

A L^AT_EX Package of utility macros *

Arthur Ogawa (<mailto:ogawa@teleport.com>), 0.1
Copyright (C) 1999 Arthur Ogawa

October 1, 2020

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

The distribution point for this work is `ftp://ftp.teleport.com/users/ogawa/macros/latex/con` which contains fully unpacked, prebuilt runtime files and documentation.

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

To use this document class, you must have a working T_EX installation equipped with L^AT_EX 2_ε 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 file `ltxutil.sty` into a location in your filesystem where it will be found by L^AT_EX.

To use, read the user documentation `ltxutil.pdf`.

Contents

1	Processing Instructions	2
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>	3
1.2.3	Generated by <code>tex ltxdocext.ins</code>	3
1.2.4	Documentation	3
1.2.5	Auxiliary	4
2	Code common to all modules	4
3	The driver module <code>doc</code>	4
3.1	Stubs for <code>ltxdocext.sty</code> and <code>acrofont.sty</code>	4
3.2	The Preamble	5

*This file has version number 0.1, last revised 2020/09/30. For version number and date, search on "0.1" in the `.dtx` file, or see the end of the `00readme.txt` file.

3.2.1	Docstrip and info directives	5
3.3	The installer file	5
3.4	The “Read Me” File	6
3.5	The Document Body	8
4	Using the ltxdoc and acrofont packages	9
4.1	Extensions to the ltxdoc class	9
4.1.1	Extensions to the <code>verbatim</code> environment and <code>\verb</code> command	9
4.1.2	The <code>-matter</code> Commands Work	9
4.1.3	The <code>\GetFileInfo</code> command	9
4.1.4	Self-Indexing Commands	10
4.1.5	Unnumbered Tables	10
4.1.6	Structuring Tables	10
4.1.7	A Sectioning Command Below <code>\subsection</code>	10
4.2	Alterations to the ltxdoc class	11
5	Extensions to the ltxdoc class	11
5.1	Beginning of the <code>extensions</code> DOCSTRIP module	11
5.2	Incorporate <code>ltxguide.cls</code> extensions	11
5.3	Changes to the base class of the ltxdoc class	14
5.4	Extensions to the base class of <code>ltxdoc.cls</code>	15
5.5	In lieu of <code>ltxdoc.cfg</code>	16
5.6	Extension to L ^A T _E X’s <code>filecontents</code> Environment	17
5.7	End of the <code>extensions</code> DOCSTRIP module	18
6	Font Package for Acrobat Compatability	18
6.1	Beginning of the <code>fonts</code> DOCSTRIP module	18
6.2	Variants on <code>psfonts</code> packages	18
6.3	Font definition files	19
6.4	More math substitutions	19
6.5	End of the <code>fonts</code> DOCSTRIP module	20
7	Programming Conventions	20
7.1	Whitespace Conventions	20
7.2	Conventions For Procedures	21
7.3	Conventions For L ^A T _E X	22
	Index	23

1 Processing Instructions

The package file `ltxutil.sty` is generated from this file, `ltxutil.dtx`, using the DOCSTRIP facility of L^AT_EX via `tex ltxutil.ins`. The typeset documentation that you are now reading is generated from the same file by typesetting it with L^AT_EX or `pdftex` via `latex ltxutil.dtx` or `pdflatex ltxutil.dtx`.

1.1 Build Instructions

You may bootstrap this suite of files solely from `ltxutil.dtx`. Prepare by installing $\text{\LaTeX} 2_{\epsilon}$ (and either `tex` or `pdftex`) on your computer, then carry out the following steps:

1. Within an otherwise empty directory, typeset `ltxutil.dtx` with \LaTeX or `pdflatex`; you will obtain the typeset documentation you are now reading, along with the installer `ltxutil.ins`, and the file `00readme.txt`.

Note: you will have to run \LaTeX twice, then `makeindex`, then \LaTeX again in order to obtain a valid index and table of contents.

2. Now typeset `ltxutil.ins`, thereby generating the package file `ltxutil.sty`.
3. Install `ltxutil.sty` by moving it to a location in your filesystem where they will be found by \LaTeX .

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 \LaTeX generates the readme and the installer.

```
%00readme.txt  ltxdocext.ins
%
```

1.2.3 Generated by `tex ltxdocext.ins`

Typesetting the installer generates the package files.

```
%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^AT_EX:

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

2 Code common to all modules

The following may look a bit klutzy, but 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^AT_EX 2_ε. An appropriate message is displayed if a different T_EX format is used.

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

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

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

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 [2020/09/30 0.1 ltxdoc extensions package]% \fileversion
```

3 The driver module doc

This module, consisting of the present section, typesets the programmer's documentation, generating the `.ins` installer and `00readme.txt` 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.

```
10 %<*doc>
```

3.1 Stubs for `ltxdocext.sty` and `acrofont.sty`

This `filecontents` environment generates stub files `ltxdocext.sty` and `acrofont.sty`. Stubs are needed because the documentation for this package requires them and they might not yet be present.

