \if@cfthaschapter
663 \newcommand{\cftlottitlefont}{\normalfont\Huge\bfseries}
664 \if@cfthaschapter\renewcommand{\cftlottitlefont}{\size@chapter\sectfont}\fi
665 \else
666 \newcommand{\cftlottitlefont}{\normalfont\Large\bfseries}
667 \if@cfthaschapter\renewcommand{\cftlottitlefont}{\size@section\sectfont}\fi
668 \fi
669 \newcommand{\cftafterlottitle}{}
670

```

`\l@table` `\l@table{<title>}{<page>}` 为表格标题 (table caption heading) 排版 LoT 条目。它是默认 `\l@table` 的参数化副本 (parameterised copy)(有关原始定义, 请参见

classes.dtx 以获取原始定义)。

```
671 \renewcommand*{\l@table}[2]{%
672   \ifnum\c@lotdepth >\z@
673     \vskip \cftbeforetabskip
674     {\leftskip \cfttabindent\relax
675      \rightskip \@tocrmarg
676      \parfillskip -\rightskip
677      \parindent \cfttabindent\relax\@afterindenttrue
678      \interlinepenalty\@M
679      \leavevmode
680      \@tempdima \cfttabnumwidth\relax
681      \let\@cftbsnum \cfttabpresnum
682      \let\@cftasnum \cfttabaftersnum
683      \let\@cftasnumb \cfttabaftersnumb
684      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
685      {\cfttabfont #1}\nobreak
686      \cfttabfillnum{#2}}%
687   \fi
688 }
```

`\cftbeforetabskip` 这些是用户命令，用于控制表格标题条目 (Table caption entries) 的排版。它们经过初始化，以提供标准外观 (standard appearance)。

```
\cfttabnumwidth 689 \newlength{\cftbeforetabskip}
\cfttabfont 690 \setlength{\cftbeforetabskip}{\z@ \@plus.2\p@}
\cfttabpresnum 691 \newlength{\cfttabindent}
\cfttabaftersnum 692 \setlength{\cfttabindent}{1.5em}
\cfttabaftersnumb 693 \newlength{\cfttabnumwidth}
\cfttableader 694 \setlength{\cfttabnumwidth}{2.3em}
\cfttabdotsep 695 \newcommand{\cfttabfont}{\normalfont}
\cfttabpagefont 696 \newcommand{\cfttabpresnum}{\cftdotfill{\cfttabdotsep}}
\cfttabafterpnum 697 \newcommand{\cfttabaftersnum}{\cftdotsep}
\cfttabfillnum 698 \newcommand{\cfttabaftersnumb}{\cftdotsep}
699 \newcommand{\cfttableader}{\normalfont\cftdotfill{\cfttabdotsep}}
700 \newcommand{\cfttabdotsep}{\cftdotsep}
701 \newcommand{\cfttabpagefont}{\normalfont}
702 \newcommand{\cfttabafterpnum}{\cftdotsep}
```

```

703 \newcommand{\cfttabfillnum}[1]{%
704   {\cfttableader}\nobreak
705   \makebox[\@pnumwidth][\cftpnumalign]{\cfttabpagefont #1}\cfttabafterpnum\par
706 }

```

3.1 对 subfigure 宏包的支持

支持 subfigure 宏包的代码在本质上与图形 (figure) 和表格 (table) 标题的代码相同，只有名称被更改。但是，只有在 subfigure 宏包实际加载时才需要执行代码。

`\@cftl@subfig` 此命令重新定义了 `\l@subfigure` 命令。

```

707 \newcommand{\@cftl@subfig}{%

```

`\l@subfigure` `\l@subfigure{<title>}{<page>}` 为子图形标题 (subfigure caption heading) 排版 LoF 条目。它本质上与 `\l@figure` 的参数化代码 (parameterised code) 相同，除了必须考虑到 `lofdepth`。

```

708 \renewcommand*{\l@subfigure}[2]{%
709   \ifnum \c@lofdepth > \toclevel@subfigure
710     \vskip \cftbeforesubfigskip
711     {\leftskip \cftsubfigindent\relax
712       \rightskip \@tocrmarg
713       \parfillskip -\rightskip
714       \parindent \cftsubfigindent\relax\@afterindenttrue
715       \interlinepenalty\@M
716       \leavevmode
717       \@tempdima \cftsubfignumwidth\relax
718       \let\@cftbsnum \cftsubfigpresnum
719       \let\@cftasnum \cftsubfigaftersnum
720       \let\@cftasnumb \cftsubfigaftersnumb
721       \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
722       {\cftsubfigfont ##1}\nobreak
723       \cftsubfigfillnum{##2}}%
724   \fi
725 }%
726 }
727

```

`\cftsetsubfig` 此命令初始化 LoF 中的子图形标题 (subfigure captions) 的设置。

```
728 \newcommand{\cftsetsubfig}{%  
  
\cftbeforesubfigskip  
    \cftsubfigindent 729 \newlength{\cftbeforesubfigskip}  
\cftsubfignumwidth 730 \setlength{\cftbeforesubfigskip}{\z@ \@plus.2\p@}  
    \cftsubfigfont 731 \newlength{\cftsubfigindent}  
    \cftsubfigpresnum 732 \setlength{\cftsubfigindent}{3.8em}  
    \cftsubfigaftersnum 733 \newlength{\cftsubfignumwidth}  
    \cftsubfigaftersnumb 734 \setlength{\cftsubfignumwidth}{2.5em}  
    \cftsubfigleader 735 \newcommand{\cftsubfigfont}{\normalfont}  
    \cftsubfigdotsep 736 \newcommand{\cftsubfigpresnum}{}  
    \cftsubfigpagefont 737 \newcommand{\cftsubfigaftersnum}{}  
    \cftsubfigafterpnum 738 \newcommand{\cftsubfigaftersnumb}{}  
    \toclevel@subfig 739 \newcommand{\cftsubfigleader}{\normalfont\cftdotfill{\cftsubtabdotsep}}  
    \cftsubfigfillnum 740 \newcommand{\cftsubfigdotsep}{\cftdotsep}  
    741 \newcommand{\cftsubfigpagefont}{\normalfont}  
    742 \newcommand{\cftsubfigafterpnum}{}  
    743 \providecommand{\toclevel@subfigure}{1}  
    744 \newcommand{\cftsubfigfillnum}[1]{%  
    745     {\cftsubfigleader}\nobreak  
    746     \makebox[\@pnumwidth][\cftpnumalign]{  
    747         \cftsubfigpagefont ##1}\cftsubfigafterpnum\par  
    748 }
```

这是 `\cftsetsubfig` 的结尾 (end)。

```
749 }  
750
```

`\cftl@subtab` 此代码重新定义了 `\l@subtable` 的代码。

```
751 \newcommand{\cftl@subtab}{%
```

`\l@subtable` `\l@subtable{\<title>}{\<page>}` 为子表格标题 (subtable caption heading) 设置 LoT 条目。它本质上与 `\l@table` 的参数化代码 (parameterised code) 相同, 除了必须考虑到 `lotdepth`。

```
752 \renewcommand*{\l@subtable}[2]{%  
753     \ifnum \c@lotdepth > \toclevel@subtable
```

