

# 在 L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 中使用 ltxdoc.dtx 文件 \*

## 包含 ltxdoc.cls 的代码

David Carlisle

黄旭华 翻译

2023/03/28

This file is maintained by the L<sup>A</sup>T<sub>E</sub>X Project team.  
Bug reports can be opened (category `latex`) at  
<https://latex-project.org/bugs.html>.

## 目 录

1 L <sup>A</sup> T <sub>E</sub> X 的源文档	2
2 定制	2
3 选项	3
4 选项处理	4
5 局部配置	4
6 加载 <code>article</code> 和 <code>doc</code>	4
7 有用的缩写	5

---

\*该文件的版本号为 v2.1j, 最新修订日期为 2023/03/28。

8 旧注释	6
9 DocInclude	8
10 配置文件	11

## 1 L<sup>A</sup>T<sub>E</sub>X 的源文档

该类文件 (class file) 用于记录 L<sup>A</sup>T<sub>E</sub>X 的源文件 (source files)。然而, 您可能会发现它通常作为一个类 (class) 用于排版以 “doc” 格式生成的文件的文档 (documentation)。

标准发行版 (standard distribution) 中的每个文档文件 (documented file) 都带有扩展名 `dtx`。docstrip 系统将从源代码中提取适当的类包 (class package) 或 `initex` 文件。可以使用 L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 直接处理每个 `dtx` 文件, 例如:

```
latex2e docclass.dtx
```

将生成 Class 和 package 接口 (interface) 的文档。

如果您对 `sources2e.tex` 文件进行了 L<sup>A</sup>T<sub>E</sub>X 编译, 则用于生成 L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 格式的每个文件 (即不包括标准类和包) 将一起打印在一个文档中。这样做的好处是, 可以生成所有源文件的宏使用情况 (macro usage) 的完整索引。

如果需要自定义这些文件的排版, 有两个选择:

- 在输入源文件 (source file) 之前, 您可以使用 DOCSTRIP 和 “driver” 模块 (module) 来提取一个小的 L<sup>A</sup>T<sub>E</sub>X 文件, 可以编辑该文件以使用所需的任何类或包的选项。
- 您可以创建一个 `ltxdoc.cfg` 文件。无论何时使用 `ltxdoc` 类, 都会读取此配置文件, 因此可以用该配置文件来自定义所有源文件 (source files) 的排版, 而无需编辑许多小驱动程序文件 (small driver files)。

第二种选择通常更方便。下一节将讨论各种可能性。

## 2 定制

最简单的定制 (customisation) 形式是将更多选项传递给 `ltxdoc` 加载的 `article` 类。例如, 如果希望将所有文档格式化为 A4 纸, 请在 `ltxdoc.cfg` 中添加下面这一行:

```
\PassOptionsToClass{a4paper}{article}
```

所有源文件 (source files) 分为两部分, 以 `\MaybeStop` 分隔。第一部分 (应该) 包含 “user(用户)” 文档。第二部分是源代码 (source code) 的完整文档列表。doc 宏包提供了 `\OnlyDescription` 命令, 该命令禁止显示代码列表。这也可以在配置文件中使用时, 但由于稍后读取 doc 包, 因此必须必须将

`\OnlyDescription` 的执行 (execution) 延迟到读取 doc 包之后。最简单的方法是使用 `\AtBeginDocument`。因此, 您可以将以下内容放在 `ltxdoc.cfg` 中。

```
\AtBeginDocument{\OnlyDescription}
```

如果您的文档依赖于使用旧版的 doc, 则可以通过传递 doc2 选项来请求类加载 doc 2 版。

如果处理了完整的源代码清单 `sources2e.tex`, 那么默认情况下会生成索引 (index) 和更新历史记录 (change history), 但是通常不会为单个文件生成索引。

例如, 考虑 `ltclass.dtx`, 它包含新类和包接口命令的源代码。在没有 `cfg` 文件的情况下, 将生成一个 19 页的文档。使用上面的配置, 就生成一个可读性稍微好一点的文档 (4 页)。

相反, 如果您真的想详细阅读源代码清单, 您会想要一个索引。同样, 可以使用 doc 包提供的索引命令, 但必须延迟其执行。

```
\AtBeginDocument{\CodelineIndex\EnableCrossrefs}
\AtEndDocument{\PrintIndex}
```

doc 包使用 `MakeIndex` 和 `gind` 样式写入要排序的索引文件 (index files), 因此可以使用以下命令

```
makeindex -s gind.ist ltclass.idx
```

