

# Extensions to the `ltxdoc` class <sup>\*†</sup>

Arthur Ogawa <sup>‡</sup>

October 6, 2020

This file embodies the `ltxdocext` package, the implementation and its user documentation.

The distribution point for this work is [publish.aps.org/revtex](http://publish.aps.org/revtex), which contains prebuilt runtime files, documentation, and full source, ready to add to a TDS-compliant  $\TeX$  installation.

The `ltxdocext` package was commissioned by the American Physical Society and is distributed under the terms of the  $\LaTeX$  Project Public License, the same license under which all the portions of  $\LaTeX$  itself are distributed. Please see <http://ctan.tug.org/macros/latex/base/lppl.txt> for details.

To use this document class, you must have a working  $\TeX$  installation equipped with  $\LaTeX 2_{\epsilon}$  and possibly `pdftex` and Adobe Acrobat Reader or equivalent.

To install, retrieve the distribution, unpack it into a directory on the target computer, and move the files `ltxdocext.sty` and `acrofont.sty` into a location in your filesystem where they will be found by  $\LaTeX$ .

If you will be using the `acrofont` package, you must also install the virtual fonts `zpsynocmr`, `zptmnochr`, `zptmnochr`, and `zpzcnocmry`. The corresponding `.tfm`, `.vf`, and `.vpl` files are part of this distribution.

To use, read the user documentation `ltxdocext.pdf`. The `.dtx` file, `ltxdocext.dtx`, constitutes in itself an instance of use of the `ltxdocext` package and the `acrofont` package.

## Contents

<b>1</b>	<b>Processing Instructions</b>	<b>2</b>
1.1	Build Instructions . . . . .	3
1.2	Bill of Materials . . . . .	3
1.2.1	Primary Source . . . . .	3
1.2.2	Generated by <code>latex ltxdocext.dtx</code> . . . . .	4
1.2.3	Generated by <code>tex ltxdocext.ins</code> . . . . .	4
1.2.4	Documentation . . . . .	4
1.2.5	Auxiliary . . . . .	4

---

<sup>\*</sup>This file has version number 1.1c, last revised 2020/09/30.

<sup>†</sup>Version 1.1c © 2008 The American Physical Society

<sup>‡</sup>[mailto:arthur\\_ogawa@sbcglobal.net](mailto:arthur_ogawa@sbcglobal.net)

<b>2</b>	<b>Code common to all modules</b>	<b>4</b>
<b>3</b>	<b>The driver module doc</b>	<b>5</b>
3.1	The Preamble . . . . .	5
3.1.1	Docstrip and info directives . . . . .	5
3.2	The “Read Me” File . . . . .	5
3.3	The Document Body . . . . .	8
<b>4</b>	<b>Using the ltxdoc and acrofont packages</b>	<b>8</b>
4.1	Extensions to the ltxdoc class . . . . .	8
4.1.1	Extensions to the verbatim environment and \verb command	8
4.1.2	The -matter Commands Work . . . . .	9
4.1.3	The \GetFileInfo command . . . . .	9
4.1.4	Self-Indexing Commands . . . . .	9
4.1.5	Unnumbered Tables . . . . .	9
4.1.6	Structuring Tables . . . . .	10
4.1.7	A Sectioning Command Below \subsection . . . . .	10
4.2	Alterations to the ltxdoc class . . . . .	10
<b>5</b>	<b>Extensions to the ltxdoc class</b>	<b>10</b>
5.1	Beginning of the package DOCSTRIP module . . . . .	10
5.2	Beginning of the kernel DOCSTRIP module . . . . .	11
5.3	Incorporate ltxguide.cls extensions . . . . .	11
5.4	Changes to the base class of the ltxdoc class . . . . .	14
5.5	Extensions to the base class of ltxdoc.cls . . . . .	14
5.6	In lieu of ltxdoc.cfg . . . . .	16
5.7	Extension to L <sup>A</sup> T <sub>E</sub> X’s filecontents Environment . . . . .	17
5.8	End of the kernel DOCSTRIP module . . . . .	17
5.9	Tail of the package DOCSTRIP module . . . . .	17
<b>6</b>	<b>Font Package for Acrobat Compatability</b>	<b>17</b>
6.1	Beginning of the fonts DOCSTRIP module . . . . .	18
6.2	Variants on psfonts packages . . . . .	18
6.3	Font definition files . . . . .	19
6.4	More math substitutions . . . . .	19
6.5	End of the fonts DOCSTRIP module . . . . .	20
<b>7</b>	<b>Programming Conventions</b>	<b>20</b>
7.1	Whitespace Conventions . . . . .	20
7.2	Conventions For Procedures . . . . .	21
7.3	Conventions For L <sup>A</sup> T <sub>E</sub> X . . . . .	22
	<b>Index</b>	<b>23</b>

# 1 Processing Instructions

The package files `ltxdocext.sty` and `acrofont.sty` are generated from this file, `ltxdocext.dtx`, using the DOCSTRIP facility of L<sup>A</sup>T<sub>E</sub>X via `tex ltxdocext.dtx` (Note: do *not* use L<sup>A</sup>T<sub>E</sub>X for this step). The typeset documentation that you are now reading is generated from the same file by typesetting it with L<sup>A</sup>T<sub>E</sub>X or `pdftex` via `latex ltxdocext.dtx` or `pdflatex ltxdocext.dtx`.

## 1.1 Build Instructions

You may bootstrap this suite of files solely from `ltxdocext.dtx`. Prepare by installing L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> (and either `tex` or `pdftex`) on your computer, then carry out the following steps:

1. Within an otherwise empty directory, typeset `ltxdocext.dtx` with T<sub>E</sub>X or `pdftex`; thereby generating the package file `ltxdocext.sty`, and the package file `acrofont.sty`. Make sure that DOCSTRIP receives permission to overwrite existing versions of these packages.

2. Now typeset `ltxdocext.dtx` with L<sup>A</sup>T<sub>E</sub>X or `pdflatex`; you will obtain the typeset documentation you are now reading, along with the file `README`.

Note: you will have to run L<sup>A</sup>T<sub>E</sub>X, then `makeindex -s gind.ist ltxdocext.idx`, then `makeindex -s gglo.ist -o ltxdocext.gls ltxdocext.glo`, then L<sup>A</sup>T<sub>E</sub>X again in order to obtain a valid index and table of contents.

3. Install the following files into indicated locations within your TDS-compliant `texmf` tree (you may need root access):

- `$TEXMF/tex/latex/revtex/ltxdocext.sty` and `$TEXMF/tex/latex/revtex/acrofont.sty`
- `$TEXMF/source/latex/revtex/ltxdocext.dtx`
- `$TEXMF/doc/latex/revtex/ltxdocext.pdf`

where `$TEXMF/` stands for `texmf-local/`, or some other `texmf` tree in your installation.

4. Run `mktexlsr` on `$TEXMF/` (you may need root access).
5. Build and installation are now complete; now put a `\usepackage{ltxutil}` in your document preamble!

## 1.2 Bill of Materials

Following is a list of the files in this distribution arranged according to provenance.

### 1.2.1 Primary Source

One single file generates all.

```
%ltxdocext.dtx
%
```

### 1.2.2 Generated by latex ltxdocext.dtx

Typesetting the source file under pdflatex generates the readme and the documentation.

```
%README ltxdocext.pdf
%
```

### 1.2.3 Generated by tex ltxdocext.ins

Typesetting this file with TeX generates the package file.

```
%ltxdocext.sty acrofont.sty
%
```

### 1.2.4 Documentation

The following are the online documentation:

```
%ltxdocext.pdf
%
```

### 1.2.5 Auxiliary

The following are auxiliary files generated in the course of running L<sup>A</sup>T<sub>E</sub>X:

```
%ltxdocext.aux ltxdocext.idx ltxdocext.ind ltxdocext.log ltxdocext.toc
%
```

## 2 Code common to all modules

We want to require only one place in this file where the version number is stated, and we also want to ensure that the version number is embedded into every generated file.