A real version of these files will be generated when you typeset `ltxdocext.ins`, at which point the stub file will be overwritten (you will have to give DOCSTRIP permission to do this).

```
11 \begin{filecontents}{ltxdocext.sty}
12 % Stub version of ltxdocext.sty
13 \end{filecontents}
14 \begin{filecontents}[overwrite]{acrofont.sty}
15 % Stub version of acrofont.sty
16 \end{filecontents}
```

3.2 The Preamble

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

```
17 \documentclass[draft]{ltxdoc}
18 \RequirePackage{ltxdocext}%
19 \RequirePackage{acrofont}%
20 \CodelineIndex\EnableCrossrefs
```

3.2.1 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}%
```

3.3 The installer file

The installer `ltxdocext.ins` appears here. If you have retrieved the standard distribution of this package, the installer file is already on your filesystem. If you are bootstrapping, the first typesetting of the `.dtx` file will cause the installer to be generated.

The following modules are used to direct DOCSTRIP in generating the external files:

Module	File	Description
doc	<code>ltxdocext.drv</code>	driver for programmer's documantation
extensions	<code>ltxdocext.sty</code>	ltxdoc extensions package
fonts	<code>acrofont.sty</code>	package to use only Acrobat fonts

```
23 \begin{filecontents}{ltxdocext.ins}
24 %% This file will generate documentation and runtime files
25 %% from ltxdocext.dtx when run through LaTeX or TeX.
26 \input docstrip
27 \preamble
28
29 This is a generated file;
```

```

30 altering it directly is inadvisable;
31 instead, modify the original source file.
32 See the URL in the file 00readme.txt.
33
34 Copyright notice.
35
36   These files are distributed
37   WITHOUT ANY WARRANTY; without even the implied warranty of
38   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
39
40 \endpreamble
41 \keepsilent
42 \generate{%
43   \file{ltxdocext.drv}{\from{ltxdocext.dtx}{doc}}%
44   \file{ltxdocext.sty}{\from{ltxdocext.dtx}{extensions}}%
45   \file{acrofont.sty}{\from{ltxdocext.dtx}{fonts}}%
46 }%
47 \ifToplevel{
48 \Msg{*****}
49 \Msg{*}
50 \Msg{* To finish the installation, please move}
51 \Msg{*   ltxdocext.sty}
52 \Msg{* into a directory searched by TeX.}
53 \Msg{*}
54 \Msg{* To produce the user documentation,
55       run ltxdocext.tex through LaTeX.}
56 \Msg{*}
57 \Msg{* To produce the programmer documentation,
58       run ltxdocext.dtx through LaTeX.}
59 \Msg{*}
60 \Msg{* Happy TeXing}
61 \Msg{*****}
62 }
63 \endbatchfile
64 \end{filecontents}

```

Note that, because all of the files generated by the installer are part of the standard distribution, it will be necessary to run the installer only when bootstrapping (or, of course, during development). Note, too, that it is rare to generate the doc module because it suffices to simply typeset the .dtx file itself.

3.4 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.

```
65 \begin{filecontents*}{00readme.txt}
66 \title{%
67 Extensions to the \classname{ltxdoc} class%
68 \thanks{%
69 This file has version number \fileversion,
70 last revised \filedate.%
71 % For version number and date, search on "\fileversion" in the .dtx file,
72 % or see the end of the 00readme.txt file.
73 }%
74 }%
75
76 \author{%
77 Arthur Ogawa (\texttt{mailto:ogawa@teleport.com}),
78 \fileversion\Copyright (C) 1999 Arthur Ogawa
79 }%
80 \maketitle
81
82 This file embodies the \classname{ltxdocext} package,
83 the implementation and its user documentation.
84
85 The distribution point for this work is
86 \url{ftp://ftp.teleport.com/users/ogawa/macros/latex/contrib/supported/ltxdocext...},
87 which contains fully unpacked, prebuilt runtime files and documentation.
88
89 To use this document class, you must have a working
90 \TeX\ installation equipped with \LaTeX\
91 and possibly pdftex and Adobe Acrobat Reader or equivalent.
92
93 To install, retrieve the distribution,
94 unpack it into a directory on the target computer,
95 and move the files \file{ltxdocext.sty} and \file{acrofont.sty}
96 into a location in your filesystem where they will be found by \LaTeX.
97
98 If you will be using the \classname{acrofont} package, you must
99 also install the virtual fonts
100 \file{zpsynocmr}, \file{zptmnochr},
101 \file{zptmnochr}, and \file{zpcnocmry}.
102 The corresponding \file{.tfm}, \file{.vf}, and \file{.vpl}
103 files are part of this distribution.
104
105 To use, read the user documentation \file{ltxdocext.pdf}.
106 The \file{.dtx} file, \file{ltxdocext.dtx}, constitutes
107 in itself an instance of use of the \classname{ltxdocext}
108 package and the \classname{acrofont} package.
109
```

```

110 \tableofcontents
111
112 \section{Processing Instructions}
113
114 The package files \file{ltxdocext.sty} and \file{acrofont.sty}
115 are generated from this file, \file{ltxdocext.dtx},
116 via the {\sc docstrip} facility of \LaTeX
117 via |tex ltxdocext.ins|.
118 The typeset documentation that you are now reading is generated from
119 the same file by typesetting it with \LaTeX\ or pdftex
120 via |latex ltxdocext.dtx| or |pdflatex ltxdocext.dtx|.
121
122 \subsection{Build Instructions}
123
124 You may bootstrap this suite of files solely from \file{ltxdocext.dtx}.
125 Prepare by installing \LaTeXe\ (and either tex or pdftex) on your computer,
126 then carry out the following steps:
127 \begin{enumerate}
128 \item
129 Within an otherwise empty directory,
130 typeset \file{ltxdocext.dtx} with \LaTeX\ or pdflatex;
131 you will obtain the typeset documentation you are now reading,
132 along with
133 the installer \file{ltxdocext.ins},
134 and the file \file{00readme.txt}.
135 \item
136 Now typeset \file{ltxdocext.ins},
137 thereby generating the package file \file{ltxdocext.sty},
138 and the package file \file{acrofont.sty}.
139 Make sure that {\sc docstrip} receives permission
140 to overwrite existing versions of these packages.
141 \item
142 Install \classname{ltxdocext.sty} and \classname{acrofont.sty}
143 by moving them to a location
144 in your filesystem where they will be found by \LaTeX.
145 \item
146 Now complete the typesetting of the documentation by
147 retypesetting \file{ltxdocext.dtx}.
148 Note: you will have to run \LaTeX\ twice, then \file{makeindex}, then
149 \LaTeX\ again in order to obtain a valid index and table of contents.
150 \end{enumerate}
151 \end{filecontents*}

```

3.5 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.