然后重新运行  $\text{\LaTeX}$

与打印更改历史记录 (print a Change history) 类似, 您可以添加

```
\AtBeginDocument{\RecordChanges}
\AtEndDocument{\PrintChanges}
```

到 `ltxdoc.cfg`, 并将 `MakeIndex` 与以下命令一起使用

```
makeindex -s gglo.ist -o ltclass.gls ltclass.glo
```

最后, 如果您不想列出 `source2e.tex` 的所有节 (sections), 则只能在 `cfg` 文件中使用 `\includeonly`:

```
\includeonly{ltvers,ltboxes}
```

### 3 选项

1  $\langle *class \rangle$

```

2 \DeclareOption{a5paper}{\@latexerr{Option not supported}%
3   {}}

```

阻止加载配置文件 (config file)。

```

4 \newif\ifltxdoc@load@cfg@ \ltxdoc@load@cfg@true
5 \DeclareOption{nocfg}{\ltxdoc@load@cfg@false}

```

支持将 doc 回滚到版本 2:

```

6 \let\ltxdoc@doc@version\@empty % 默认情况下使用当前版本
7 \DeclareOption{doc2}{\def\ltxdoc@doc@version{=v2}}
8 \DeclareOption*{%
9   \PassOptionsToClass {\CurrentOption}{article}}

```

## 4 选项处理

```

10 \ProcessOptions

```

## 5 局部配置

导入局部配置文件 (如果存在)。

```

11 \ifltxdoc@load@cfg@
12 \InputIfFileExists{ltxdoc.cfg}
13     {\typeout{*****^^J%
14               * 使用局部配置文件 ltxdoc.cfg      ^^J%
15               *****}}
16     {}
17 \else
18   \typeout{*****^^J%
19             * 忽略局部配置文件          ^^J%
20             *****}
21 \fi

```

## 6 加载 article 和 doc

```

22 \LoadClass{article}

```

默认情况下, 加载当前 doc 版本 (\ltxdoc@doc@version 为空)。如果给定选项 doc2, 则加载版本 2 (\ltxdoc@doc@version 包含 =v2)。

```

23 \RequirePackage{doc}[\ltxdoc@doc@version]

```

使 | 成为“短 verb”字符, 但不能在文档前言中, 因为活动字符 (active character) 可能会干扰加载的包。

```
24 \AtBeginDocument{\MakeShortVerb{\|}}
```

由于“doc”文档往往有很多等宽材料 (monospaced material), 请设置一些 tt 替换 (substitutions) 以便静默地进行。

```
25 \DeclareFontShape{OT1}{cmtt}{bx}{n}{<-> ssub * cmtt/m/n}{}
26 \DeclareFontFamily{OMS}{cmtt}{\skewchar\font 48} % '60
27 \DeclareFontShape{OMS}{cmtt}{m}{n}{<-> ssub * cmsy/m/n}{}
28 \DeclareFontShape{OMS}{cmtt}{bx}{n}{<-> ssub * cmsy/b/n}{}

```

此替换 (substitution) 位于标准 fd 文件中, 但不是静默的 (silent)。

```
29 \DeclareFontShape{OT1}{cmss}{m}{it}{<->ssub*cmss/m/sl}{}
30 \CodelineNumbered
31 \DisableCrossrefs

```

稍微增加文本宽度, 以便在 macrocode 环境中使用标准字体 (standard fonts) 以显示 72 栏代码。

```
32 \setlength{\textwidth}{355pt}
```

对于较长的命令名, 请稍微增加边距宽度 (marginpar width)。并将左侧边距 (left margin) 增加相似的量。

```
33 \addtolength\marginparwidth{30pt}
34 \addtolength\oddsidemargin{20pt}
35 \addtolength\evensidemargin{20pt}
36 \setcounter{StandardModuleDepth}{1}

```

## 7 有用的缩写

`\cmd{\foo}` 按原样打印 `\foo`。它可以在移动参数 (moving arguments) 中使用。它不能用于记录定义为“`\outer`”的命令, 也不能用于诸如 `\iftrue` 或由 `\newif` 定义的条件。doc 包中已经提供的 `\cs{\foo}` 也会为喜欢这种语法的人打印 `\foo`。(第二种形式可用于记录所有类型的命令, 因此上述限制不适用)

```
\cmd
\cs 37 %\DeclareRobustCommand\cs[1]... % 稍后定义
38 %\def\cmd#1{\cs{\expandafter\cmd@to\cs\string#1}}% 不能与新的\cs 一起用
39 \def\cmd#1{\texttt{\char`\expandafter\cmd@to\cs\string#1}}
40 \def\cmd@to\cs#1#2{\char\number`#2\relax}