Now we declare that these files can only be used with L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. An appropriate message is displayed if a different TeX format is used.

```
1 %<*driver|package|fonts>
2 \NeedsTeXFormat{LaTeX2e}[1995/12/01]%
3 %</driver|package|fonts>
```

As desired, the following modules all take common version information:

```
4 %<package>\ProvidesFile{ltxdocext.sty}%
5 %<font>\ProvidesFile{acrofont.sty}%
6 %<*driver>
7 \expandafter\ProvidesFile\expandafter{\jobname.dtx}%
8 %</driver>
```

The following line contains, for once and for all, the version and date information. By various means, this information is reproduced consistently in all generated files and in the typeset documentation.

```
9 %<version>
10 [2020/09/30 1.1c ltxdoc extensions package]% \fileversion
```

### 3 The driver module doc

This module, consisting of the present section, typesets the programmer’s documentation, generating the README as required.

Because the only uncommented-out lines of code at the beginning of this file constitute the doc module itself, we can simply typeset the .dtx file directly, and there is thus rarely any need to generate the “doc” DOCSTRIP module. Module delimiters are nonetheless required so that this code does not find its way into the other modules.

The `\end{document}` command concludes the typesetting run.

```
11 %<*driver>
```

#### 3.1 The Preamble

The programmers documentation is formatted with the `ltxdoc` document class, with local customizations, and with the usual code line indexing.

```
12 \documentclass[draft]{ltxdoc}
13 \RequirePackage{ltxdocext}%
14 \RequirePackage[colorlinks=true,linkcolor=blue]{hyperref}%
15 \makeatletter
16 \@ifundefined{package@font}{}%
17   {\expandafter\RequirePackage\expandafter{\csname package@font\endcsname}}%
18 \makeatother
19 \CodelineIndex\EnableCrossrefs % makeindex -s gind.ist ltxdocext
20 \RecordChanges % makeindex -s gglo.ist -o ltxdocext.gls ltxdocext.glo
```

##### 3.1.1 Docstrip and info directives

We use so many DOCSTRIP modules that we set the `StandardModuleDepth` counter to 1.

```
21 \setcounter{StandardModuleDepth}{1}
```

The following command retrieves the date and version information from this file.

```
22 \expandafter\GetFileInfo\expandafter{\jobname.dtx}%
```

#### 3.2 The “Read Me” File

As promised above, here is the contents of the “Read Me” file. That file serves a double purpose, since it also constitutes the beginning of the programmer’s documentation. What better thing, after all, to have appear at the beginning of the typeset documentation?

A good discussion of how to write a ReadMe file can be found in Engst, Tonya, “Writing a ReadMe File? Read This” *MacTech* October 1998, p. 58.

Note the appearance of the `\StopEventually` command, which marks the dividing line between the user documentation and the programmer documentation.

The usual user will not be asked to do a full build, not to speak of the bootstrap. Instructions for carrying these processes begin the programmer’s manual.

```

23 \begin{filecontents*}[overwrite]{README}
24 \title{%
25 Extensions to the \classname{ltxdoc} class%
26 \thanks{%
27 This file has version number \fileversion,
28 last revised \filedate.%
29 }%
30 \thanks{%
31 Version \fileversion\ \copyright\ 2008 The American Physical Society
32 }%
33 }%
34 \author{%
35 Arthur Ogawa%
36 \thanks{\texttt{mailto:arthur\_ogawa at sbcglobal.net}}%
37 }%
38 %\iffalse
39 % For version number and date,
40 % search on "\fileversion" in the .dtx file,
41 % or see the end of the README file.
42 %\fi
43
44 \maketitle
45
46 This file embodies the \classname{ltxdocext} package,
47 the implementation and its user documentation.
48
49 The distribution point for this work is
50 \url{publish.aps.org/revtex},
51 which contains prebuilt runtime files, documentation, and full source,
52 ready to add to a TDS-compliant \TeX\ installation.
53
54 The \classname{ltxdocext} package was commissioned by the American Physical Society
55 and is distributed under the terms of the \LaTeX\ Project Public License,
56 the same license under which all the portions of \LaTeX\ itself are distributed.
57 Please see \url{http://ctan.tug.org/macros/latex/base/lppl.txt} for details.
58
59 To use this document class, you must have a working
60 \TeX\ installation equipped with \LaTeXe\
61 and possibly pdftex and Adobe Acrobat Reader or equivalent.
62
63 To install, retrieve the distribution,
64 unpack it into a directory on the target computer,
65 and move the files \file{ltxdocext.sty} and \file{acrofont.sty}
66 into a location in your filesystem where they will be found by \LaTeX.
67
68 If you will be using the \classname{acrofont} package, you must
69 also install the virtual fonts
70 \file{zpsynocmr}, \file{zptmnochr},
71 \file{zptmnochr}, and \file{zpcnocmry}.
72 The corresponding \file{.tfm}, \file{.vf}, and \file{.vpl}

```

73 files are part of this distribution.  
74  
75 To use, read the user documentation `\file{ltxdocext.pdf}`.  
76 The `\file{.dtx}` file, `\file{ltxdocext.dtx}`, constitutes  
77 in itself an instance of use of the `\classname{ltxdocext}`  
78 package and the `\classname{acrofont}` package.  
79  
80 `\tableofcontents`  
81  
82 `\section{Processing Instructions}`  
83  
84 The package files `\file{ltxdocext.sty}` and `\file{acrofont.sty}`  
85 are generated from this file, `\file{ltxdocext.dtx}`,  
86 using the `{\sc docstrip}` facility of `\LaTeX`  
87 via `|tex ltxdocext.dtx|` (Note: do `\emph{not}` use `\LaTeX` for this step).  
88 The typeset documentation that you are now reading is generated from  
89 the same file by typesetting it with `\LaTeX` or `pdftex`  
90 via `|latex ltxdocext.dtx|` or `|pdflatex ltxdocext.dtx|`.  
91  
92 `\subsection{Build Instructions}`  
93  
94 You may bootstrap this suite of files solely from `\file{ltxdocext.dtx}`.  
95 Prepare by installing `\LaTeXe` (and either `tex` or `pdftex`) on your computer,■  
96 then carry out the following steps:  
97 `\begin{enumerate}`  
98 `\item`  
99 Within an otherwise empty directory,  
100 `typeset \file{ltxdocext.dtx}` with `\TeX` or `pdftex`;  
101 thereby generating the package file `\file{ltxdocext.sty}`,  
102 and the package file `\file{acrofont.sty}`.  
103 Make sure that `{\sc docstrip}` receives permission  
104 to overwrite existing versions of these packages.  
105 `\item`  
106 Now  
107 `typeset \file{ltxdocext.dtx}` with `\LaTeX` or `pdflatex`;  
108 you will obtain the typeset documentation you are now reading,  
109 along with  
110 the file `\file{README}`.  
111  
112 Note: you will have to run `\LaTeX`, then  
113 `\file{makeindex} \texttt{-s gind.ist ltxdocext.idx}`, then  
114 `\file{makeindex} \texttt{-s gglo.ist -o ltxdocext.gls ltxdocext.glo}`, then■  
115 `\LaTeX` again in order to obtain a valid index and table of contents.  
116 `\item`  
117 Install the following files into indicated locations within your  
118 TDS-compliant `\texttt{texmf}` tree (you may need root access):  
119 `\begin{itemize}`  
120 `\item`  
121 `\file{${TEXMF}/}\file{tex/}\file{latex/}\file{revtex/}\classname{ltxdocext.sty}` and■  
122 `\file{${TEXMF}/}\file{tex/}\file{latex/}\file{revtex/}\classname{acrofont.sty}`■

```

123 \item
124 \file{$TEXMF/}\file{source/}\file{latex/}\file{revtex/}\classname{ltxdocext.dtx}■
125 \item
126 \file{$TEXMF/}\file{doc/}\file{latex/}\file{revtex/}\classname{ltxdocext.pdf}■
127 \end{itemize}
128 where \file{$TEXMF/} stands for \file{texmf-local/}, or some other \texttt{texmf} tree■
129 in your installation.
130 \item
131 Run \texttt{mktexlsr} on \file{$TEXMF/} (you may need root access).
132 \item
133 Build and installation are now complete;
134 now put a \cmd\usepackage\texttt{\{ltxutil\}} in your document preamble!
135 \end{enumerate}
136 \end{filecontents*}