```

754 \vskip \cftbeforesubtabskip
755 {\leftskip \cftsubtabindent\relax
756 \rightskip \@tocrmarg
757 \parfillskip -\rightskip
758 \parindent \cftsubtabindent\relax\@afterindenttrue
759 \interlinepenalty\@M
760 \leavevmode
761 \@tempdima \cftsubtabnumwidth\relax
762 \let\@cftbsnum \cftsubtabpresnum
763 \let\@cftasnum \cftsubtabaftersnum
764 \let\@cftasnumb \cftsubtabaftersnumb
765 \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
766 {\cftsubtabfont ##1}\nobreak
767 \cftsubtabfillnum{##2}}%
768 \fi
769 }%
770 }

```

`\cftsetsubtab` 此命令为 LoT 中的子表格条目 (subtable entries) 设置默认值。

```

771 \newcommand{\cftsetsubtab}{%

```

`\cftbeforesubtabskip` 这些是用于控制子表格标题条目 (Subtable caption entries) 排版的用户命令。它

`\cftsubtabindent` 们经过了初始化, 以提供标准外观 (standard appearance)。

```

\cftsubtabnumwidth 772 \newlength{\cftbeforesubtabskip}
\cftsubtabfont 773 \setlength{\cftbeforesubtabskip}{\z@ \@plus.2\p@}
\cftsubtabpresnum 774 \newlength{\cftsubtabindent}
\cftsubtabaftersnum 775 \setlength{\cftsubtabindent}{3.8em}
\cftsubtabaftersnumb 776 \newlength{\cftsubtabnumwidth}
\cftsubtableader 777 \setlength{\cftsubtabnumwidth}{2.5em}
\cftsubtabdotsep 778 \newcommand{\cftsubtabfont}{\normalfont}
\cftsubtabpagefont 779 \newcommand{\cftsubtabpresnum}{\cftdotfill{\cftsubtabdotsep}}
\cftsubtabafterpnum 780 \newcommand{\cftsubtabaftersnum}{\cftdotsep}
\cftsubtabafterpnumb 781 \newcommand{\cftsubtabaftersnumb}{\cftdotsep}
\cftsubtabafterpnum 782 \newcommand{\cftsubtableader}{\normalfont\cftdotfill{\cftsubtabdotsep}}
\cftsubtabfillnum 783 \newcommand{\cftsubtabdotsep}{\cftdotsep}
784 \newcommand{\cftsubtabpagefont}{\normalfont}
785 \newcommand{\cftsubtabafterpnum}{\cftdotsep}

```

```

786 \providecommand{\toclevel@subtable}{1}
787 \newcommand{\cftsubtabfillnum}[1]{%
788   {\cftsubtableleader}\nobreak
789   \makebox[\@pnumwidth][\cftpnumalign]{\cftsubtabpagefont ##1}
790   \cftsubtabafterpnum\par
791 }

```

这是 `\@cftsetsubtab` 的结尾 (end)。

```

792 }
793

```

仅当指定了 `subfigure` 选项时，才调用 `subfigure` 宏包设置代码。`\l@...` 的重定义 (redefinitions) 必须在加载了 `subfigure` 宏包之后进行。

```

794
795 \if@cftsubfigopt
796   \@cftsetsubfig\@cftsetsubtab
797   \AtBeginDocument{\@cftl@subfig\@cftl@subtab}
798 \fi
799 %%   \AtBeginDocument{\if@cftsubfigopt
800 %%     \@cftsetsubfig\@cftsetsubtab
801 %%     \@cftl@subfig\@cftl@subtab
802 %%   \fi}
803

```

3.2 新列表 ...

`\newlistentry` `\newlistentry[\langle within \rangle]{\langle counter \rangle}{\langle ext \rangle}{\langle level-1 \rangle}` 为列表 (List) 中的新类型条目 (new kind of entry) 创建一组命令。

```

804 \newcommand{\newlistentry}[4][\@empty]{%

```

`\c@X` 检查是否已定义 `\langle within \rangle` 和 `\langle counter \rangle`。如果没有定义 `\langle within \rangle`，则出现错误。如 `\theX` 果已定义 `\langle counter \rangle`，也会出现错误。设置默认计数器值 (default counter values)。

```

805   \@ifundefined{c@#2}{%      check & set the counter
806     \ifx \@empty#1\relax
807       \newcounter{#2}
808     \else
809       \@ifundefined{c@#1}{\PackageWarning{tocloft}%

```

```

810             {#1 has no counter for use as a `within'}
811     \newcounter{#2}}%
812     {\newcounter{#2}[#1]%
813     \expandafter\edef\csname the#2\endcsname{%
814     \expandafter\noexpand\csname the#1\endcsname.\noexpand\arabic{#2}}}
815     \fi
816     \setcounter{#2}{0}
817 }
818 {\PackageError{tocloft}{#2 has been previously defined}{\@eha}}
819

```

这样就完成了错误检查 (error checking)。无论结果如何, 新命令的其余部分都已定义。

`\l@X \l@X{<title>}{<page>}` 排版条目。

```

820 \@namedef{l@#2}##1##2{%
只有当 \Zdepth 深度大于 <level-1> 时才会排版。
821     \ifnum \@nameuse{c@#3depth} > #4\relax
822     \vskip \@nameuse{cftbefore#2skip}
823     {\leftskip \@nameuse{cft#2indent}\relax
824     \rightskip \@tocrmarg
825     \parfillskip -\rightskip
826     \parindent \@nameuse{cft#2indent}\relax\@afterindenttrue
827     \interlinepenalty\@M
828     \leavevmode
829     \@tempdima \@nameuse{cft#2numwidth}\relax
830     \expandafter\let\expandafter\@cftbsnum\csname cft#2presnum\endcsname
831     \expandafter\let\expandafter\@cftasnum\csname cft#2aftersnum\endcsname
832     \expandafter\let\expandafter\@cftasnumb\csname cft#2aftersnumb\endcsname
833     \advance\leftskip\@tempdima \null\nobreak\hskip -\leftskip
834     {\@nameuse{cft#2font}##1}\nobreak
835     \@nameuse{cft#2fillnum}{##2}}%
836     \fi
837 } % end of \l@#2
838

```

现在定义 `\l@X` 使用的所有布局命令 (layout commands)。它们的默认值对应

于非分章文档 (non-chaptered documents) 中节条目 (section entries) 的默认值。

`\cftbeforeXskip`

```
839 \expandafter\newlength\csname cftbefore#2skip\endcsname
840 \setlength{\@nameuse{cftbefore#2skip}}{\z@ \@plus .2\p@}
```

`\cftXindent`

```
\cftXnumwidth 841 \expandafter\newlength\csname cft#2indent\endcsname
842 \expandafter\newlength\csname cft#2numwidth\endcsname
```

根据条目的级别 (level) 设置缩进 (indent) 和编号宽度 (numwidth) 的默认值。级别 1 对应于图形条目 (figure entry)。

```
843 \ifcase #4\relax % 0
844 \setlength{\@nameuse{cft#2indent}}{0em}
845 \setlength{\@nameuse{cft#2numwidth}}{1.5em}
846 \or % 1
847 \setlength{\@nameuse{cft#2indent}}{1.5em}
848 \setlength{\@nameuse{cft#2numwidth}}{2.3em}
849 \or % 2
850 \setlength{\@nameuse{cft#2indent}}{3.8em}
851 \setlength{\@nameuse{cft#2numwidth}}{3.2em}
852 \or % 3
853 \setlength{\@nameuse{cft#2indent}}{7.0em}
854 \setlength{\@nameuse{cft#2numwidth}}{4.1em}
855 \else % anything else
856 \setlength{\@nameuse{cft#2indent}}{10.0em}
857 \setlength{\@nameuse{cft#2numwidth}}{5.0em}
858 \fi
```

`\cftXfont` And the remaining commands.