```

152 \begin{document}%
153 \expandafter\DocInput\expandafter{\jobname.dtx}%

```



```

154 % ^^A\PrintChanges
155 \end{document}
156 %</doc>

```

4 Using the ltxdoc and acrofont packages

These packages are an adjunct to the standard L^AT_EX 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^AT_EX 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>\m{<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 `extensions` `DOCSTRIP` module comprises the package `ltxdocext.sty`, which provides extensions to the standard L^AT_EX `ltxdoc` class.

5.1 Beginning of the extensions `DOCSTRIP` module

```
157 %<*extensions>
158 \def\class@name{ltxdocext}%
159 \expandafter\PackageInfo\expandafter{\class@name}{%
160 An extension to the \protect\LaTeXe\space ltxdoc class
161 by A. Ogawa (ogawa@teleport.com)%
162 }%
```

5.2 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^AT_EX installation.”

```
163 \RequirePackage{verbatim}%
164 \let\o@verbatim\verbatim
165 \def\verbatim{%
166   \ifhmode\unskip\par\fi
167   % \nopagebreak           % Overridden by list penalty
168   \ifx\@currsize\normalsize
169     \small
170   \fi
171   \o@verbatim
172 }%
```

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

```
173 \renewcommand \verbatim@font {%
174   \normalfont \ttfamily
175   \catcode'\<=\active
176   \catcode'\>=\active
177 }%
```

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

```
178 \RequirePackage{shortvrb}
179 \AtBeginDocument{%
180   \MakeShortVerb{|\|}%
181 }%
```

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

```
182 \begingroup
183   \catcode'\<=\active
184   \catcode'\>=\active
185   \gdef<{\@ifnextchar<\lt\@meta}
186   \gdef>{\@ifnextchar>\gt\@gtr@err}
187   \gdef\@meta#1>{\m{#1}}
188   \gdef\@lt<{\char'\<}
189   \gdef\@gt>{\char'\>}
190 \endgroup
191 \def\@gtr@err{%
192   \ClassError{ltxguide}{%
193     Isolated \protect>%
194   }{%
195     In this document class, \protect<...\protect>
196     is used to indicate a parameter.\MessageBreak
197     I've just found a \protect> on its own.
198     Perhaps you meant to type \protect>\protect>?
199   }%
200 }
201 \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`):

```
202 \def\GetFileInfo#1{%
203   \def\filename{#1}%
204   \def\@tempb##1 ##2 ##3\relax##4\relax{%
205     \def\filedate{##1}%
206     \def\fileversion{##2}%
207     \def\fileinfo{##3}}%
208   \edef\@tempa{\csname ver@#1\endcsname}%
209   \expandafter\@tempb\@tempa\relax? ? \relax\relax}
```

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

Various forms of self-indexing commands:

```

210 \DeclareRobustCommand{\m}[1]{%
211   \meta{#1}%
212   \index{#1=\string\meta{#1} placeholder}\index{placeholder>#1=\string\meta{#1}}%
213 }%
214 \DeclareRobustCommand\meta[1]{%
215   \mbox{\LANGLE\itshape#1\/\RANGLE}%
216 }%
217 \def\LANGLE{\langle}%
218 \def\RANGLE{\rangle}%
219 \DeclareRobustCommand{\arg}[1]{%
220   {\ttfamily\string}\meta{#1}{\ttfamily\string}}%
221   \index{#1=\string\ttt{#1}, argument}\index{argument>#1=\string\ttt{#1}}%
222 }%
223 \let\oarg\undefined
224 \DeclareRobustCommand{\oarg}[1]{%
225   {\ttfamily[%]
226   }\meta{#1}{\ttfamily%[
227   ]}%
228   \index{#1=\string\ttt{#1}, optional argument}%
229   \index{argument, optional>#1=\string\ttt{#1}}%
230 }%
231 \DeclareRobustCommand\cmd{\begingroup\makeatletter\@cmd}%
232 \def\@cmd#1{%
233   \endgroup
234   \cs{expandafter\cmd@to@cs\string#1}%
235   \expandafter\cmd@to@index\string#1\@nil
236 }%
237 \def\cmd@to@cs#1#2{\char\number‘#2\relax}%
238 \def\cmd@to@index#1#2\@nil{%
239   \index{#2=\string\cmd#1#2}\index{command>#2=\string\cmd#1#2}%
240 }%
241 \DeclareRobustCommand\cs[1]{\ttfamily\char‘\#1}%
242 \def\scmd#1{%
243   \cs{expandafter\cmd@to@cs\string#1}%
244   \expandafter\scmd@to@index\string#1\@nil
245 }%
246 \def\scmd@to@index#1#2\@nil#3{%
247   \index{\string$#3=\string\cmd#1#2---#3}%
248   \index{command>\string$#3=\string\cmd#1#2---#3}%
249 }%
250 \DeclareRobustCommand\env{\name@idx{environment}}%
251 \DeclareRobustCommand\envb[1]{%
252   {\ttfamily\string\begin\string}\env{#1}{\ttfamily\string}}%
253 }%
254 \DeclareRobustCommand\enve[1]{\ttfamily\string\end\string}\env{#1}{\ttfamily\string}}%
255 \DeclareRobustCommand*\file[1]{%
256   {\ttfamily#1}%
257   \index{#1=\string\ttt{#1}}\index{file>#1=\string\ttt{#1}}%
258 }%
259 \DeclareRobustCommand\substyle{\name@idx{document substyle}}%