```

### 3.3 The Document Body

Here is the document body, containing only a `\DocInput` directive—referring to this very file. This very cute self-reference is a common `ltxdoc` idiom.

```

137 \begin{document}%
138 \expandafter\DocInput\expandafter{\jobname.dtx}%
139 \PrintChanges
140 \end{document}
141 %</driver>

```

## 4 Using the `ltxdoc` and `acrofont` packages

These packages are an adjunct to the standard  $\text{\LaTeX}$  `ltxdoc` class and may be simply invoked as follows:

```

%\documentclass[draft]{ltxdoc}
%\RequirePackage{ltxdocext}%
%\RequirePackage{acrofont}%
%\CodelineIndex\EnableCrossrefs
%

```

Your document should simply cleave to the standards of the `ltxdoc` class, with extensions and alterations as noted.

### 4.1 Extensions to the `ltxdoc` class

#### 4.1.1 Extensions to the `verbatim` environment and `\verb` command

The delimiters `<<` and `>>` within the scope of the `verbatim` environment or within the argument of the `\verb` command produce italics surrounded by angle brackets. This typographic convention usually indicates *metalanguage*, i.e., a placeholder.

To obtain the angle bracket character per se, double the character, viz., “the delimiter `\verb+<<<<+`”.



### 4.1.2 The `-matter` Commands Work

The sectioning commands `\frontmatter`, `\mainmatter`, and `\backmatter` of the standard L<sup>A</sup>T<sub>E</sub>X book class are operative in the `ltxdoc` class.

### 4.1.3 The `\GetFileInfo` command

You can use the `\GetFileInfo` command to extract the date, version, and file info of a file which has registered itself via the `\ProvidesFile` or `\ProvidesClass` command (employing the optional argument thereto).

For instance, if your document contains the following:

```
%\RequirePackage{ltxdocext}%
%\GetFileInfo{ltxdocext.sty}%
%
```

then the following control sequence names will have a value corresponding to that package's `\ProvidesFile` command: `\filedate`: the file's date, `\fileversion`: the file's version, and `\fileinfo`: the file's info.

### 4.1.4 Self-Indexing Commands

Certain commands automatically produce an index entry (or several related entries) according to the meaning.

meta-text	<code>\marg{<i>text</i>}</code>
command	<code>\cmd{<i>csname</i>}</code>
environment name	<code>\env{<i>name</i>}</code>
<code>\begin{foo}</code>	<code>\envb{<i>foo</i>}</code>
<code>\end{foo}</code>	<code>\enve{<i>foo</i>}</code>
argument	<code>\arg{<i>name</i>}</code>
optional	<code>\oarg{<i>name</i>}</code>
filename	<code>\file{<i>name</i>}</code>
url	<code>\url{<i>name</i>}</code>
document class	<code>\classname{<i>name</i>}</code>
document substyle	<code>\substyle{<i>name</i>}</code>
class option	<code>\classoption{<i>name</i>}</code>

### 4.1.5 Unnumbered Tables

When your documentation requires the use of an unnumbered table, use the `unnumtable` environment:

```
%\begin{unnumtable}
%\begin{tabular}{ll}
%table rows
%\end{tabular}
%\end{unnumtable}
%
```

#### 4.1.6 Structuring Tables

The commands `\toprule`, `\colrule`, and `\botrule` allow you to mark the beginning of the column heads the beginning of the table body, and the end of the table body, respectively. In context,

```
%\begin{tabular}{ll}
%\toprule
%<table head rows>
%\colrule
%<table rows>
%\botrule
%\end{tabular}
%
```

#### 4.1.7 A Sectioning Command Below `\subsection`

The `\subsubsection` command is defined.

### 4.2 Alterations to the `ltxdoc` class

The following involve no new markup, but they do change the appearance of your formatted documentation:

1. Using the `acronym` package causes your document to be formatted using the standard Acrobat fonts to the greatest extent possible. This means that for most documents, Computer Modern is not used at all. Math that unavoidable must use CM still exists, however.
2. An index will be produced at the end of the document without your needing to explicitly mark it up, and it will have an entry in the TOC.
3. The `quote` environment has a slightly smaller left margin.
4. Array columns are set tight by default.
5. A host of `\DoNotIndex` directives are invoked. I intend this list to grow to encompass even more commands. Send me your suggestions.

## 5 Extensions to the `ltxdoc` class

The package `DOCSTRIP` module comprises the package `ltxdocext.sty`, which provides extensions to the standard L<sup>A</sup>T<sub>E</sub>X `ltxdoc` class.

### 5.1 Beginning of the package `DOCSTRIP` module

This portion of code is only present in the L<sup>A</sup>T<sub>E</sub>X package (`.sty` file), not in the kernel portion.

```
142 %<*package>
```

```

143 \def\class@name{ltxdocext}%
144 \expandafter\PackageInfo\expandafter{\class@name}{%
145 An extension to the \protect\LaTeXe\space ltxdoc class
146 by A. Ogawa (arthur\_ogawa sbcglobal.net)%
147 }%
148 %</package>

```

## 5.2 Beginning of the kernel DOCSTRIP module

The bulk of the code is the kernel portion; a brief tail of package code then follows.

```
149 %<*kernel>
```

## 5.3 Incorporate ltxguide.cls extensions

Code extracted from `ltxguide.cls`, by Alan Jeffrey. “This code stolen from `ltxguide.cls`: Some hacks with `verbatim`... NB: this would be better done with the `verbatim` package, but this document has to run on any L<sup>A</sup>T<sub>E</sub>X installation.”

```

150 \RequirePackage{verbatim}%
151 \let\o@verbatim\verbatim
152 \def\verbatim{%
153   \ifhmode\unskip\par\fi
154   % \nopagebreak           % Overridden by list penalty
155   \ifx@currsize\normalsize
156     \small
157   \fi
158   \o@verbatim
159 }%

```

Here we extend the font-setting command to include making `<>` active (i.e., adjusting the input encoding).

```

160 \renewcommand \verbatim@font {%
161   \normalfont \ttfamily
162   \catcode'\<=\active
163   \catcode'\>=\active
164 }%

```

Make `|...|` a synonym for `\verb|...|`.

```

165 \RequirePackage{shortvrb}
166 \AtBeginDocument{%
167   \MakeShortVerb{\|}%
168 }%

```

Make active bracket characters produce italics surrounded by angle brackets (used in `verbatim` and `\verb`). `<<` produces a less-than, and `>>` produces a greater-than.

```

169 \begingroup
170   \catcode'\<=\active
171   \catcode'\>=\active
172   \gdef<{\@ifnextchar<\@lt\@meta}

```

```

173 \gdef>{\@ifnextchar>\@gt\@gtr@err}
174 \gdef\@meta#1>{\marg{#1}}
175 \gdef\@lt<{\char'\<}
176 \gdef\@gt>{\char'\>}
177 \endgroup
178 \def\@gtr@err{%
179   \ClassError{ltxguide}{%
180     Isolated \protect>%
181   }{%
182     In this document class, \protect<...\protect>
183     is used to indicate a parameter.\MessageBreak
184     I've just found a \protect> on its own.
185     Perhaps you meant to type \protect>\protect>?
186   }%
187 }
188 \def\verbatim@nolig@list{\do\'\do\,\do\'\do\'}

```

End of code stolen from ltxguide.cls. Thanks, Alan.