```
\cftXpresnum 859 \@namedef{cft#2font}{\normalfont}
\cftXaftersnum 860 \@namedef{cft#2presnum}{ }
\cftXaftersnumb 861 \@namedef{cft#2aftersnum}{ }
\cftXdotsep 862 \@namedef{cft#2aftersnumb}{ }
\cftXleader 863 \@namedef{cft#2dotsep}{\cftdotsep}
\cftXpagefont 864 \@namedef{cft#2leader}{\normalfont\cftdotfill{\@nameuse{cft#2dotsep}}}
865 \@namedef{cft#2pagefont}{\normalfont}
\cftXafterpnum 866 \@namedef{cft#2afterpnum}{ }
```


`\toclevel@X` Hyperref 宏包需要一个命令 `\toclevel@X`, 以保存 $\langle level-1 \rangle$ 的值。

```
867 \namedef{toclevel@#2}{#4}
```

`\cftXfillnum` 排版页眉 (leader) 和页码 (page number)。

```
868 \namedef{cft#2fillnum}##1{%
869   {\@nameuse{cft#2leader}}\nobreak
870   \makebox[\@pnumwidth][\cftpnalign]{\@nameuse{cft#2pagefont}##1}
871   \@nameuse{cft#2afterpnum}\par
872 }
```

这样就结束了 `\newlistentry` 的定义。

```
873 }
```

`\newlistof` `\newlistof` [$\langle within \rangle$] { $\langle entry \rangle$ } { $\langle ext \rangle$ } { $\langle listofname \rangle$ } 创建新 List 的命令。

```
874 \newcommand{\newlistof}[4][\@empty]{%
```

调用 `\newlistentry` 设置第一级条目 (first level entry)。

```
875 \ifx \@empty#1\relax
876   \newlistentry{#2}{#3}{0}
877 \else
878   \newlistentry[#1]{#2}{#3}{0}
879 \fi
```

`\ext@Z` 文件扩展名 (file extension) 和列表深度 (listing depth)。

```
\Zdepth 880 \namedef{ext@#2}{#3}
881 \newcounter{#3depth}
882 \setcounter{#3depth}{1}
```

`\cftmarkZ` 列表 (listing) 的标题标记 (heading marks)。

```
883 \if@cftkoma
884   \namedef{cftmark#3}{%
885     \@mkboth{#4}{#4}}
886 \else
887   \namedef{cftmark#3}{%
888     \@mkboth{\MakeUppercase{#4}}{\MakeUppercase{#4}}}
889 \fi
```

`\listofX` 排版列表标题 (listing title) 和条目 (entries)。

```
890 \if@cftnctoc
```

对于 titles 选项，基本上是从标准 \tableofcontents 命令复制代码。

```

891 \namedef{listof#2}{%
892   \@cfttocstart
893   \if@cfthaschapter
894     \chapter*{#4}
895   \else
896     \section*{#4}
897   \fi
898   \@nameuse{cftmark#3}
899   \@starttoc{#3}%
900   \@cfttocfinish}
901 \else

```

否则，请使用完全参数化的定义 (fully parameterised definition)。

```

902 \namedef{listof#2}{%
903   \@cfttocstart
904   \par
905   \begingroup
906     \parindent\z@ \parskip\cftparskip
907     \@nameuse{@cftmake#3title}
908     \@starttoc{#3}%
909   \endgroup
910   \@cfttocfinish}
911 \fi
912

```

\@cftmakeZtitle 排版标题。

```

913 \namedef{@cftmake#3title}{%
914   \addpenalty\@secpenalty
915   \if@cfthaschapter
916     \vspace*{\@nameuse{cftbefore#3titleskip}}%
917   \else
918     \vspace{\@nameuse{cftbefore#3titleskip}}%
919   \fi
920   \@cftpagestyle
921   {\interlinepenalty\@M
922     {\@nameuse{cft#3titlefont}#4}{\@nameuse{cftafter#3title}}}%

```

```

923 \nameuse{cftmark#3}%
924 \par\nobreak
925 \vskip \nameuse{cftafter#3titleskip}%
926 \@afterheading}}
927

```

`\cftbeforeZtitleskip` 标题 (title heading) 之前和之后的间距 (skips), 以及标题字体 (title font)。默认

`\cftafterZtitleskip` 值取决于文档类 (document class) 是否具有章 (chapters)。

```

\cftZtitlefont 928 \expandafter\newlength\csname cftbefore#3titleskip\endcsname
929 \expandafter\newlength\csname cftafter#3titleskip\endcsname
930 \if@cfthaschapter
931 \setlength{\nameuse{cftbefore#3titleskip}}{50pt}
932 \setlength{\nameuse{cftafter#3titleskip}}{40pt}
933 \if@cfthaschapter
934 \namedef{cft#3titlefont}{\size@chapter\sectfont}
935 \else
936 \namedef{cft#3titlefont}{\normalfont\Huge\bfseries}
937 \fi
938 \else
939 \setlength{\nameuse{cftbefore#3titleskip}}{3.5ex \@plus 1ex \@minus .2ex}
940 \setlength{\nameuse{cftafter#3titleskip}}{2.3ex \@plus .2ex}
941 \if@cfthaschapter
942 \namedef{cft#3titlefont}{\size@section\sectfont}
943 \else
944 \namedef{cft#3titlefont}{\normalfont\Huge\bfseries}
945 \fi
946 \fi

```

`\cftafterZtitle` 标题 (title) 后面的东西。

```

947 \namedef{cftafter#3title}{}

```

`\cftZprehook` 条目列表 (list of entries) 前后的钩子 (Hooks)。

```

\cftZposthook 948 \namedef{cft#3prehook}{}
949 \namedef{cft#3posthook}{}

```

这是 `\newlistof` 定义的结尾 (end)。

```

950 }

```

`\cftsetindents` `\cftsetindents{<entry>}{<indent>}{<numwidth>}` 设置 `<entry>` 条目 `<entry>` 的缩进 (`indent`) 和编号宽度 (`numwidth`)。宏必须在外部条目名称 (external entry name) 和内部简写 (internal shorthand) 之间进行映射 (map)。

```

951 \newcommand{\cftsetindents}[3]{%
952   \def\@cftemp{#1}
953   \ifx\@cftemp\cftchapname
954     \@cftsetindents{chap}{#2}{#3}
955   \else
956     \ifx\@cftemp\cftsecname \@cftsetindents{sec}{#2}{#3}
957   \else
958     \ifx\@cftemp\cftsubsecname \@cftsetindents{subsec}{#2}{#3}
959   \else
960     \ifx\@cftemp\cftsubsubsecname \@cftsetindents{subsubsec}{#2}{#3}
961   \else
962     \ifx\@cftemp\cftparaname \@cftsetindents{para}{#2}{#3}
963   \else
964     \ifx\@cftemp\cftsubparaname \@cftsetindents{subpara}{#2}{#3}
965   \else
966     \ifx\@cftemp\cftfigname \@cftsetindents{fig}{#2}{#3}
967   \else
968     \ifx\@cftemp\cftsubfigname \@cftsetindents{subfig}{#2}{#3}
969   \else
970     \ifx\@cftemp\cfttabname \@cftsetindents{tab}{#2}{#3}
971   \else
972     \ifx\@cftemp\cftsubtabname \@cftsetindents{subtab}{#2}{#3}
973   \else
974     \@cftsetindents{#1}{#2}{#3}
975   \fi
976   \fi
977   \fi
978   \fi
979   \fi
980   \fi
981   \fi
982   \fi
983   \fi

```