```

```

260 \DeclareRobustCommand\classoption{\name@idx{document class option}}%
261 \DeclareRobustCommand\classname{\name@idx{document class}}%
262 \def\name@idx#1#2{%
263   {\ttfamily#2}%
264   \index{#2\space#1=\string\ttt{#2}\space#1}\index{#1>#2=\string\ttt{#2}}%■
265 }%
266 \DeclareRobustCommand\url@ltxdocext{\begingroup\catcode'\active\catcode'\active\catcode
267 \AtBeginDocument{%
268   \ifx\url\undefined\let\url\url@ltxdocext\fi
269 }%
270 \def\@url#1{%
271   \url@break{\ttfamily#1}%
272   \url@char\edef\@tempa{#1=\string\url{#1}}%
273   \expandafter\index\expandafter{\@tempa}%
274   \expandafter\index\expandafter{\expandafter u\expandafter r\expandafter l\expandafter >\@tempa}
275   \endgroup
276 }%
277 {\catcode'\active\aftergroup\def\aftergroup:}{\active@colon}%
278 \def\colon@break{\colon@char\allowbreak}%
279 \def\colon@char{:}%
280 {\catcode'\active\aftergroup\def\aftergroup/}{\active@slash}%
281 \def\slash@break{\slash@char\allowbreak}%
282 \def\slash@char{/}%
283 {\catcode'\active\aftergroup\def\aftergroup.}{\active@dot}%
284 \def\dot@break{\dot@char\allowbreak}%
285 \def\dot@char{.}%
286 \def\url@break{\let\active@slash\slash@break\let\active@dot\dot@break\let\active@colon\colon@break}
287 \def\url@char{\let\active@slash\slash@char\let\active@dot\dot@char\let\active@colon\colon@char}

```

5.3 Changes to the base class of the ltxdoc class

Modify theindex environment so that it produces a TOC entry

```

288 \renewenvironment{theindex}
289     {\if@twocolumn
290       \@restonecolfalse
291       \else
292         \@restonecoltrue
293         \fi
294         \columnseprule \z@
295         \columnsep 35\p@
296 \def\see##1##2{\textit{See} ##1}%
297 \def\seealso##1##2{\textit{See also} ##1}%
298 \def\cmd##1{\cs{\expandafter\cmd@to\cs\string##1}}%
299 \def\@url##1{\url@break\ttt{##1}\endgroup}%
300 \def\ttt##1{{\ttfamily##1}}%
301 \mathchardef\save@secnumdepth\c@secnumdepth
302 \c@secnumdepth\m@ne
303     \twocolumn[\section{\indexname}]%
304     \mkboth{\MakeUppercase\indexname}%

```

```

305 %                {\MakeUppercase\indexname}%
306 \c@secnumdepth\save@secnumdepth
307                \thispagestyle{plain}\parindent\z@
308                \parskip\z@ \@plus .3\p@\relax
309                \let\item\@idxitem}
310                {\if@restonecol\onecolumn\else\clearpage\fi}
311 \renewenvironment{quote}
312                {\list{}{
313                \leftmargin1em\relax
314                \rightmargin\leftmargin
315                }%
316                \item\relax}
317                {\endlist}

```

5.4 Extensions to the base class of ltxdoc.cls

Matter commands from book.cls

```

318 \newif\if@mainmatter
319 \newif\if@openright
320 \@openrighttrue
321 \DeclareRobustCommand\frontmatter{%
322 \cleartorecto
323 \@mainmatterfalse
324 \pagenumbering{roman}%
325 }%
326 \DeclareRobustCommand\mainmatter{%
327 \cleartorecto
328 \@mainmattertrue
329 \pagenumbering{arabic}%
330 }%
331 \DeclareRobustCommand\backmatter{%
332 \if@openright
333 \cleartorecto
334 \else
335 \clearpage
336 \fi
337 \@mainmatterfalse
338 }%
339 \ifx\undefined\cleartorecto
340 \def\cleartorecto{\cleardoublepage}%
341 \fi

```

Unnumbered tables

`unnumtable` An unnumbered table does not float.

```

342 \def\@to{to}%
343 \newenvironment{unnumtable}{%
344 \par
345 \addpenalty\predisplaypenalty
346 \addvspace\abovedisplayskip

```

```

347 \hbox\@to\hsize\bgroup\hfil\ignorespaces
348 \let\@Hline\@empty
349 }{%-
350 \unskip\hfil\egroup
351 \penalty\postdisplaypenalty
352 \vskip\belowdisplayskip
353 \aftergroup\ignorespaces
354 \@endpetrue
355 }%

```

Emulate \toprule and friends

```

356 \newcommand\toprule{\hline\hline}%
357 \newcommand\colrule{\\\hline}%
358 \newcommand\botrule{\\\hline\hline}%