```

`\marg \marg{text}` 打印 `{\text}`, “强制参数 (mandatory argument)”。

```
41 \providecommand\marg[1]{%
42 {\ttfamily\char`\}\meta{#1}{\ttfamily\char`\}}

```

`\oarg \oarg{text}` 打印 [*text*], “可选参数 (optional argument)”。

```
43 \providecommand\oarg[1]{%  
44   {\ttfamily[]\meta{#1}{\ttfamily}}}
```

`\parg \parg{te,xt}` 打印 (*te,xt*), “图像模式参数 (picture mode argument)”。

```
45 \providecommand\parg[1]{%  
46   {\ttfamily{}\meta{#1}{\ttfamily}}}
```

## 8 旧注释

L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 源包含大量继承自 L<sup>A</sup>T<sub>E</sub>X 2.09 的代码。此代码中的注释 (comments) 不是为了排版而设计的, 并且不包含必要的 L<sup>A</sup>T<sub>E</sub>X 标记。oldcomments 环境对这些注释进行排版, 自动检测何时出现控制序列 (control sequence), 并隐式 (implicitly) 添加 `\verb`。这个过程不会产生特别漂亮的页面, 但它允许我们完整地记录新节 (new sections), 并对所有旧代码进行某种形式的排版注释 (typeset comments)。

扫描控件名称 (control names) 并将其放入 tt。实际上会 (错误地) 扫描以前的 `\\`, 但这并不重要, 因为在实践中, 这几乎从来没有后面跟着字母 (letter)。(即 `\\foo`) 会将 `foo` 放在 `\ttfamily` 中。

```
47 \def\oc@scan#1{%  
48   \ifx\oc@bslash#1%  
49     \egroup\let\next\oc@bslash\else  
50   \ifcat a\noexpand#1%  
51     #1\let\next\oc@scan\else  
52   \ifx\oc@percent#1%  
53     \def\next{\char`%\egroup}%  
54   \else  
55     #1\let\next\egroup  
56   \fi\fi\fi\next}  
  
57 \def\oc@bslash{\bgroup\oc@ttf\char`\\ \oc@scan}%  
  
58 \def\oc@verb#1{%  
59   \catcode`#1\active  
60   \uccode`~`#1%  
61   \uppercase{\def~{\oc@ttf\char`#1}}}  
  
62 \begingroup  
63   \obeyspaces%  
64   \catcode`\/=\catcode`\\
```

```

65 /catcode`\ /active
66 /catcode`<=/catcode`{%
67 /catcode`>=/catcode`}%
68 /catcode`{/active%
69 /catcode`}/active%
70 /gdef/oldc< \end{oldcomments}>%
71 /gdef/begmac< \begin{macrocode}>%
72 /gdef/obs</def <</oc@ttf/ >>>%
73 /endgroup%

74 \begingroup
75 \catcode`\ /=\catcode`\
76 \catcode`\ =13
77 /catcode`/|=/catcode`/%
78 /catcode`/%=13
79 /gdef/oldcomments{|
80 /makeatletter
81 /let/do/oc@verb/dospecials
82 /frenchspacing/@vobeyspaces/obs
83 /raggedright
84 /oc@verb/>|
85 /oc@verb/<|
86 /let\ /oc@bslash
87 /let%/oc@percent
88 /obeylines
89 /parindent/z@
90 /ttfamily/expandafter/let/expandafter/oc@ttf/the/font
91 /rmfamily
92 /textit{Historical \LaTeX/,2.09 comments (not necessarily accurate any more):}
93 /hfuzz/maxdimen
94 }
95 /endgroup

96 \begingroup
97 \sloppy%
98 \obeylines%
99 \gdef\oc@percent#1^~M{%
100 \ifvmode%
101 \def\commentline{#1}%
102 \ifx\commentline\oldc%
103 \textit{End of historical \LaTeX/,2.09 comments.}
104 \end{oldcomments}%