```

984 \fi
985 }
986

```

`\@cftsetindents` `\@cftsetindents{⟨X⟩}{⟨indent⟩}{⟨numwidth⟩}` 是 `\cftsetindents` 的内部版本 (internal version), 在本例中 `⟨X⟩` 是条目的内部 (简写) 名称。

```

987 \newcommand{\@cftsetindents}[3]{%
988   \setlength{\@nameuse{cft#1indent}}{#2}
989   \setlength{\@nameuse{cft#1numwidth}}{#3}
990 }
991

```

3.3 切换页码

`\@cftpnumoff` `\@cftpnumoff{⟨shorthand⟩}` 是关闭页码的主要工具。`⟨shorthand⟩` 参数是条目的简写名称 (shorthand name)(例如 `subsec` 表示 `subsection`)。宏重新定义 `\cftXnumfill` 命令, 这样就没有引线 (leader), 页码将被忽略。

```

992 \newcommand{\@cftpnumoff}[1]{%
993   \@namedef{cft#1fillnum}##1{%
994     \cftparfillskip\@nameuse{cft#1afterpnum}\par}}
995

```

`\cftchapname` 不幸的是, 早期的设计决定 (design decision) 是使用 `sec` 这样的缩写 (shorthands) `\cftsecname` 来表示 `section`。对于页面切换 (page switching), 我需要能够将缩写 (shorthands) `\cftsubsecname` 和普通写法 (longhands) 关联起来。

```

\cftsubsubsecname 996 \newcommand*{\cftchapname}{chapter}
\cftparaname       997 \newcommand*{\cftsecname}{section}
\cftsubparaname    998 \newcommand*{\cftsubsecname}{subsection}
\cftfigname        999 \newcommand*{\cftsubsubsecname}{subsubsection}
\cftsubfigname     1000 \newcommand*{\cftparaname}{paragraph}
\cfttabname        1001 \newcommand*{\cftsubparaname}{subparagraph}
\cftsubtabname     1002 \newcommand*{\cftfigname}{figure}
                  1003 \newcommand*{\cftsubfigname}{subfigure}
                  1004 \newcommand*{\cfttabname}{table}
                  1005 \newcommand*{\cftsubtabname}{subtable}
1006

```

`\cftpagenumbersoff` 用于关闭页码的用户级命令是 `\cftpagenumbersoff{⟨entry⟩}`，这里的 `⟨entry⟩` 是普通写法 (longhands)。此宏的主要任务是确定 `⟨entry⟩` 相应缩写名 (shorthand name)，然后调用 `\cftpnumoff` 完成工作。对于 `part` 和用户定义的条目 (user-defined entries)，普通写法和缩写的条目名称 (entry names) 相同。

```

1007 \DeclareRobustCommand{\cftpagenumbersoff}[1]{%
1008   \def\@cftemp{#1}
1009   \ifx\@cftemp\cftchapname
1010     \@cftpnumoff{chap}
1011   \else
1012     \ifx\@cftemp\cftsecname \@cftpnumoff{sec}
1013   \else
1014     \ifx\@cftemp\cftsubsecname \@cftpnumoff{subsec}
1015   \else
1016     \ifx\@cftemp\cftsubsubsecname \@cftpnumoff{subsubsec}
1017   \else
1018     \ifx\@cftemp\cftparaname \@cftpnumoff{para}
1019   \else
1020     \ifx\@cftemp\cftsubparaname \@cftpnumoff{subpara}
1021   \else
1022     \ifx\@cftemp\cftfigname \@cftpnumoff{fig}
1023   \else
1024     \ifx\@cftemp\cftsubfigname \@cftpnumoff{subfig}
1025   \else
1026     \ifx\@cftemp\cfttabname \@cftpnumoff{tab}
1027   \else
1028     \ifx\@cftemp\cftsubtabname \@cftpnumoff{subtab}
1029   \else
1030     \@cftpnumoff{#1}
1031   \fi
1032   \fi
1033   \fi
1034   \fi
1035   \fi
1036   \fi
1037   \fi
1038   \fi

```

```

1039     \fi
1040 \fi
1041 }
1042

```

`\cftpagenumberon` `\cftpagenumberon{<entry>}` 是用于反转相应的 `\cftpagenumbersoff` 的用户级命令。

```

1043 \DeclareRobustCommand{\cftpagenumberon}[1]{%
1044   \def\@cftmp{#1}
1045   \ifx\@cftmp\cftchapname
1046     \@cftpnumon{chap}
1047   \else
1048     \ifx\@cftmp\cftsecname \@cftpnumon{sec}
1049   \else
1050     \ifx\@cftmp\cftsubsecname \@cftpnumon{subsec}
1051   \else
1052     \ifx\@cftmp\cftsubsubsecname \@cftpnumon{subsubsec}
1053   \else
1054     \ifx\@cftmp\cftparaname \@cftpnumon{para}
1055   \else
1056     \ifx\@cftmp\cftsubparaname \@cftpnumon{subpara}
1057   \else
1058     \ifx\@cftmp\cftfigname \@cftpnumon{fig}
1059   \else
1060     \ifx\@cftmp\cftsubfigname \@cftpnumon{subfig}
1061   \else
1062     \ifx\@cftmp\cfttabname \@cftpnumon{tab}
1063   \else
1064     \ifx\@cftmp\cftsubtabname \@cftpnumon{subtab}
1065   \else
1066     \@cftpnumon{#1}
1067   \fi
1068   \fi
1069   \fi
1070   \fi
1071   \fi

```

```

1072         \fi
1073     \fi
1074     \fi
1075     \fi
1076 \fi
1077 }
1078

```

`\@cftpnumon \@cftpnumon{<shorthand>}` 是关闭页面编号的主要工具。`<shorthand>` 参数是条目 (entry) 的缩写名 (shorthand name)(例如 `subsec` 表示 `subsection`)。宏定义 `\cftXnumfill` 命令以对应于默认定义 (default definition)。

```

1079 \newcommand{\@cftpnumon}[1]{%
1080     \@namedef{cft#1fillnum}##1{%
1081         {\@nameuse{cft#1leader}}\nobreak
1082         \makebox[\@pnumwidth][\cftpnumalign]{
1083             \@nameuse{cft#1pagefont}##1}\@nameuse{cft#1afterpnum}\par
1084     }%
1085 }

```

3.4 实验性工具

本节中的代码是实验性的 (experimental), 但从某种意义上来说, 在未来这些功能 (future) 可能会被修改, 而不是代码无法工作。

`\cftchapterprecis` 这是实验性的 (experimental)。在调用它的位置处 `\cftchapterprecis{<text>}` 排版 `<text>`, 并将 `<text>` 添加到 `.toc` 文件。期望在 `\chapter` 命令后立即调用它。