```

Define sectioning command below \subsubsection.

```

359 \DeclareRobustCommand\subsubsubsection{%
360 \@startsection{subsubsection}{4}%
361 {\z@}{-15\p@\@plus-5\p@\@minus-2\p@}%
362 {5\p@}{\normalfont\normalsize\itshape}%
363 }%

```

5.5 In lieu of ltxdoc.cfg

We don't want everything to appear in the index

```

364 \DoNotIndex{\',\.,\@M,\@input,\@Alph,\@alph,\@addtoreset,\@arabic}
365 \DoNotIndex{\@badmath,\@centercr,\@cite}
366 \DoNotIndex{\@dotsep,\@empty,\@float,\@gobble,\@gobbletwo,\@ignoretrue}
367 \DoNotIndex{\@input,\@ixpt,\@m,\@minus,\@mkboth}
368 \DoNotIndex{\@ne,\@nil,\@nomath,\@plus,\roman,\@set@topoint}
369 \DoNotIndex{\@tempboxa,\@tempcnta,\@tempdima,\@tempdimb}
370 \DoNotIndex{\@tempswafalse,\@tempwatrue,\@viipt,\@viipt,\@vipt}
371 \DoNotIndex{\@vpt,\@warning,\@xiipt,\@xipt,\@xivpt,\@xpt,\@xvipt}
372 \DoNotIndex{\@xxpt,\@xxvpt,\\,\\ ,\addpenalty,\addtolength,\addvspace}
373 \DoNotIndex{\advance,\ast,\begin,\begingroup,\bfseries,\bgroup,\box}
374 \DoNotIndex{\bullet}
375 \DoNotIndex{\cdot,\cite,\CodelineIndex,\cr,\day,\DeclareOption}
376 \DoNotIndex{\def,\DisableCrossrefs,\divide,\DocInput,\documentclass}
377 \DoNotIndex{\DoNotIndex,\egroup,\ifdim,\else,\fi,\em,\endtrivlist}
378 \DoNotIndex{\EnableCrossrefs,\end,\end@dblfloat,\end@float,\endgroup}
379 \DoNotIndex{\endlist,\everycr,\everypar,\ExecuteOptions,\expandafter}
380 \DoNotIndex{\fbox}
381 \DoNotIndex{\filedate,\filename,\fileversion,\fontsize,\framebox,\gdef}
382 \DoNotIndex{\global,\halign,\hangindent,\hbox,\hfil,\hfill,\hrule}
383 \DoNotIndex{\hsize,\hskip,\hspace,\hss,\if@tempwa,\ifcase,\or,\fi,\fi}
384 \DoNotIndex{\ifhmode,\ifvmode,\ifnum,\iftrue,\ifx,\fi,\fi,\fi,\fi,\fi}
385 \DoNotIndex{\input}
386 \DoNotIndex{\jobname,\kern,\leavevmode,\let,\leftmark}
387 \DoNotIndex{\list,\llap,\long,\m@ne,\m@th,\mark,\markboth,\markright}
388 \DoNotIndex{\month,\newcommand,\newcounter,\newenvironment}

```



```

389 \DoNotIndex{\NeedsTeXFormat, \newdimen}
390 \DoNotIndex{\newlength, \newpage, \nobreak, \noindent, \null, \number}
391 \DoNotIndex{\numberline, \OldMakeindex, \OnlyDescription, \p@}
392 \DoNotIndex{\pagestyle, \par, \paragraph, \paragraphmark, \parfillskip}
393 \DoNotIndex{\penalty, \PrintChanges, \PrintIndex, \ProcessOptions}
394 \DoNotIndex{\protect, \ProvidesClass, \raggedbottom, \raggedright}
395 \DoNotIndex{\refstepcounter, \relax, \renewcommand}
396 \DoNotIndex{\rightmargin, \rightmark, \rightskip, \rlap, \rmfamily}
397 \DoNotIndex{\secdef, \selectfont, \setbox, \setcounter, \setlength}
398 \DoNotIndex{\settowidth, \sfcode, \skip, \sloppy, \slshape, \space}
399 \DoNotIndex{\symbol, \the, \trivlist, \typeout, \tw@, \undefined, \uppercase}
400 \DoNotIndex{\usecounter, \usefont, \usepackage, \vfil, \vfill, \viiipt}
401 \DoNotIndex{\vipt, \vipt, \vskip, \vspace}
402 \DoNotIndex{\wd, \xiipt, \year, \z@}
403 \DoNotIndex{\next}

```

Direct ltxdoc to produce an index.

```
404 \AtEndDocument{\PrintIndex}%
```

5.6 Extension to L^AT_EX'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`.

```

405 \makeatletter
406 \def\endfilecontents{%
407   \immediate\write\reserved@c{%
408     \string\iffalse\space ltxdoc klotch^^J%
409     \ifx\undefined\fileversion\else
410       \ifx\undefined\filedate\else
411         This file has version number \fileversion, last revised \filedate.%
412       \fi\fi
413     \string\fi
414   }%
415   \immediate\closeout\reserved@c
416   \def\T##1##2##3{%
417     \ifx##1\@undefined\else
418       \@latex@warning@no@line{##2 has been converted to Blank ##3e}%
419     \fi
420   }%
421   \T\L{Form Feed}{Lin}%
422   \T\I{Tab}{Spac}%
423   \immediate\write\@unused{}%
424 }%
425 \expandafter\let\csname endfilecontents*\endcsname\endfilecontents
426 \makeatother

```

Alter formatting in arrays; set them tight.

```
427 \setlength\arraycolsep{0pt}%
```

5.7 End of the extensions DOCSTRIP module

Here ends the module.

```
428 %</extensions>
```

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^AT_EX book class: <ftp://ctan.tug.org/tex-archive/macros>

```
429 %<*fonts>
430 \def\class@name{ltxdocext}%
431 \expandafter\ClassInfo\expandafter{\class@name}{%
432   Written for \protect\LaTeXe\space
433   by A. Ogawa (ogawa@teleport.com)%
434 }%
```

6.2 Variants on psfonts packages

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

```
435 \RequirePackage{times}%
```

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

```
436 \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.