Add functionality from doc.dtx: (code stolen from doc.dtx):

```

189 \def\GetFileInfo#1{%
190   \def\filename{#1}%
191   \def\@tempb##1 ##2 ##3\relax##4\relax{%
192     \def\filedate{##1}%
193     \def\fileversion{##2}%
194     \def\fileinfo{##3}}%
195   \edef\@tempa{\csname ver@#1\endcsname}%
196   \expandafter\@tempb\@tempa\relax? ? \relax\relax}

```

(end of code stolen from doc.dtx. Thanks FMi.)

Various forms of self-indexing commands:

```

197 \DeclareRobustCommand{\marg}[1]{%
198   \meta{#1}%
199   \index{#1=\string\meta{#1} placeholder}\index{placeholder>#1=\string\meta{#1}}%
200 }%
201 \DeclareRobustCommand\meta[1]{%
202   \mbox{\LARGE\itshape#1/\RANGLE}%
203 }%
204 \def\LARGE{\langle$}%
205 \def\RANGLE{\rangle$}%
206 \DeclareRobustCommand{\arg}[1]{%
207   {\ttfamily\string{}}\meta{#1}{\ttfamily\string{}}%
208   \index{#1=\string\ttt{#1}, argument}\index{argument>#1=\string\ttt{#1}}%
209 }%
210 \let\oarg\undefined
211 \DeclareRobustCommand{\oarg}[1]{%
212   {\ttfamily[%]
213   }\meta{#1}{\ttfamily%[
214   ]}%
215   \index{#1=\string\ttt{#1}, optional argument}%
216   \index{argument, optional>#1=\string\ttt{#1}}%

```

```

217 }%
218 \DeclareRobustCommand\cmd{\begingroup\makeatletter\@cmd}%
219 \long\def\@cmd#1{%
220 \endgroup
221 \cs{\expandafter\cmd@to@cs\string#1}%
222 \expandafter\cmd@to@index\string#1\@nil
223 }%
224 \def\cmd@to@cs#1#2{\char\number'#2\relax}%
225 \def\cmd@to@index#1#2\@nil{%
226 \index{#2=\string\cmd#1#2}\index{command>#2=\string\cmd#1#2}%
227 }%
228 \DeclareRobustCommand\cs[1]{\tffamily\char'\#1}%
229 \def\scmd#1{%
230 \cs{\expandafter\cmd@to@cs\string#1}%
231 \expandafter\scmd@to@index\string#1\@nil
232 }%
233 \def\scmd@to@index#1#2\@nil#3{%
234 \index{\string$#3=\string\cmd#1#2---#3}%
235 %\index{command>\string$#3=\string\cmd#1#2---#3}%
236 }%
237 \DeclareRobustCommand\env{name@idx{environment}}%
238 \DeclareRobustCommand\envb[1]{%
239 {\tffamily\string\begin\string{ }\env{#1}{\tffamily\string}}%
240 }%
241 \DeclareRobustCommand\enve[1]{\tffamily\string\end\string{ }\env{#1}{\tffamily\string}}%
242 \DeclareRobustCommand{file}{\begingroup@sanitize@file}%
243 \long\def\@file#1{\endgroup
244 {\tffamily#1}%
245 \index{#1=\string\ttt{#1}}\index{file>#1=\string\ttt{#1}}%
246 }%
247 \DeclareRobustCommand\substyle{name@idx{document substyle}}%
248 \DeclareRobustCommand\classoption{name@idx{document class option}}%
249 \DeclareRobustCommand\classname{name@idx{document class}}%
250 \def\name@idx#1#2{%
251 {\tffamily#2}%
252 \index{#2\space#1=\string\ttt{#2}\space#1}\index{#1>#2=\string\ttt{#2}}%
253 }%
254 \DeclareRobustCommand\url@ltxdocext{\begingroup\catcode'\active\catcode'\active\catcode'\active\
255 \AtBeginDocument{%
256 \ifx\url\undefined\let\url\url@ltxdocext\fi
257 }%
258 \def\@url#1{%
259 \url@break{\tffamily#1}%
260 \url@char\edef\@tempa#1=\string\url{#1}%
261 \expandafter\index\expandafter{\@tempa}%
262 \expandafter\index\expandafter{\expandafter u\expandafter r\expandafter l\expandafter >\@tempa}
263 \endgroup
264 }%
265 {\catcode'\active\aftergroup\def\aftergroup:}{\active@colon}%
266 \def\colon@break{\colon@char\allowbreak}%

```

```

267 \def\colon@char{:}%
268 {\catcode'\active\aftergroup\def\aftergroup/}{\active@slash}%
269 \def\slash@break{/\slash@char\allowbreak}%
270 \def\slash@char{/}%
271 {\catcode'\active\aftergroup\def\aftergroup.}{\active@dot}%
272 \def\dot@break{\dot@char\allowbreak}%
273 \def\dot@char{.}%
274 \def\url@break{\let\active@slash\slash@break\let\active@dot\dot@break\let\active@colon\colon@br
275 \def\url@char{\let\active@slash\slash@char\let\active@dot\dot@char\let\active@colon\colon@char}

```

## 5.4 Changes to the base class of the ltxdoc class

Modify theindex environment so that it produces a TOC entry

```

276 \renewenvironment{theindex}
277     {\if@twocolumn
278         \@restonecolfalse
279         \else
280             \@restonecoltrue
281             \fi
282             \columnseprule \z@
283             \columnsep 35\p@
284 \def\see##1##2{\textit{See} ##1}%
285 \def\seealso##1##2{\textit{See also} ##1}%
286 \long\def\cmd##1{\cs{\expandafter\cmd@to@cs\string##1}}%
287 \def\@url##1{\url@break\ttt{##1}\endgroup}%
288 \def\ttt{\begingroup\@sanitize\ttfamily\@ttt}%
289 \def\@ttt##1{##1}\endgroup}%
290 \mathchardef\save@secnumdepth\c@secnumdepth
291 \c@secnumdepth\m@ne
292     \twocolumn[\section{\indexname}]%
293 %     \mkboth{\MakeUppercase\indexname}%
294 %         {\MakeUppercase\indexname}%
295 \c@secnumdepth\save@secnumdepth
296     \thispagestyle{plain}\parindent\z@
297     \parskip\z@ \@plus .3\p@\relax
298     \let\item\@idxitem}
299     {\if@restonecol\onecolumn\else\clearpage\fi}
300 \renewenvironment{quote}
301     {\list{}{%
302         \leftmargin1em\relax
303         \rightmargin\leftmargin
304     }}%
305     \item\relax}
306     {\endlist}

```

## 5.5 Extensions to the base class of ltxdoc.cls

Matter commands from book.cls

```

307 \newif\if@mainmatter

```

```

308 \newif\if@openright
309 \@openrighttrue
310 \DeclareRobustCommand\frontmatter{%
311   \cleartorecto
312   \@mainmatterfalse
313   \pagenumbering{roman}%
314 }%
315 \DeclareRobustCommand\mainmatter{%
316   \cleartorecto
317   \@mainmattertrue
318   \pagenumbering{arabic}%
319 }%
320 \DeclareRobustCommand\backmatter{%
321   \if@openright
322     \cleartorecto
323   \else
324     \clearpage
325   \fi
326   \@mainmatterfalse
327 }%
328 \ifx\undefined\cleartorecto
329 \def\cleartorecto{\cleardoublepage}%
330 \fi

```

Unnumbered tables

`unnumtable` An unnumbered table does not float.

```

331 \def\@to{to}%
332 \newenvironment{unnumtable}{%
333 \par
334 \addpenalty\predisplaypenalty
335 \addvspace\abovedisplayskip
336 \hbox\@to\hsize\bgroup\hfil\ignorespaces
337 \let\@Hline\@empty
338 }{%
339 \unskip\hfil\egroup
340 \penalty\postdisplaypenalty
341 \vskip\belowdisplayskip
342 \aftergroup\ignorespaces
343 \@endpetrue
344 }%