```



```

105 \else%
106 \ifx\commentline\begmac%
107 \begin{macrocode}%
108 \else%
109 \leavevmode%
110 #1^~M%
111 \fi\fi%
112 \else%
113 {\oc@ttf\char`\}%#1^~M%
114 \fi}%
115 \endgroup%

```

## 9 DocInclude

```

116 \@addtoreset{CodelineNo}{part}

```

`\DocInclude` 或多或少与 `\include` 完全相同, 但在 `dtx` 文件上使用 `\DocInput`, 而在 `tex` 文件上不使用 `\input`。

```

117 \def\partname{File}

118 \newcommand*{\DocInclude}[1]{%
119 \relax
120 \clearpage
121 \docincludeaux
122 \IfFileExists{#1.fdd}%
123 {\def\currentfile{#1.fdd}}%
124 {\def\currentfile{#1.dtx}}%
125 \ifnum\@auxout=\@partaux
126 \@latexerr{\string\include\space cannot be nested}\@eha
127 \else
128 \set@curr@file{#1}%
129 \edef\@curr@file{\@strip@tex@ext\@curr@file}%
130 \expandafter\@docinclude\expandafter{\@curr@file}
131 \fi}

132 \def\@docinclude#1 {\clearpage

133 \if@filesw \immediate\write\@mainaux{\string\@input{#1.aux}}\fi
134 \@tempswtrue\if@partsw \@tempswafalse\edef\@tempb{#1}\@for
135 \@tempa:=\@partlist\do{\ifx\@tempa\@tempb\@tempswtrue\fi}\fi
136 \if@tempsw \let\@auxout\@partaux \if@filesw
137 \immediate\openout\@partaux "#1.aux"
138 \immediate\write\@partaux{\relax}\fi

```

```
139 \@filehook@set@CurrentFile
```

我们需要保存 (并稍后恢复) 与索引相关的各种命令, 这些命令可能会被包含的文件更改。

```
140 \let\@ltxdoc@PrintIndex\PrintIndex
141 \let\PrintIndex\relax
142 \let\@ltxdoc@PrintChanges\PrintChanges
143 \let\PrintChanges\relax
144 \let\@ltxdoc@theglossary\theglossary
145 \let\@ltxdoc@endtheglossary@endtheglossary
146 \part{\currentfile}%
147 {\let\ttfamily\relax
148 \xdef\filekey{\filekey, \thepart={\ttfamily\currentfile}}}%
149 \DocInput{\currentfile}%
150 \let\PrintIndex\@ltxdoc@PrintIndex
151 \let\PrintChanges\@ltxdoc@PrintChanges
152 \let\theglossary\@ltxdoc@theglossary
153 \let\endtheglossary\@ltxdoc@endtheglossary
154 \clearpage
155 \@writeckpt{#1}\if@filesw \immediate\closeout\@partaux \fi
156 \else\@nameuse{cp@#1}\fi\let\@auxout\@mainaux}
157 \gdef\codeline@wrindex#1{\if@filesw
```

在索引项 (index entries) 中将 \protect 设置为合适的值 (我们不能使用 \set@display@protect, 因为这会导致命令后出现不同的空格数, 具体取决于写入索引之前发生的展开的数量)。

```
158 \begingroup
159 \let\protect\noexpand
160 \immediate\write\@indexfile
161 {\string\indexentry{#1}%
162 {\filesep\number\c@CodelineNo}}%
163 \endgroup\fi}
164 \let\filesep\@empty
```

\aalph 特殊形式的 \alph 作为当前 source2e.tex 包含 26 个以上的文件。.

```
165 \def\aalph#1{\@aalph{\csname c@#1\endcsname}}
166 \def\@aalph#1{%
167 \ifcase#1\or a\or b\or c\or d\or e\or f\or g\or h\or i\or
168 j\or k\or l\or m\or n\or o\or p\or q\or r\or s\or
169 t\or u\or v\or w\or x\or y\or z\or A\or B\or C\or
```

```

170      D\or E\or F\or G\or H\or I\or J\or K\or L\or M\or
171      N\or O\or P\or Q\or R\or S\or T\or U\or V\or W\or
172      X\or Y\or Z\else\@ctrerr\fi}