```
437 \DeclareFontFamily{OT1}{ptmcm}{}
438 \DeclareFontShape{OT1}{ptmcm}{m}{n}{
439   <-> zptmnocmr
440 }{}
441 \DeclareFontShape{OT1}{ptmcm}{l}{n}{<->ssub * ptmnocm/m/n}{}
```

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

```
442 \DeclareFontFamily{OML}{ptmcm}{\skewchar \font =127}
```

```

443 \DeclareFontShape{OML}{ptmcm}{m}{it}{
444   <-> zptmnocmrm
445 }{}
446 \DeclareFontShape{OML}{ptmcm}{l}{it}{<->ssub * ptmcm/m/it}{}
447 \DeclareFontShape{OML}{ptmcm}{m}{sl}{<->ssub * ptmcm/m/it}{}
448 \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.

```

449 \DeclareFontFamily{OMS}{pzccm}{}
450 \DeclareFontShape{OMS}{pzccm}{m}{n}{
451   <-> zpzcnocmry
452 }{}% cmsy10 Symbol Zapf Chancery Medium-Italic Times-Roman
453 \DeclareFontShape{OMS}{pzccm}{l}{n}{<->ssub * pzccm/m/n}{}

```

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

```

454 \DeclareFontFamily{OMX}{psycm}{}
455 \DeclareFontShape{OMX}{psycm}{m}{n}{
456   <-> zpsynocmrv
457 }{}
458 \DeclareFontShape{OMX}{psycm}{l}{n}{<->ssub * psycm/m/n}{}
459 %
460 \DeclareFontEncoding{8r}{}{}% from file: 8renc.def
461 \DeclareFontFamily{8r}{cmr}{\hyphenchar\font45 }% from file: 8rcmr.fd
462 \DeclareFontShape{8r}{cmr}{m}{n}{
463   <-> cmr10
464 }{}

```

6.3 Font definition files

The following forces \LaTeX 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.

```

465 \input{8rphv.fd}%
466 \input{8rptm.fd}%
467 \input{ot1phv.fd}%
468 \input{ot1ptm.fd}%
469 \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.

```

470 \def\eightRChar#1{\def\encodingdefault{8r}\fontencoding\encodingdefault\selectfont\char"#1

```

```

471 \def\ANGLE{<>}\eightRChar{8B}%
472 \def\RANGLE{>>}\eightRChar{9B}%
473 %\def\ASTER{\eightRChar{2A}}%
474 %\def\DAGGER{\eightRChar{86}}%
475 %\def\DDAGGER{\eightRChar{87}}%
476 \def\BULLET{\eightRChar{95}}%
477 %\def\SECTION{\eightRChar{A7}}%
478 %\def\PARAGRAPH{\eightRChar{B6}}%
479 \def\VERTBAR{\eightRChar{7C}}%
480 \def\COPYRIGHT{\eightRChar{A9}}%
481 \def\REGISTERED{\eightRChar{AE}}%

482 \def\textbar{\VERTBAR}%
483 \def\textbullet{\BULLET}%
484 \def\textcopyright{\COPYRIGHT}%
485 \def\textregistered{\REGISTERED}%

```

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

```

486 \def\@makefnmark{\@thefnmark}%
487 \def\@fnsymbol#1{{\ifcase#1\or *\or \dagger\or \ddagger\or
488 \textsection\or \textparagraph\or |\or **\or \dagger\dagger
489 \or \ddagger\ddagger \else\@ctrerr\fi}}

```

6.5 End of the fonts DOCSTRIP module

Here ends the module.

```
490 %</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 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 `\\`.

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 separations 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 didn't already experience it).

7.3 Conventions For L^AT_EX

Team L^AT_EX 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^AT_EX itself fails to comply with many of the coding recommendations of Team L^AT_EX.

Here ends the programmer's documentation.