```

Emulate `\toprule` and friends

```

345 \providecommand\toprule{\hline\hline}%
346 \providecommand\colrule{\hline\hline}%
347 \providecommand\botrule{\hline\hline}%

```

Define sectioning command below `\subsubsection`.

```

348 \DeclareRobustCommand\subsubsection{%
349 \startsection{subsubsection}{4}%
350 {\z@}{-15\p@\@plus-5\p@\@minus-2\p@}%

```

```

351 {5\p@}{\normalfont\normalsize\itshape}%
352 }%

```

## 5.6 In lieu of ltxdoc.cfg

We don't want everything to appear in the index

```

353 \DoNotIndex{\',\.,\@M,\@input,\@Alph,\@alph,\@addtoreset,\@arabic}
354 \DoNotIndex{\@badmath,\@centercr,\@cite}
355 \DoNotIndex{\@dotsep,\@empty,\@float,\@gobble,\@gobbletwo,\@ignoretrue}
356 \DoNotIndex{\@input,\@ixpt,\@m,\@minus,\@mkboth}
357 \DoNotIndex{\@ne,\@nil,\@nomath,\@plus,\@roman,\@set@topoint}
358 \DoNotIndex{\@tempboxa,\@tempcnta,\@tempdima,\@tempdimb}
359 \DoNotIndex{\@tempswafalse,\@tempswatruer,\@viipt,\@viipt,\@vipt}
360 \DoNotIndex{\@vpt,\@warning,\@xipt,\@xipt,\@xivpt,\@xpt,\@xvipt}
361 \DoNotIndex{\@xxpt,\@xxvpt,\@, \ ,\addpenalty,\addtolength,\addvspace}
362 \DoNotIndex{\advance,\ast,\begin,\begingroup,\bfseries,\bgroup,\box}
363 \DoNotIndex{\bullet}
364 \DoNotIndex{\cdot,\cite,\CodelineIndex,\cr,\day,\DeclareOption}
365 \DoNotIndex{\def,\DisableCrossrefs,\divide,\DocInput,\documentclass}
366 \DoNotIndex{\DoNotIndex,\egroup,\ifdim,\else,\fi,\em,\endtrivlist}
367 \DoNotIndex{\EnableCrossrefs,\end,\end@dblfloat,\end@float,\endgroup}
368 \DoNotIndex{\endlist,\everycr,\everypar,\ExecuteOptions,\expandafter}
369 \DoNotIndex{\fbox}
370 \DoNotIndex{\filedate,\filename,\fileversion,\fontsize,\framebox,\gdef}
371 \DoNotIndex{\global,\halign,\hangindent,\hbox,\hfil,\hfill,\hrule}
372 \DoNotIndex{\hsize,\hskip,\hspace,\hss,\if@tempswa,\ifcase,\or,\fi,\fi}
373 \DoNotIndex{\ifhmode,\ifvmode,\ifnum,\iftrue,\ifx,\fi,\fi,\fi,\fi,\fi}
374 \DoNotIndex{\input}
375 \DoNotIndex{\jobname,\kern,\leavevmode,\let,\leftmark}
376 \DoNotIndex{\list,\llap,\long,\m@ne,\m@th,\mark,\markboth,\markright}
377 \DoNotIndex{\month,\newcommand,\newcounter,\newenvironment}
378 \DoNotIndex{\NeedsTeXFormat,\newdimen}
379 \DoNotIndex{\newlength,\newpage,\nobreak,\noindent,\null,\number}
380 \DoNotIndex{\numberline,\OldMakeindex,\OnlyDescription,\p@}
381 \DoNotIndex{\pagestyle,\par,\paragraph,\paragraphmark,\parfillskip}
382 \DoNotIndex{\penalty,\PrintChanges,\PrintIndex,\ProcessOptions}
383 \DoNotIndex{\protect,\ProvidesClass,\raggedbottom,\raggedright}
384 \DoNotIndex{\refstepcounter,\relax,\renewcommand}
385 \DoNotIndex{\rightmargin,\rightmark,\rightskip,\rlap,\rmfamily}
386 \DoNotIndex{\secdef,\selectfont,\setbox,\setcounter,\setlength}
387 \DoNotIndex{\settowidth,\sfcode,\skip,\sloppy,\slshape,\space}
388 \DoNotIndex{\symbol,\the,\trivlist,\typeout,\tw@,\undefined,\uppercase}
389 \DoNotIndex{\usecounter,\usefont,\usepackage,\vfil,\vfill,\viipt}
390 \DoNotIndex{\viipt,\vipt,\vskip,\vspace}
391 \DoNotIndex{\wd,\xipt,\year,\z@}
392 \DoNotIndex{\next}

Direct ltxdoc to produce an index.
393 \AtEndDocument{\PrintIndex\PrintChanges}%

```



## 5.7 Extension to L<sup>A</sup>T<sub>E</sub>X's filecontents Environment

We want to coax the version number into `filecontents`-generated files. Note that we expect `\fileversion` and `\filedate` to hold the needed information. For this to be the case, your document should execute the `\GetFileInfo` command (as documented in section 4.1.3) before any instances of `filecontents`.

```
394 \makeatletter
395 \def\endfilecontents{%
396 \immediate\write\reserved@cf%
397 \string\iffalse\space ltxdoc klookch^^J%
398 \ifx\undefined\fileversion\else
399 \ifx\undefined\filedate\else
400 This file has version number \fileversion, last revised \filedate.%
401 \fi\fi
402 \string\fi
403 }%
404 \immediate\closeout\reserved@c
405 \def\T##1##2##3{%
406 \ifx##1\@undefined\else
407 \latex@warning@no@line{##2 has been converted to Blank ##3e}%
408 \fi
409 }%
410 \T\L{Form Feed}{Lin}%
411 \T\I{Tab}{Spac}%
412 \immediate\write\@unused{}%
413 }%
414 \expandafter\let\csname endfilecontents*\endcsname\endfilecontents
415 \makeatother

Alter formatting in arrays; set them tight.
416 \setlength\arraycolsep{0pt}%
```

## 5.8 End of the kernel DOCSTRIP module

```
417 %</kernel>
```

## 5.9 Tail of the package DOCSTRIP module

Here is the remainder of the package code.

```
418 %<*package>
```

Currently, there is little.

```
419 %</package>
```

## 6 Font Package for Acrobat Compatability

The package `acrofont` substitutes Acrobat-standard fonts for Computer Modern where possible, even in math mode. Documents typeset with this package in effect will require as little downloaded font data as possible, but will not be exemplars of fine math typesetting.

## 6.1 Beginning of the fonts DOCSTRIP module

The document class module comprises this and the next four sections.

`\class@base` We define in exactly one spot the base class. Typically that class will be one of `book`, `article`, or `report`. The base class effectively defines the use and the markup scheme of the class of documents to be handled by this class.