```

1086 \newcommand{\cftchapterprecis}[1]{%
1087     \cftchapterprecishere{#1}
1088     \cftchapterprecistoc{#1}}

```

`\cftchapterprecishere` `\cftchapterprecishere{<text>}` 排版 `<text>`。期望在 `\chapter` 命令后立即调用它。首先添加一些负垂直空间 (negative vertical space), 使其更靠近章标题 (chapter heading)。

```

1089 \newcommand{\cftchapterprecishere}[1]{%
1090     \vspace*{-2\baselineskip}

```

在 `quote` 环境中使用斜体字体 (italic font) 排版其参数。

```

1091     \begin{quote}\textit{#1}\end{quote}}

```


`\cftchapterprecistoc` `\cftchapterprecistoc{<text>}` 将 `<text>` 添加到 `.toc` 文件。`<text>` 将使用斜体字体 (italic font) 在与 `\chapter` 标题 (heading) 的标题文本 (title text) 相同的页边距 (margins) 内排版。

```
1092 \newcommand{\cftchapterprecistoc}[1]{\addtocontents{toc}{%
```

启动一个组 (group) 以对分段 (paragraphing) 进行局部更改 (localize changes)。将左边距 (left margin) 设置为章缩进 (chapter indent) 加上章编号宽度 (chapter number width)。

```
1093   {\leftskip \cftchapindent\relax
```

```
1094    \advance\leftskip \cftchapnumwidth\relax
```

将右侧边距设置为 `\@tocrmarg`。

```
1095    \rightskip \@tocrmarg\relax
```

使用斜体字体 (italic font) 排版 `<text>`，然后确保段落结束 (使用局部间距 [local skips])。最后结束组 (group)，我们就完成了。

```
1096    \textit{#1}\protect\par}}}
```

```
1097
```

`\cftlocalchange` `\cftmakelocalchange{<file>}{<pnumwidth>}{<tocrmarg>}` 在 `<file>` 中创建一个条目 (entry) 来更改 `\@pnumwidth` 和 `\@tocrmarg` 的值。

```
1098 \newcommand{\cftlocalchange}[3]{%
```

```
1099   \addtocontents{#1}{\protect\cftsetpnumwidth{#2} \protect\cftsetrmarg{#3}}}
```

`\cftaddtitleline` `\cftaddtitleline{<file>}{<kind>}{<title>}{<page>}` 在 `<file>` 中添加一个 `\contentsline` 条目 (entry) 和给定的信息 (given information)。

```
1100 \newcommand{\cftaddtitleline}[4]{\addtocontents{#1}{%
```

```
1101   \protect\contentsline{#2}{#3}{#4}}}
```

`\cftaddnumtitleline` `\cftaddtitleline{<file>}{<kind>}{<num>}{<title>}{<page>}` 在 `<file>` 中添加一个 `\contentsline` 条目 (entry) 和给定的信息 (given information)。

```
1102 \newcommand{\cftaddnumtitleline}[5]{\addtocontents{#1}{%
```

```
1103   \protect\contentsline{#2}{\protect\numberline{#3}#4}{#5}}}
```

而且，如果已使用了旧的 `hyperref`，我们必须修复 (fix up) 这两个宏。

```
1104 \AtBeginDocument{%
```

```
1105   \ifpackageloaded{hyperref}{%
```

```
1106     \renewcommand{\cftaddtitleline}[4]{\addtocontents{#1}{%
```

```

1107     \protect\contentsline{#2}{#3}{#4}{\@currentHref}}
1108     \renewcommand{\cftaddnumtitleline}[5]{\addtocontents{#1}{%
1109         \protect\contentsline{#2}{\protect\numberline{#3}{#4}{#5}{\@currentHref}}}
1110     }{}}
1111 }
1112

```

`\@starttoc` 好了，下面是一个大致全面的列表 (list)，其中 `\@starttoc` 在 T_EX Live 2014 中被重新定义。

- amsart, amsbook, amsdtx, amsproc
- asect
- latex.ltx (当然)
- newfloat
- flowfram
- gmampulex (?), gmtypos
- hyperref, memhfixc
- ijmart
- scrartcl, scrbook, screprt
- scrwtfle
- tocbasic
- tocstyle
- memoir
- multitoc
- nccsect
- notoccite
- artikel3, boek3, rapport3,
- rerunfilecheck

- parskip
- pdfwin
- revtex4
- devanagari

如果我们想在这里重新定义 `\starttoc`，这就很困难了。上面的许多包/类不会与 `tocloft` 一起使用；另一方面，我们不想过分践踏 (trample) 他人的代码。

因此，对于我们自己的工作，至少在现在，让我们更加保守一些，如果它是标准的 L^AT_EX 版本，那么只需连接 (hook into) `\starttoc`。

```

1113 \def\starttoc@latex@orig#1{%
1114   \begingroup
1115     \makeatletter
1116     \@input{\jobname.#1}%
1117     \if@files
1118       \expandafter\newwrite\csname tf@#1\endcsname
1119       \immediate\openout \csname tf@#1\endcsname \jobname.#1\relax
1120     \fi
1121     \@nobreakfalse
1122 \endgroup}
1123 \ifx\starttoc\starttoc@latex@orig
1124   \def\starttoc#1{%
1125     \csname cft#1prehook\endcsname
1126     \begingroup
1127       \makeatletter
1128       \@input{\jobname.#1}%
1129       \if@files
1130         \expandafter\newwrite\csname tf@#1\endcsname
1131         \immediate\openout \csname tf@#1\endcsname \jobname.#1\relax
1132       \fi
1133       \@nobreakfalse
1134     \endgroup
1135     \csname cft#1posthook\endcsname}
1136 \newcommand\cfttocprehook{}
1137 \newcommand\cftlofprehook{}
1138 \newcommand\cftlotprehook{}

```

```

1139 \newcommand\cfttocposthook{}
1140 \newcommand\cftlofposthook{}
1141 \newcommand\cftlotposthook{}
1142 \else
1143 \PackageWarning{tocloft}{
1144   \string\@starttoc\space has already been redefined; tocloft bailing out.
1145 }
1146 \fi