Index

Symbols		A	
<code>\,</code>	201	<code>\abovedisplayskip</code>	346
<code>\-</code>	201	acrofont document class	2, 9, 11, 18
<code>.dtx</code>	4–6	acrofont.sty	1, 4, 5
<code>.ins</code>	4	<code>\active</code>	175, 176, 183, 184, 266, 277, 280, 283
<code>\/</code>	215, 266, 280	<code>\active@colon</code>	277, 286, 287
<code>\:</code>	266, 277	<code>\active@dot</code>	283, 286, 287
<code>\<</code>	175, 183, 188	<code>\active@slash</code>	280, 286, 287
<code>\></code>	176, 184, 189	<code>\aftergroup</code>	277, 280, 283, 353
<code>\@Hline</code>	348	<code>\allowbreak</code>	278, 281, 284
<code>\@cmd</code>	231, 232	<code>\arg</code>	10
<code>\@ctrerr</code>	489	<code>\arg</code>	219
<code>\@currsize</code>	168	argument	
<code>\@endptrue</code>	354	foo	10
<code>\@fnsymbol</code>	487	name	10
<code>\@gt</code>	186, 189	text	10
<code>\@gtr@err</code>	186, 191	<code>\arraycolsep</code>	427
<code>\@idxitem</code>	309	article document class	18
<code>\@ifnextchar</code>	185, 186	<code>\ASTER</code>	473
<code>\@ifx</code>	22	<code>\AtBeginDocument</code>	179, 267
<code>\@latex@warning@no@line</code>	418	<code>\AtEndDocument</code>	404
<code>\@lt</code>	185, 188	<code>\author</code>	76
<code>\@mainmatterfalse</code>	323, 337	B	
<code>\@mainmattertrue</code>	328	<code>\backmatter</code>	9
<code>\@makefnmark</code>	486	<code>\backmatter</code>	331
<code>\@meta</code>	185, 187	<code>\begin</code>	10
<code>\@openrighttrue</code>	320	<code>\belowdisplayskip</code>	352
<code>\@restonecolfalse</code>	290	book document class	9, 18
<code>\@restonecoltrue</code>	292	book.cls document class	15
<code>\@startsection</code>	360	<code>\botrule</code>	10
<code>\@tempa</code>	208, 209, 272–274	<code>\botrule</code>	358
<code>\@tempb</code>	204, 209	<code>\BULLET</code>	476, 483
<code>\@thefnmark</code>	486	C	
<code>\@to</code>	342, 347	<code>\c@secnumdepth</code>	301, 302, 306
<code>\@undefined</code>	417	<code>\catcode</code>	175, 176, 183, 184, 266, 277, 280, 283
<code>\@unused</code>	423	<code>\char</code>	188, 189, 237, 241, 470
<code>\@url</code>	266, 270, 299	<code>\class@base</code>	429
<code>\‘</code>	201	<code>\class@name</code>	158, 159, 430, 431
<code>\ </code>	180, 488		
00readme.txt	3, 4		
<code>_</code>	90, 119, 125, 130, 148, 149, 372		

`\ClassError` 192
`\ClassInfo` 431
`\classname` 10
`\classname` . 67, 82, 98, 107, 108,
142, 261
`\classoption` 10
`\classoption` 260
`\cleardoublepage` 340
`\clearpage` 310, 335
`\cleartorecto` 322, 327, 333, 339,
340
`\closeout` 415
`clsguide.tex` 22
`\cmd` 10
`\cmd` 231, 239, 247, 248, 298
`\cmd@to@cs` ... 234, 237, 243, 298
`\cmd@to@index` 235, 238
`cmsy` 19
`\colon@break` 278, 286
`\colon@char` 278, 279, 287
`\colrule` 10
`\colrule` 357
`\columnsep` 295
`\columnseprule` 294
`\COPYRIGHT` 480, 484
`\cs` 234, 241, 243, 298
`\csname` 10
`\csname` 208, 425

D

`\DAGGER` 474
`\dagger` 487, 488
`\DDAGGER` 475
`\ddagger` 487, 489
`\DeclareFontEncoding` 460
`\DeclareFontFamily` .. 437, 442,
449, 454, 461
`\DeclareFontShape` 438, 441, 443,
446–448, 450, 453, 455, 458,
462
`\DeclareRobustCommand` 210, 214,
219, 224, 231, 241, 250, 251,
254, 255, 259–261, 266, 321,
326, 331, 359
`\def` 20, 22
`\do` 201

`doc` 4, 6
`\DocInput` 8
document class
 `acronym` 2, 9, 11, 18
 `article` 18
 `book` 9, 18
 `book.cls` 15
 `ltxdoc` . 2, 5, 8, 9, 11, 17, 21
 `ltxdoc.cls` 2, 15
 `ltxdocext.sty` 11
 `ltxguide.cls` 2, 11
 `ltxutil` 1
 `ltxutil.sty` 3
 `report` 18
document environment 4
`\DoNotIndex` 11
`\dot@break` 284, 286
`\dot@char` 284, 285, 287
`\dump` 19

E

`\edef` 208, 272
`\eightRChar` 470–481
`\encodingdefault` 470
`\end` 10
`\endbatchfile` 63
`\endcsname` 208, 425
`\endfilecontents` 406, 425
`\endfoo` 22
`\endpreamble` 40
`\ensuremath` 20
`\env` 10
`\env` 250, 252, 254
`\envb` 10
`\envb` 251
`\enve` 10
`\enve` 254
environment
 `document` 4
 `filecontents` 2, 4, 17
 `minipage` 22
 `quote` 11
 `theindex` 14
 `unnumtable` 10
 `verbatim` 2, 9, 12

environments:	
unnumtable	342
extensions	2, 11, 18
F	
\fi	21, 22
file	
.dtx	4–6
.ins	4
00readme.txt	3, 4
acrofont.sty	1, 4, 5
clsguide.tex	22
cmsy	19
doc	4, 6
extensions	2, 11, 18
fonts	2, 18, 20
ltxdoc.cfg	2, 16
ltxdocext.drv	5
ltxdocext.ins	5
ltxdocext.sty	1, 4, 5
ltxguide.cls	12
ltxutil.dtx	2, 3
ltxutil.ins	3
ltxutil.pdf	1
ltxutil.sty	1–3
makeindex	3
mathptm.sty	18
omlptmcm.fd	18
omspzccm.fd	19
omxpsycm.fd	19
ot1ptmcm.fd	18
times.sty	18
zpsycmrv	19
zpsynocmrv	19
zptmcmr	18
zptmcmrm	18
zptmnocmr	18
zptmnocmrm	18
zpzccmry	19
zpzcnocmry	19
\file	10
\file	43–45, 95, 100–102, 105, 106, 114, 115, 124, 130, 133, 134, 136–138, 147, 148, 255
filecontents environment	2, 4, 17
\filedate	9, 17
\fileinfo	9
\fileinfo	207
\fileversion	9, 17
\font	442, 461
\fontencoding	470
fonts	2, 18, 20
\foo	22
foo, argument	10
\from	43–45
\frontmatter	9
\frontmatter	321
G	
\generate	42
\GetFileInfo	9, 17
\GetFileInfo	22, 202
H	
\hbox	22
\hline	356–358
\hyphenchar	461
I	
\I	422
\if	21, 22
\if@mainmatter	318
\if@openright	319, 332
\if@restonecol	310
\if@twocolumn	289
\iffalse	408
\ifToplevel	47
\ignorespaces	347, 353
\immediate	407, 415, 423
\index ...	212, 221, 228, 229, 239, 247, 248, 257, 264, 273, 274
\indexname	303–305
\item .	128, 135, 141, 145, 309, 316
\itshape	215, 362
K	
\keepsilent	41
L	
\L	421
\LANGLE	215, 217, 471
\langle	217