This class is a variant of the standard L<sup>A</sup>T<sub>E</sub>X book class: `ftp://ctan.tug.org/tex-archive/macros/latex/unpacked`.

```
420 %<*fonts>
421 \def\class@name{ltxdocext}%
422 \expandafter\ClassInfo\expandafter{\class@name}{%
423   Written for \protect\LaTeXe\space
424   by A. Ogawa (arthur_ogawa at sbcglobal.net)%
425 }
```

## 6.2 Variants on psfonts packages

The following uses `times.sty` from `/packages/psnfss/psfonts.dtx`

```
426 \RequirePackage{times}%
```

The following uses `mathptm.sty` from `/packages/psnfss/psfonts.dtx`

```
427 \RequirePackage{mathptm}%
```

The following is a customization of `ot1ptmcm.fd`. The virtual font referred to here `zptmnocmr` is a variant of Sebastian Rahtz's `zptmcmr`, but with even more glyphs moved from CM to Acrobat-standard fonts.

```
428 \DeclareFontFamily{OT1}{ptmcm}{}
429 \DeclareFontShape{OT1}{ptmcm}{m}{n}{
430   <-> zptmnocmr
431 }{}
432 \DeclareFontShape{OT1}{ptmcm}{l}{n}{<->ssub * ptmnocm/m/n}{}
```

The following is a customization of `omlptmcm.fd`. The virtual font referred to here `zptmnocmrm` is a variant of Sebastian Rahtz's `zptmcmrm`, but with even more glyphs moved from CM to Acrobat-standard fonts.

```
433 \DeclareFontFamily{OML}{ptmcm}{\skewchar \font =127}
434 \DeclareFontShape{OML}{ptmcm}{m}{it}{
435   <-> zptmnocmrm
436 }{}
437 \DeclareFontShape{OML}{ptmcm}{l}{it}{<->ssub * ptmcm/m/it}{}
438 \DeclareFontShape{OML}{ptmcm}{m}{sl}{<->ssub * ptmcm/m/it}{}
439 \DeclareFontShape{OML}{ptmcm}{l}{sl}{<->ssub * ptmcm/m/sl}{}
```

The following is a customization of `omspzccm.fd`. The virtual font referred to here `zpzcnocmry` is a variant of Sebastian Rahtz's `zpzccmry`, but with even more glyphs moved from CM to Acrobat-standard fonts.

```
440 \DeclareFontFamily{OMS}{pzccm}{}
441 \DeclareFontShape{OMS}{pzccm}{m}{n}{
442   <-> zpzcnocmry
```

```

443 }{}% cmsy10 Symbol Zapf Chancery Medium-Italic Times-Roman
444 \DeclareFontShape{OMS}{pzcmm}{l}{n}{<->ssub * pzcmm/m/n}{}

```

The following is a customization of `omxpsycm.fd`. The virtual font referred to here `zpsynocmr` is a variant of Sebastian Rahtz's `zpsycmrv`, but with even more glyphs moved from CM to Acrobat-standard fonts.

```

445 \DeclareFontFamily{OMX}{psycm}{}
446 \DeclareFontShape{OMX}{psycm}{m}{n}{
447   <-> zpsynocmr
448 }{}
449 \DeclareFontShape{OMX}{psycm}{l}{n}{<->ssub * psycm/m/n}{}
450 %
451 \DeclareFontEncoding{8r}{}{}% from file: 8renc.def
452 \DeclareFontFamily{8r}{cmr}{\hyphenchar\font45 }% from file: 8rcmr.fd
453 \DeclareFontShape{8r}{cmr}{m}{n}{
454   <-> cmr10
455 }{}

```

### 6.3 Font definition files

The following forces L<sup>A</sup>T<sub>E</sub>X to do now what it would do anyway: load the ‘font definition’ information for the fonts that we use. In this way, we prepare for faster processing through the `\dump` of a preformatted macro package that will not need to read in any packages or font definitions from disk.

```

456 \input{8rphv.fd}%
457 \input{8rptm.fd}%
458 \input{ot1phv.fd}%
459 \input{ot1ptm.fd}%
460 \input{t1ptm.fd}%

```

### 6.4 More math substitutions

The following definitions arrange to get certain glyphs from the text font instead of out of math pi fonts. In particular, the copyright and registered symbols are single glyphs instead of composites involving the big circle from the `cmsy` font.

```

461 \def\eightRChar#1{{\def\encodingdefault{8r}\fontencoding\encodingdefault\selectfont\char"#1}}%
462 \def\LANGLE{<}&{\eightRChar{8B}}%
463 \def\RANGLE{>}&{\eightRChar{9B}}%
464 %\def\ASTER{\eightRChar{2A}}%
465 %\def\DAGGER{\eightRChar{86}}%
466 %\def\DDAGGER{\eightRChar{87}}%
467 \def\BULLET{\eightRChar{95}}%
468 %\def\SECTION{\eightRChar{A7}}%
469 %\def\PARAGRAPH{\eightRChar{B6}}%
470 \def\VERTBAR{\eightRChar{7C}}%
471 \def\COPYRIGHT{\eightRChar{A9}}%
472 \def\REGISTERED{\eightRChar{AE}}%
473 \def\textbar{\VERTBAR}%

```

```

474 \def\textbullet{\BULLET}%
475 \def\textcopyright{\COPYRIGHT}%
476 \def\textregistered{\REGISTERED}%

```

I have removed `\ensuremath` from the following definition, and all commands like `\mathsection` have been converted to e.g., `\textsection`.

```

477 \def\@makefnmark{\@thefnmark}%
478 \def\@fnsymbol#1{{\ifcase#1\or *\or \dagger\or \ddagger\or
479 \textsection\or \textparagraph\or \|\or **\or \dagger\dagger
480 \or \ddagger\ddagger \else\@ctrerr\fi}}

```

## 6.5 End of the fonts DOCSTRIP module

Here ends the module.

```

481 %</fonts>

```

# 7 Programming Conventions

In writing the above code, I cleave to certain conventions, noted here. My goal in explaining them is to encourage others maintaining this body of code to consider following them as well. The benefits are twofold: Some of the coding conventions aim to avoid programming pitfalls; following them reduces maintenance costs. Others make the code easier to read; following these eases the process of understanding how the code works.

And, for my part, I prefer to read code of this type.

## 7.1 Whitespace Conventions

Exactly where code lines break and indent, and where additional whitespace is inserted is explained here.

- Each new macro definition or register assignment begins a new line. Therefore, `\def`, `\newcommand`, and their ilk will start in column 1.
- Code is indented one space for each level of nesting within braces `{}`.
- Likewise, if possible, for `\if...` and matching `\fi`.
- However, the closing brace or `\fi` is outdented by one so that it falls at the same level of indentation as its matching brace or `\if`, and it appears alone on its line.
- Use of the `tab` character is deprecated (tabs are not standardized across all applications and operating systems).
- Lines of code are limited to 72 characters. I follow this convention mostly to ease the transmission of files via email (a deprecated practice) and to accommodate people with small monitors. But `ltxdoc` output looks better with the shorter lines, too.

- Extraneous whitespace in the replacement part of a macro definition is avoided by using the comment character `%`. In most cases, if falling at the end of a line of code, a brace will be immediately followed by a comment character, as will the macro parameter `#1` and any one-letter control sequence, like `\\`.
- Extraneous whitespace in the package file is also avoided. When `TEX` reads in the `.sty` file, it will process `\def`s, and other commands, but will not process blank spaces. This practice is simply a discipline. You don't need to do this. But sometimes `TEX` has to read in a file while it is in horizontal mode, at which point this practice is essential.

These conventions taken together are illustrated by the following:

```

%\def\prepdef#1#2{%
% \ifxundefined#1{\toks@{}}{\toks@\expandafter{#1}}%
% \toks@ii{#2}%
% \edef#1{\the\toks@ii\the\toks@}%
%}%
%
```

In the above, the definition of `\prepdef` would not fit on a single line, and required breaking. The first and last lines have matching braces, and are at the same level of indentation, with the last line containing a single brace.

Each of the three intervening lines has balanced braces and is indented by one space. Each line that would otherwise end in a single brace character is terminated by a comment character.

Some coders rely on the fact that a space character seen by `TEX`'s scanner while in vertical mode is a no-op. Be that as it may, I eliminate them unless actually intentional.