    这个宏包结束。

1147 </usc>

```

参考文献

- [Coc95] Steven Douglas Cochran. *The subfigure package*. March 1995. (Available from CTAN as file `subfigure.dtx`)
- [Dru99] Jean-Pierre Drucbert. *The minitoc package*. August 1999. (Available from CTAN in subdirectory `/minitoc`)
- [GMS94] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley Publishing Company, 1994.
- [Lin97] Ulf A. Lindgren. *FncyChap V1.11*. April 1997. (Available from CTAN in subdirectory `/fncychap`)
- [Lin95] Anselm Lingnau. *An Improved Environment for Floats*. March 1995. (Available from CTAN in subdirectory `/float`)
- [Wil96a] Peter Williams. *Algorithms*. April 1996. (Available from CTAN in subdirectory `/algorithm`)
- [Wil96b] Peter R. Wilson. *LaTeX for standards: The LaTeX package files user manual*. NIST Report NISTIR, June 1996.
- [Wil00] Peter R. Wilson. *The tocbibind package*. March 2000. (Available from CTAN as file `tocbibind.dtx`)
- [Wil01] Peter R. Wilson. *The ccaption package*. March 2001. (Available from CTAN as file `ccaption.dtx`)

索引

斜体数字 (*italicnumbers*) 表示描述相应条目所在的页面 (pages); 带下划线的数字 (*numbersunderlined*) 表示定义 (definition) 所在的代码行 (code lines); 其它数字表示使用该条目所在的代码行。

符号	964, 966, 968, 110, <u>119</u>
\@afterheading	970, 972, 1008,	\@cftnctocfalse 40
. 132, 548, 651, 926	1009, 1012,	\@cftnctoctrue 41
\@afterindenttrue .	1014, 1016,	\@cftpagestyle .. <u>48</u> ,
..... 198, 255,	1018, 1020,	126, 542, 645, 920
303, 349, 390,	1022, 1024,	\@cftpnumoff .. <u>992</u> ,
431, 472, 574,	1026, 1028,	1010, 1012,
677, 714, 758, 826	1044, 1045,	1014, 1016,
\@cftasnum . <u>23</u> , 167,	1048, 1050,	1018, 1020,
<u>168</u> , 203, 260,	1052, 1054,	1022, 1024,
308, 354, 395,	1056, 1058,	1026, 1028, 1030
436, 477, 579,	1060, 1062, 1064	\@cftpnumon .. 1046,
682, 719, 763, 831	\@cfthaschapterfalse 19	1048, 1050,
\@cftasnumb <u>23</u> , 167,	\@cfthaschaptertrue 25	1052, 1054,
<u>168</u> , 204, 261,	\@cfthaspartfalse . 173	1056, 1058,
309, 355, 396,	\@cfthasparttrue .. 173	1060, 1062,
437, 478, 580,	\@cftifundefined ..	1064, 1066, <u>1079</u>
683, 720, 764, 832 <u>2</u> , 18, 20, 173	\@cftquit <u>9</u> , 23, 27
\@cftbsnum . <u>23</u> , 167,	\@cftkomafalse 12	\@cftsetindents ...
<u>168</u> , 202, 259,	\@cftkomatrue 13, 14, 15 954, 956,
307, 353, 394,	\@cftl@subfig	958, 960, 962,
435, 476, 578, <u>707</u> , 797, 801	964, 966, 968,
681, 718, 762, 830	\@cftl@subtab	970, 972, 974, <u>987</u>
\@cftdobiblof . <u>507</u> , 527 <u>751</u> , 797, 801	\@cftsetsubfig
\@cftdobiblot . <u>610</u> , 630	\@cftmakeZtitle ... <u>913</u> <u>728</u> , 796, 800
\@cftdobibtoc .. <u>80</u> , 112	\@cftmakeloftitle .	\@cftsetsubtab
\@cftdopartfalse .. 176 <u>525</u> , <u>535</u> <u>771</u> , 796, 800
\@cftdoparttrue 179, 183	\@cftmakelottitle .	\@cftsubfigoptfalse 43
\@cfttemp <u>952</u> , <u>953</u> , <u>956</u> , <u>628</u> , <u>638</u>	\@cftsubfigopttrue . 44
958, 960, 962,	\@cftmaketoctitle .	\@cfttitlesectrue .. 17

\cfttocbibindfalse	30	898, 907, 916,	\addtocontents				
\cfttocbibindtrue	30	918, 922, 923,	6, 1092,				
\cfttocfinish	925, 931, 932,	1099,	1100,				
.....	66, 116,	939, 940, 988,	1102,	1106, 1108				
531, 634, 900, 910		989, 994, 1081, 1083	\AtBeginDocument	..				
\cfttocstart	66, 106,	\@nobreakfalse	17, 29, 94, 518,				
521, 624, 892, 903		211, 215, 1121, 1133		621, 797, 799, 1104				
\@currentHref	1107, 1109	\@nobreaktrue	.	210, 214				
\@dotsep	4	\@pnumwidth	.	4, 150,			
\@dottedtocline	6	238, 284, 340,	\baselineskip	1090		
\@eha	818	381, 422, 463,					
\@firstoftwo	5	504, 602, 705,	\c@lofdepth	..	569, 709		
\@highpenalty	.	189, 250	746, 789, 870, 1082	\c@lotdepth	..	672, 753		
\@ifclassloaded	...		\@restonecolfalse	..	71	\c@tocdepth	...	177,
.....	13, 14, 15		\@restonecoltrue	...	69	181, 249, 293,		
\@ifpackagelater	...	32	\@secondoftwo	7	344, 385, 426, 467		
\@ifpackageloaded	.		\@secpenalty	120, 191,	\c@X	805	
.....	17, 30, 1105		297, 536, 639, 914	\cftaddnumtitleline				
\@ifundefined	.	805, 809	\@starttoc	21, 1102, 1108		
\@namedef	820,	97, 98, 114, 529,	\cftaddtitleline	..			
859, 860, 861,			632, 899, 908, 1113	...	21, 1100, 1106			
862, 863, 864,			\@starttoc@latex@orig	\cftafterloftitle	.			
865, 866, 867,			1113, 1123	544, 559		
868, 880, 884,			\@tempa	155, 157	\cftafterloftitleskip		
887, 891, 902,			\@tempb	156, 157	547, 550	
913, 934, 936,			\@tocextra	.	87, 514, 617	\cftafterlottitle	.	
942, 944, 947,			\@tocrmarg	..	4, 151,	647, 662	
948, 949, 993, 1080			196, 253, 301,	\cftafterlottitleskip				
\@nameuse	821,	347, 388, 429,	650, 653			
822, 823, 826,			470, 572, 675,	\cftaftertoctitle	.			
829, 834, 835,			712, 756, 824, 1095	128, 142			
840, 844, 845,				\cftaftertoctitleskip				
847, 848, 850,				131, 133			
851, 853, 854,			\addcontentsline	..	\cftafterZtitle	10, 947		
856, 857, 864,			3, 84, 87,	\cftafterZtitleskip			
869, 870, 871,			511, 514, 614, 617	10, 928			

\cftbeforechapskip	\cftchapdotsep 267	\cftfigindent
. 251 , 267	\cftchapfillnum 264 , 267 571 , 574 , 586
\cftbeforefigskip .	\cftchapfont . 263 , 267	\cftfigleader 586
. 570 , 586	\cftchapindent	\cftfigname
\cftbeforeloftitleskip	252 , 255 , 267 , 1093	966 , 996 , 1022 , 1058
. 538 , 540 , 550	\cftchapleader 267	\cftfignumwidth 577 , 586