```

\docincludeaux

```

173 \def\docincludeaux{%
174   \def\thepart{\aalph{part}}\def\filesep{\thepart-}%
175   \let\filekey\@gobble
176   \g@addto@macro\index@prologue{%
177     \gdef\@oddfoot{\parbox[t]{\textwidth}{\strut\footnotesize
178       \raggedright{\bfseries File Key:} \filekey}}%
179     \let\@evenfoot\@oddfoot}%
180   \global\let\docincludeaux\relax
181   \gdef\@oddfoot{%
182     \expandafter\ifx\csname ver@\currentfile\endcsname\relax
183       File \thepart: {\ttfamily\currentfile} %
184     \else
185       \GetFileInfo{\currentfile}%
186       File \thepart: {\ttfamily\filename} %
187       Date: \filedate\ %
188       Version \fileversion
189       \fi
190       \hfill\thepage}%
191   \let\@evenfoot\@oddfoot}%

```

\MaintainedByLaTeXTeam 生成对错误数据库 (bug database) 的样板引用 (boilerplate reference)。

```

192 \def\MaintainedBy#1{\gdef\@maintainedby{#1}}
193 \let\@maintainedby\@empty
194 \def\MaintainedByLaTeXTeam#1{%
195   {\gdef\@maintainedby{%
196     此文件由 \LaTeX{} 项目团队维护。\\%
197     可以在以下位置打开 Bug 报告 (类别 \texttt{#1})\\%
198     \url{https://latex-project.org/bugs.html}.}}}%
199 \def\@maketitle{%
200   \newpage
201   \null
202   \vskip 2em%
203   \begin{center}%
204     \let \footnote \thanks
205     {\LARGE \@title \par}%

```

```

206 \vskip 1.5em%
207 {\large
208 \lineskip .5em%
209 \begin{tabular}[t]{c}%
210 \@author
211 \end{tabular}\par}%
212 \vskip 1em%
213 {\large \@date}%
214 \ifx\@maintainedby\@empty
215 \else
216 \vskip 1em%
217 \fbox{\fbox{\begin{tabular}{@{}l@{}}\@maintainedby\end{tabular}}}%
218 \fi
219 \end{center}%
220 \par
221 \vskip 1.5em}

```

```

222 \def\task#1#2{}

```

l3doc.cls 的一些功能

```

223 \DeclareRobustCommand\cs[1]{\texttt{\backslash\detokenize{#1}}}%
224 \AtBeginDocument{%
225 \renewcommand\PrintMacroName[1]{\MacroFont\detokenize{#1}}%

```

如果有人加载了 `csquotes` 或在前言中做了一些定义, 我们会提供这些延迟 (delayed)。

```

226 \providecommand\LuaTeX{Lua\TeX}
227 \providecommand\cls{\textsf}
228 \providecommand\pkg{\textsf}
229 \providecommand\enquote[1]{``#1''}
230 \providecommand\url{\texttt}
231 }
232 \</class>

```

## 10 配置文件

```

233 <*cfg>
234 %
235 %
236 % 这是我们用来格式化 LaTeX 内核源代码的 ltxdoc 配置文件。
237 %

```

```

238 %
239 % 版权所有 2006、2007、2011 Heiko Oberdiek (海科·奥伯代克)
240 % 版权所有 2014–2021 LaTeX\ 项目
241 %
242 \ProvidesFile{ltxdoc.cfg}%
243 [2022/06/14 v2.0d ltxdoc.cls 配置 (LaTeX 项目)]
244 \PassOptionsToClass{a4paper}{article}
245 % hyperref 和 hypdoc 现在加载较晚 (或由用户加载), 因此我们必须等待一些调整,
246 % 直到完成为止
247 \AddToHook{package/hyperref/after}{%
248 %% \RequirePackage{hypdoc}% % 这现在由 doc 触发
249 \RequirePackage{pdftexcmds}\relax
250 \ifnum\pdf@strcmp{\jobname}{inputenc}=0 %
251 \hypersetup{pdfencoding=auto}%
252 \pdfstringdefDisableCommands{%
253 \def\meta#1{% inputenc.dtx
254 \9060\010#1\9060\011%
255 }%
256 }%
257 \else
258 \fi
259 \pdfstringdefDisableCommands{%
260 \let\env\relax % longtable.dtx
261 \let\mytt\relax % tabularx.dtx
262 }%
263 }
264 % 对于 LaTeX 内核之外的包文档来说, 这应该可以很好地工作, 但是如果不行, 您可
265 % 以使用 “nocfg” 选项来阻止加载, 即,
266 %
267 % \documentclass[nocfg]{ltxdoc}
268 %
269 % 或者提供自己的配置文件
270 \endinput
271 </cfg>

```