## 7.2 Conventions For Procedures

Here are some of my preferences when writing procedures:

- I dislike defining a macro within another macro, especially when the pattern part is non-nil. You know, you are not saving much space in `mem` when you do this, right? You do save space in the hash table and the string pool, though. On the other hand, we are not dealing with small `TEX` engines anymore; Team `LATEX` has made sure of this.
- If two or more macros have very similar replacement parts, consider layering.
- Macros may perform parsing, may maintain tokens or registers, or may set type (produce marks). I try to avoid combining the three functions in a single macro.
- When a procedure both does assignments and sets type, I try to have a clean separation between the two activities. Try to avoid:

```
% \vskip10pt
% \parindent=0pt
% \leavevmode
%
```

- The Boolean calculus (cf. `\@ifx`) is very useful and produces code that executes nicely. Using it also helps avoid your having to debug problems where `\if...` and `\fi` are not properly balanced (a nightmare, in case you have not already experienced it).

### 7.3 Conventions For L<sup>A</sup>T<sub>E</sub>X

Team L<sup>A</sup>T<sub>E</sub>X make certain recommendations in `clsguide.tex`. Ones that I particularly pay attention to are:

- For the sake of “color safety”, use `\sbox` rather than `\setbox`, `\mbox` rather than `\hbox`, and `\parbox` or `minipage` rather than `\vbox`.
- Use `\newcommand` and `\newenvironment` to declare user-level commands and environments. Avoid the idiom `\def\foo`, `\def\endfoo` to define an environment. Ideally, all user-level markup could be extracted from the document class by grepping on `\newcommand` and `\newenvironment`.
- Prefer to use `\setlength` to assign registers.

I cannot help but notice that much of the code of L<sup>A</sup>T<sub>E</sub>X itself fails to comply with many of the coding recommendations of Team L<sup>A</sup>T<sub>E</sub>X.

## Index

Symbols	
<code>\$TEXMF/</code> .....	3
<code>\,</code> .....	188
<code>\-</code> .....	188
<code>.dtx</code> .....	1, 5
<code>.tfm</code> .....	1
<code>.vf</code> .....	1
<code>.vpl</code> .....	1
<code>\/</code> .....	202, 254, 268
<code>\:</code> .....	254, 265
<code>\&lt;</code> .....	162, 170, 175
<code>\&gt;</code> .....	163, 171, 176
<code>\@Hline</code> .....	337
<code>\@cmd</code> .....	218, 219
<code>\@ctrerr</code> .....	480
<code>\@currsize</code> .....	155
<code>\@endpetrue</code> .....	343
<code>\@file</code> .....	242, 243
<code>\@fnsymbol</code> .....	478
<code>\@gt</code> .....	173, 176
<code>\@gtr@err</code> .....	173, 178
<code>\@idxitem</code> .....	298
<code>\@ifnextchar</code> .....	172, 173
<code>\@ifundefined</code> .....	16
<code>\@ifx</code> .....	22
<code>\@latex@warning@no@line</code> ..	407
<code>\@lt</code> .....	172, 175
<code>\@mainmatterfalse</code> .....	312, 326
<code>\@mainmattertrue</code> .....	317
<code>\@makefnmark</code> .....	477
<code>\@meta</code> .....	172, 174
<code>\@openrighttrue</code> .....	309
<code>\@restonecolfalse</code> .....	278
<code>\@restonecoltrue</code> .....	280
<code>\@sanitize</code> .....	242, 288
<code>\@startsection</code> .....	349
<code>\@tempa</code> .....	195, 196, 260–262
<code>\@tempb</code> .....	191, 196
<code>\@thefnmark</code> .....	477
<code>\@to</code> .....	331, 336
<code>\@ttt</code> .....	288, 289
<code>\@undefined</code> .....	406
<code>\@unused</code> .....	412
<code>\@url</code> .....	254, 258, 287
<code>\{</code> .....	134
<code>\}</code> .....	134
<code>\_</code> .....	36, 146
<code>\‘</code> .....	188
<code>\ </code> .....	167, 479
<code>\_</code> ..	31, 52, 55, 56, 60, 87, 89, 95, 100, 107, 115, 361
<b>A</b>	
<code>\abovedisplayskip</code> .....	335
acrofont document class ..	1, 2, 8, 10, 17
acrofont.sty .....	1–3
acrofont.sty document class ..	3
<code>\active</code> ..	162, 163, 170, 171, 254, 265, 268, 271
<code>\active@colon</code> .....	265, 274, 275
<code>\active@dot</code> .....	271, 274, 275
<code>\active@slash</code> .....	268, 274, 275
<code>\aftergroup</code> ..	265, 268, 271, 342
<code>\allowbreak</code> .....	266, 269, 272
<code>\arg</code> .....	9
<code>\arg</code> .....	206
argument	
foo .....	9
name .....	9
text .....	9
<code>\arraycolsep</code> .....	416
article document class .....	18
<code>\ASTER</code> .....	464
<code>\AtBeginDocument</code> .....	166, 255
<code>\AtEndDocument</code> .....	393
<code>\author</code> .....	34
<b>B</b>	
<code>\backmatter</code> .....	9
<code>\backmatter</code> .....	320
<code>\begin</code> .....	9
<code>\belowdisplayskip</code> .....	341
book document class .....	9, 18

book.cls document class . . . . 14  
\botrule . . . . . 10  
\botrule . . . . . 347  
\BULLET . . . . . 467, 474

**C**

\c@secnumdepth . . . 290, 291, 295  
\catcode . 162, 163, 170, 171, 254,  
265, 268, 271  
\char . . . . 175, 176, 224, 228, 461  
\class@base . . . . . 420  
\class@name . . 143, 144, 421, 422  
\ClassError . . . . . 179  
\ClassInfo . . . . . 422  
\classname . . . . . 9  
\classname 25, 46, 54, 68, 77, 78,  
121, 122, 124, 126, 249  
\classoption . . . . . 9  
\classoption . . . . . 248  
\cleardoublepage . . . . . 329  
\clearpage . . . . . 299, 324  
\cleartorecto 311, 316, 322, 328,  
329  
\closeout . . . . . 404  
clsguide.tex . . . . . 22  
\cmd . . . . . 9  
\cmd . 134, 218, 226, 234, 235, 286  
\cmd@to@cs . . . 221, 224, 230, 286  
\cmd@to@index . . . . . 222, 225  
cmsy . . . . . 19  
\colon@break . . . . . 266, 274  
\colon@char . . . . 266, 267, 275  
\colrule . . . . . 10  
\colrule . . . . . 346  
\columnsep . . . . . 283  
\columnseprule . . . . . 282  
\COPYRIGHT . . . . . 471, 475  
\copyright . . . . . 31  
\cs . . . . . 221, 228, 230, 286  
\csname . . . . . 9  
\csname . . . . . 17, 195, 414

**D**

\DAGGER . . . . . 465  
\dagger . . . . . 478, 479  
\DDAGGER . . . . . 466

\ddagger . . . . . 478, 480  
\DeclareFontEncoding . . . . 451  
\DeclareFontFamily . . 428, 433,  
440, 445, 452  
\DeclareFontShape 429, 432, 434,  
437–439, 441, 444, 446, 449,  
453  
\DeclareRobustCommand 197, 201,  
206, 211, 218, 228, 237, 238,  
241, 242, 247–249, 254, 310,  
315, 320, 348  
\def . . . . . 20–22  
\do . . . . . 188  
doc . . . . . 5  
doc/ . . . . . 3  
\DocInput . . . . . 8  
document class  
    acrofont . . . . . 1, 2, 8, 10, 17  
    acrofont.sty . . . . . 3  
    article . . . . . 18  
    book . . . . . 9, 18  
    book.cls . . . . . 14  
    ltxdoc . 1, 2, 5, 8–10, 16, 20  
    ltxdoc.cls . . . . . 2, 14  
    ltxdocext . . . . . 1  
    ltxdocext.dtx . . . . . 3  
    ltxdocext.pdf . . . . . 3  
    ltxdocext.sty . . . . . 3, 10  
    ltxguide.cls . . . . . 2, 11  
    report . . . . . 18  
document environment . . . . . 5  
\DoNotIndex . . . . . 10  
\dot@break . . . . . 272, 274  
\dot@char . . . . . 272, 273, 275  
\dump . . . . . 19