\cftbeforelottitleskip	\cftchapname	\cftfigpagefont . . . 586
. 641 , 643 , 653	953 , 996 , 1009 , 1045	\cftfigpresnum 578 , 586
\cftbeforeparaskip	\cftchapnumwidth . .	\cftlocalchange 21 , 1098
. 427 , 442 258 , 267 , 1094	\cftlofposthook . . 1140
\cftbeforepartskip	\cftchappagefont . . 267	\cftlofprehook . . . 1137
. 193 , 221	\cftchappresnum 259 , 267	\cftloftitlefont . .
\cftbeforesecskip .	\cftchapterprecis 544 , 559
. 295 , 298 , 314 20 , 1086	\cftlotposthook . . 1141
\cftbeforesubfigskip	\cftchapterprecishere	\cftlotprehook . . . 1138
. 710 , 729 20 , 1087 , 1089	\cftlottitlefont . .
\cftbeforesubparaskip	\cftchapterprecistoc 647 , 662
. 468 , 483 20 , 1088 , 1092	\cftmarklof . . . 52 , 545
\cftbeforesubsecskip	\cftdot 11 , 153	\cftmarklot . . . 52 , 648
. 345 , 360	\cftdotfill . . . 153 ,	\cftmarktoc . . . 52 , 129
\cftbeforesubsubsecskip	230 , 278 , 325 ,	\cftmarkZ 10 , 883
. 386 , 401	333 , 375 , 416 ,	\cftnodots . 11 , 156 ,
\cftbeforesubtabskip	457 , 498 , 596 ,	163 , 231 , 279 , 334
. 754 , 772	699 , 739 , 782 , 864	\cftpagenumbersoff
\cftbeforetabskip .	\cftdotsep . 11 , 163 , 16 , 1007
. 673 , 689	326 , 376 , 417 ,	\cftpagenumberson .
\cftbeforetoctitleskip	458 , 499 , 597 , 16 , 1043
. 122 , 124 , 133	700 , 740 , 783 , 863	\cftparaafterpnum . 442
\cftbeforeXskip 12 , 839	\cftfigafterpnum . . 586	\cftparaaftersnum .
\cftbeforeZtitleskip	\cftfigaftersnum 436 , 442
. 10 , 928 579 , 586	\cftparaaftersnumb
\cftchapafterpnum . 267	\cftfigaftersnumb 437 , 442
\cftchapaftersnum 580 , 586	\cftparadotsep 442
. 260 , 267	\cftfigdotsep 586	\cftparafillnum 440 , 442
\cftchapaftersnumb	\cftfigfillnum 583 , 586	\cftparafont . 439 , 442
. 261 , 267	\cftfigfont . . 582 , 586	

\cftparrindent	\cftsecaftersnum ..	\cftsubfigpagefont <u>729</u>
..... <u>428</u> , <u>431</u> , <u>442</u> <u>308</u> , <u>314</u>	\cftsubfigpresnum .
\cftparaleader <u>442</u>	\cftsecaftersnumb <u>718</u> , <u>729</u>
\cftparaname <u>309</u> , <u>314</u>	\cftsubparaafterpnum
<u>962</u> , <u>996</u> , <u>1018</u> , <u>1054</u>	\cftsecdotsep <u>314</u> <u>483</u>
\cftparanumwidth ..	\cftsecfillnum <u>312</u> , <u>314</u>	\cftsubparaaftersnum
..... <u>434</u> , <u>442</u>	\cftsecfont .. <u>311</u> , <u>314</u> <u>477</u> , <u>483</u>
\cftparapagefont .. <u>442</u>	\cftsecindent	\cftsubparaaftersnumb
\cftparapresnum <u>435</u> , <u>442</u> <u>300</u> , <u>303</u> , <u>314</u> <u>478</u> , <u>483</u>
\cftparfillskip <u>165</u> , <u>994</u>	\cftsecleader <u>314</u>	\cftsubparadotsep . <u>483</u>
\cftparskip	\cftsecname	\cftsubparafillnum
<u>11</u> , <u>91</u> , <u>96</u> , <u>100</u> ,	<u>956</u> , <u>996</u> , <u>1012</u> , <u>1048</u> <u>481</u> , <u>483</u>
<u>109</u> , <u>524</u> , <u>627</u> , <u>906</u>	\cftsecnumwidth <u>306</u> , <u>314</u>	\cftsubparafont <u>480</u> , <u>483</u>
\cftpartafterpnum . <u>221</u>	\cftsecpagefont ... <u>314</u>	\cftsubparaindent .
\cftpartaftersnum .	\cftsecpresnum <u>307</u> , <u>314</u> <u>469</u> , <u>472</u> , <u>483</u>
..... <u>203</u> , <u>221</u>	\cftsetindents . <u>13</u> , <u>951</u>	\cftsubparaleader . <u>483</u>
\cftpartaftersnumb	\cftsetpnumwidth ..	\cftsubparaname ...
..... <u>204</u> , <u>221</u> <u>11</u> , <u>150</u> , <u>1099</u>	<u>964</u> , <u>996</u> , <u>1020</u> , <u>1056</u>
\cftpartdotsep <u>221</u>	\cftsetrmarg	\cftsubparanumwidth
\cftpartfillnum <u>207</u> , <u>221</u> <u>11</u> , <u>150</u> , <u>1099</u> <u>475</u> , <u>483</u>
\cftpartfont . <u>206</u> , <u>221</u>	\cftsubfigafterpnum <u>729</u>	\cftsubparapagefont <u>483</u>
\cftpartindent	\cftsubfigaftersnum	\cftsubparapresnum
..... <u>195</u> , <u>198</u> , <u>221</u> <u>719</u> , <u>729</u> <u>476</u> , <u>483</u>
\cftpartleader <u>221</u>	\cftsubfigaftersnumb	\cftsubsecafterpnum
\cftpartnumwidth <u>720</u> , <u>729</u> <u>360</u> , <u>382</u>
..... <u>201</u> , <u>221</u>	\cftsubfigdotsep .. <u>729</u>	\cftsubsecaftersnum
\cftpartpagefont .. <u>221</u>	\cftsubfigfillnum <u>354</u> , <u>360</u>
\cftpartpresnum <u>723</u> , <u>729</u>	\cftsubsecaftersnumb
..... <u>202</u> , <u>206</u> , <u>221</u>	\cftsubfigfont <u>722</u> , <u>729</u> <u>355</u> , <u>360</u>
\cftpnumalign <u>11</u> , <u>152</u> ,	\cftsubfigindent ..	\cftsubsecdotsep .. <u>360</u>
<u>238</u> , <u>284</u> , <u>340</u> , <u>711</u> , <u>714</u> , <u>729</u>	\cftsubsecfillnum .
<u>381</u> , <u>422</u> , <u>463</u> ,	\cftsubfigleader .. <u>729</u> <u>358</u> , <u>379</u>
<u>504</u> , <u>602</u> , <u>705</u> ,	\cftsubfigname	\cftsubsecfont <u>357</u> , <u>360</u>
<u>746</u> , <u>789</u> , <u>870</u> , <u>1082</u>	<u>968</u> , <u>996</u> , <u>1024</u> , <u>1060</u>	\cftsubsecindent ..
\cftsecafterpnum .. <u>314</u>	\cftsubfignumwidth <u>346</u> , <u>349</u> , <u>360</u>
 <u>717</u> , <u>729</u>	

<code>\cftsubsecleader</code> 764, 772	<code>\cftXafterpnum</code> .	13, 859
..... 360, 380	<code>\cftsubtabdotsep</code> ..	<code>\cftXaftersnum</code> .	13, 859
<code>\cftsubsecname</code> 739, 772	<code>\cftXaftersnumb</code>	13, 859
958, 996, 1014, 1050	<code>\cftsubtabfillnum</code> .	<code>\cftXdotsep</code> ...	13, 859
<code>\cftsubsecnumwidth</code> 767, 772	<code>\cftXfillnum</code>	868