**E**

\edef . . . . . 195, 260  
\eightRChar . . . . . 461–472  
\emph . . . . . 87  
\encodingdefault . . . . . 461  
\end . . . . . 9  
\endcsname . . . . . 17, 195, 414  
\endfilecontents . . . . . 395, 414  
\endfoo . . . . . 22  
\ensuremath . . . . . 20



<code>\env</code>	9	<code>revtex/</code>	3
<code>\env</code>	237, 239, 241	<code>source/</code>	3
<code>\envb</code>	9	<code>tex/</code>	3
<code>\envb</code>	238	<code>texmf-local/</code>	3
<code>\enve</code>	9	<code>times.sty</code>	18
<code>\enve</code>	241	<code>zpsycmrv</code>	19
environment		<code>zpsynocmrv</code>	1, 19
<code>document</code>	5	<code>zptmcmr</code>	18
<code>filecontents</code>	2, 17	<code>zptmcmrm</code>	18
<code>minipage</code>	22	<code>zptmnocmr</code>	1, 18
<code>quote</code>	10	<code>zptmnocmrm</code>	1, 18
<code>theindex</code>	14	<code>zpzccmry</code>	18
<code>unnumtable</code>	9	<code>zpzcnocmry</code>	1, 18
<code>verbatim</code>	2, 8, 11	<code>\file</code>	9
environments:		<code>\file</code>	65, 70–72, 75, 76, 84, 85, 94, 100–102, 107, 110, 113, 114, 121, 122, 124, 126, 128, 131, 242
<code>unnumtable</code>	331	<code>filecontents environment</code>	2, 17
<b>F</b>			
<code>\fi</code>	20, 22	<code>\filedate</code>	9, 17
file		<code>\fileinfo</code>	9
<code>\$TEXMF/</code>	3	<code>\fileinfo</code>	194
<code>.dtx</code>	1, 5	<code>\fileversion</code>	9, 17
<code>.tfm</code>	1	<code>\font</code>	433, 452
<code>.vf</code>	1	<code>\fontencoding</code>	461
<code>.vpl</code>	1	fonts	2, 18, 20
<code>acronym.sty</code>	1–3	<code>\foo</code>	22
<code>clsguide.tex</code>	22	foo, argument	9
<code>cmsy</code>	19	<code>\frontmatter</code>	9
<code>doc</code>	5	<code>\frontmatter</code>	310
<code>doc/</code>	3	<b>G</b>	
<code>fonts</code>	2, 18, 20	<code>\GetFileInfo</code>	9, 17
<code>kernel</code>	2, 11, 17	<code>\GetFileInfo</code>	22, 189
<code>latex/</code>	3	<b>H</b>	
<code>ltxdoc.cfg</code>	2, 16	<code>\hbox</code>	22
<code>ltxdocext.dtx</code>	1–3	<code>\hline</code>	345–347
<code>ltxdocext.pdf</code>	1	<code>\hyphenchar</code>	452
<code>ltxdocext.sty</code>	1–3	<b>I</b>	
<code>ltxguide.cls</code>	12	<code>\I</code>	411
<code>makeindex</code>	3	<code>\if</code>	20, 22
<code>mathptm.sty</code>	18	<code>\if@mainmatter</code>	307
<code>omlptmcm.fd</code>	18	<code>\if@openright</code>	308, 321
<code>omspzccm.fd</code>	18	<code>\if@restonecol</code>	299
<code>omxpsycm.fd</code>	19		
<code>otlptmcm.fd</code>	18		
<code>package</code>	2, 10, 11, 17		
<code>README</code>	3, 5		



`\providecommand` . . . . . 345–347  
`\ProvidesClass` . . . . . 9  
`\ProvidesFile` . . . . . 9  
`\ProvidesFile` . . . . . 4, 5, 7

### Q

`quote` environment . . . . . 10

### R

`\RANGLE` . . . . . 202, 205, 463  
`\rangle` . . . . . 205  
`README` . . . . . 3, 5  
`\RecordChanges` . . . . . 20  
`\REGISTERED` . . . . . 472, 476  
`\renewenvironment` . . . . . 276, 300  
`report` document class . . . . . 18  
`\RequirePackage` . 13, 14, 17, 150,  
165, 426, 427  
`\reserved@c` . . . . . 396, 404  
`revtex/` . . . . . 3

### S

`\save@secnumdepth` . . . . . 290, 295  
`\sbox` . . . . . 22  
`\sc` . . . . . 86, 103  
`\scmd` . . . . . 229  
`\scmd@to@index` . . . . . 231, 233  
`\SECTION` . . . . . 468  
`\section` . . . . . 82, 292  
`\see` . . . . . 284  
`\seealso` . . . . . 285  
`\setbox` . . . . . 22  
`\setlength` . . . . . 22  
`\skewchar` . . . . . 433  
`\slash@break` . . . . . 269, 274  
`\slash@char` . . . . . 269, 270, 275  
`\small` . . . . . 156  
`source/` . . . . . 3  
`\StopEventually` . . . . . 5  
`\string` . . 199, 207, 208, 215, 216,  
221, 222, 226, 230, 231, 234,  
235, 239, 241, 245, 252, 260,  
286, 397, 402  
`\subsection` . . . . . 92  
`\substyle` . . . . . 9  
`\substyle` . . . . . 247

`\subsubsection` . . . . . 10, 15  
`\subsubsubsection` . . . . . 348

### T

`\T` . . . . . 405, 410, 411  
`\table head rows` placeholder . . 10  
`\table rows` placeholder . . . 9, 10  
`\tableofcontents` . . . . . 80  
`\TeX` . . . . . 52, 60, 100  
`tex/` . . . . . 3  
`texmf-local/` . . . . . 3  
`text`, argument . . . . . 9  
`\textbar` . . . . . 473  
`\textbullet` . . . . . 474  
`\textcopyright` . . . . . 475  
`\textit` . . . . . 284, 285  
`\textparagraph` . . . . . 479  
`\textregistered` . . . . . 476  
`\textsection` . . . . . 20  
`\textsection` . . . . . 479  
`\texttt` . . 36, 113, 114, 118, 128,  
131, 134  
`\thanks` . . . . . 26, 30, 36  
`theindex` environment . . . . . 14  
`\thispagestyle` . . . . . 296  
`times.sty` . . . . . 18  
`\title` . . . . . 24  
`\toprule` . . . . . 10, 15  
`\toprule` . . . . . 345  
`\ttfamily` 161, 207, 212, 213, 228,  
239, 241, 244, 251, 259, 288  
`\ttt` 208, 215, 216, 245, 252, 287,  
288  
`\twocolumn` . . . . . 292

### U

`unnumtable` (environment) . . 331  
`unnumtable` environment . . . . . 9  
`\unskip` . . . . . 153, 339  
`\url` . . . . . 9  
`\url` . . . . . 50, 57, 256, 260  
`\url@break` . . . . . 259, 274, 287  
`\url@char` . . . . . 260, 275  
`\url@ltxdocext` . . . . . 254, 256  
`\usepackage` . . . . . 3

## V

<code>\vbox</code> .....	22
<code>\verb</code> .....	8, 11
<code>\verbatim</code> .....	151, 152
verbatim environment ..	2, 8, 11
<code>\verbatim@font</code> .....	160
<code>\verbatim@nolig@list</code> .....	188
<code>\VERTBAR</code> .....	470, 473

## W

<code>\write</code> .....	396, 412
---------------------------	----------

## Z

<code>zpsycmrv</code> .....	19
<code>zpsynocmrv</code> .....	1, 19
<code>zptmcmr</code> .....	18
<code>zptmcmrm</code> .....	18
<code>zptmnocmr</code> .....	1, 18
<code>zptmnocmrm</code> .....	1, 18
<code>zpzccmry</code> .....	18
<code>zpzcnocmry</code> .....	1, 18