..... 352, 360	<code>\cftsubtabfont</code> 766, 772	<code>\cftXfont</code>	13, 859
<code>\cftsubsecpagefont</code>	<code>\cftsubtabindent</code> ..	<code>\cftXindent</code> ...	12, 841
..... 360, 382 755, 758, 772	<code>\cftXleader</code> ...	13, 859
<code>\cftsubsecpresnum</code> .	<code>\cftsubtableader</code> ..	<code>\cftXnumwidth</code> ..	12, 841
..... 353, 360	<code>\cftsubtabname</code>	<code>\cftXpagefont</code> ..	13, 859
<code>\cftsubsubsecafterpnum</code>	972, 996, 1028, 1064	<code>\cftXpresnum</code> ..	13, 859
..... 401	<code>\cftsubtabnumwidth</code>	<code>\cftZposthook</code> ..	22, 948
<code>\cftsubsubsecaftersnum</code> 761, 772	<code>\cftZprehook</code> ..	22, 948
..... 395, 401	<code>\cftsubtabpagefont</code> 772	<code>\cftZtitlefont</code> .	10, 928
<code>\cftsubsubsecaftersnumb</code>	<code>\cftsubtabpresnum</code> .	<code>\chapter</code>	894
..... 396, 401 762, 772	<code>\contentsline</code> 4, 1101,	
<code>\cftsubsubsecdotsep</code> 401	<code>\cfttabafterpnum</code> ..	1103, 1107, 1109	
<code>\cftsubsubsecfillnum</code>	<code>\cfttabaftersnum</code> ..	<code>\contentsname</code>	
..... 399, 401 682, 689	53, 60, 84, 87, 128	
<code>\cftsubsubsecfont</code> .	<code>\cfttabaftersnumb</code> .	<code>\csname</code> .	4, 813, 814,
..... 398, 401 683, 689	830, 831, 832,	
<code>\cftsubsubsecindent</code>	<code>\cfttabdotsep</code>	839, 841, 842,	
.... 387, 390, 401	<code>\cfttabfillnum</code> 686, 689	928, 929, 1118,	
<code>\cftsubsubsecleader</code> 401	<code>\cfttabfont</code> ..	1119, 1125,	
<code>\cftsubsubsecname</code> .	<code>\cfttabindent</code>	1130, 1131, 1135	
960, 996, 1016, 1052 674, 677, 689		
<code>\cftsubsubsecnumwidth</code>	<code>\cfttableader</code>		
..... 393, 401	<code>\cfttabname</code>		
<code>\cftsubsubsecpagefont</code>	970, 996, 1026, 1062		
..... 401	<code>\cfttabnumwidth</code> 680, 689		
<code>\cftsubsubsecpresnum</code>	<code>\cfttabpagefont</code> ...		
..... 394, 401	<code>\cfttabpresnum</code> 681, 689		
<code>\cftsubtabafterpnum</code> 772	<code>\cfttocposthook</code> ..		
<code>\cftsubtabaftersnum</code>	<code>\cfttocprehook</code> ...		
..... 763, 772	<code>\cfttoctitlefont</code> ..		
<code>\cftsubtabaftersnumb</code> 128, 142		

D

`\DeclareRobustCommand`
..... 1007, 1043

E

`\edef`

813

`\endcsname` 4, 813, 814,

830, 831, 832,

839, 841, 842,

928, 929, 1118,

1119, 1125,	\ifcompatibility . 213	428, 438, 469,
1130, 1131, 1135	\ifdotoclof 508	479, 571, 581,
\endinput 23	\ifdotoclot 611	674, 684, 711,
\ext@Z 880	\ifdotocloc 81	721, 755, 765,
	\iffiles . . . 1117, 1129	823, 833, 1093, 1094
H	\ifrestonecol 76	\listfigurename . . .
\hb@xt@ 167	\if@twocolumn 68	55, 62, 511, 514, 544
\Huge 143,	\immediate . . . 1119, 1131	\listoffigures . . . 9, 518
560, 663, 936, 944	\interlinepenalty .	\listoftables . . . 9, 621
	. 127, 199, 256,	\listofX 17, 890
I	304, 350, 391,	\listtablename . . .
\if@bibchapter	432, 473, 543,	57, 64, 614, 617, 647
. 82, 509, 612	575, 646, 678,	\lofdepth 605
\if@cftdopart 171	715, 759, 827, 921	\lotdepth 605
\if@cfthaschapter 9,		
67, 75, 121, 135,	L	M
142, 178, 182,	\l@ 837	\makeatletter 1115, 1127
188, 209, 247,	\l@chapter 247	\makebox 238, 284, 340,
267, 294, 320,	\l@figure 568	381, 422, 463,
364, 405, 446,	\l@paragraph 425	504, 602, 705,
487, 537, 552,	\l@part 171	746, 789, 870, 1082
559, 640, 655,	\l@section 292	\MakeUppercase . . .
662, 893, 915, 930	\l@subfigure 708	. . . 53, 55, 57, 888
\if@cfthaspart	\l@subparagraph . . . 466	\MessageBreak 34, 35, 36
. 172, 174, 221	\l@subsection 343	\mkern 160
\if@cftkoma	\l@subsubsection . . . 384	
11, 58, 144, 147,	\l@subtable 752	N
206, 241, 287,	\l@table 671	\newif . . . 10, 11, 16,
561, 564, 664,	\l@X 18, 820	28, 40, 43, 171, 172
667, 883, 933, 941	\Large 146, 563, 666	\newlistentry
\if@cftnctoc	\large 226,	. 18, 804, 876, 878
40, 95, 519, 622, 890	230, 232, 243, 244	\newlistof 16, 874
\if@cftsubfigopt . .	\leaders 160	\newwrite . . . 1118, 1130
. 43, 605, 795, 799	\leftskip 195,	\noexpand 814
\if@cfttitlesec 16, 206	205, 252, 262,	\normalfont . . . 143,
\if@cfttocbibind . .	300, 310, 346,	146, 324, 325,
. 28, 111, 526, 629	356, 387, 397,	327, 371, 375,

377, 412, 416,	\parindent	\size@chapter
418, 453, 457,	. 109, 198, 255,	. 144, 561, 664, 934
459, 494, 498,	303, 349, 390,	\size@section
500, 560, 563,	431, 472, 524,	. 147, 564, 667, 942
592, 596, 598,	574, 627, 677,	\string 1144
663, 666, 695,	714, 758, 826, 906	
699, 701, 735,	\parskip 100,	T
739, 741, 778,	109, 524, 627, 906	\tableofcontents . 9, <u>94</u>
782, 784, 859,	\phantomsection . . .	\textit 1091, 1096
864, 865, 936, 944 <u>78</u> , 83, 86,	\theX 18, <u>805</u>
\numberline 5, <u>166</u>	510, 513, 613, 616	\thispagestyle 49
O	\providecommand . . .	\toclevel@subfig .. <u>729</u>
\OLD@starttoc .. 97, 101	66, 74, 78, 153,	\toclevel@subfigure
\onecolumn 69	154, 163, 743, 786 709, 743
\openout 1119, 1131		\toclevel@subtable
P	S 753, <u>772</u>
\PackageError 818	\sectfont 144,	\toclevel@X <u>867</u>
\PackageInfo . . . 24, 26	147, 243, 244,	\tocloftpagestyle 9, <u>48</u>
\PackageWarning . . .	288, 561, 564,	\twocolumn 76
. 21, 33, 809, 1143	664, 667, 934, 942	Z
	\section 896	\Zdepth 11, 17, <u>880</u>