<code>\LaTeX</code>	96, 116, 119, 130, 144, 148, 149	<code>\newenvironment</code>	22
<code>\LaTeXe</code>	90, 125, 160, 432	<code>\newif</code>	318, 319
<code>\leftmargin</code>	313, 314	<code>\nopagebreak</code>	167
<code>ltxdoc</code> document class	2, 5, 8, 9, 11, 17, 21	<code>\normalfont</code>	174, 362
<code>ltxdoc.cfg</code>	2, 16	<code>\normalsize</code>	168, 362
<code>ltxdoc.cls</code> document class	2, 15	O	
<code>ltxdocext.drv</code>	5	<code>\o@verbatim</code>	164, 171
<code>ltxdocext.ins</code>	5	<code>\oarg</code>	10
<code>ltxdocext.sty</code>	1, 4, 5	<code>\oarg</code>	223, 224
<code>ltxdocext.sty</code> document class	11	<code>omlptmcm.fd</code>	18
<code>ltxguide.cls</code>	12	<code>omspzccm.fd</code>	19
<code>ltxguide.cls</code> document class	2, 11	<code>omxpsycm.fd</code>	19
<code>ltxutil</code> document class	1	<code>\onecolumn</code>	310
<code>ltxutil.dtx</code>	2, 3	<code>otiptmcm.fd</code>	18
<code>ltxutil.ins</code>	3	P	
<code>ltxutil.pdf</code>	1	<code>\PackageInfo</code>	159
<code>ltxutil.sty</code>	1–3	<code>\pagenumbering</code>	324, 329
<code>ltxutil.sty</code> document class	3	<code>\PARAGRAPH</code>	478
M			
<code>\m</code>	10	<code>\parbox</code>	22
<code>\m</code>	187, 210	<code>\parindent</code>	307
<code>\mainmatter</code>	9	<code>\parskip</code>	308
<code>\mainmatter</code>	326	placeholder	
<code>\makeatletter</code>	231, 405	<code>\langle table head rows \rangle</code>	10
<code>\makeatother</code>	426	<code>\langle table rows \rangle</code>	10
<code>makeindex</code>	3	<code>\postdisplaypenalty</code>	351
<code>\MakeShortVerb</code>	180	<code>\preamble</code>	27
<code>\maketitle</code>	80	<code>\predisplaypenalty</code>	345
<code>\MakeUppercase</code>	304, 305	<code>\prepdef</code>	21
<code>\mathchardef</code>	301	<code>\ProvidesClass</code>	9
<code>mathptm.sty</code>	18	<code>\ProvidesFile</code>	9
<code>\mathsection</code>	20	<code>\ProvidesFile</code>	4, 5, 7
<code>\mbox</code>	22	Q	
<code>\mbox</code>	215	quote environment	11
<code>\MessageBreak</code>	196	R	
<code>\meta</code>	211, 212, 214, 220, 226	<code>\RANGLE</code>	215, 218, 472
<code>minipage</code> environment	22	<code>\rangle</code>	218
<code>\Msg</code>	48–54, 56, 57, 59–61	<code>\REGISTERED</code>	481, 485
N			
name, argument	10	<code>\renewenvironment</code>	288, 311
<code>\name@idx</code>	250, 259–262	report document class	18
<code>\newcommand</code>	20, 22	<code>\RequirePackage</code>	18, 19, 163, 178, 435, 436
		<code>\reserved@</code>	407, 415

S	
<code>\save@secnumdepth</code>	301, 306
<code>\sbox</code>	22
<code>\sc</code>	116, 139
<code>\scmd</code>	242
<code>\scmd@to@index</code>	244, 246
<code>\SECTION</code>	477
<code>\section</code>	112, 303
<code>\see</code>	296
<code>\seealso</code>	297
<code>\setbox</code>	22
<code>\setlength</code>	22
<code>\skewchar</code>	442
<code>\slash@break</code>	281, 286
<code>\slash@char</code>	281, 282, 287
<code>\small</code>	169
<code>\StopEventually</code>	7
<code>\string</code>	212, 220, 221, 228, 229, 234, 235, 239, 243, 244, 247, 248, 252, 254, 257, 264, 272, 298, 408, 413
<code>\subsection</code>	122
<code>\substyle</code>	10
<code>\substyle</code>	259
<code>\subsection</code>	10, 16
<code>\subsubsection</code>	359
T	
<code>\T</code>	416, 421, 422
<code><table head rows></code> placeholder	10
<code><table rows></code> placeholder	10
<code>\tableofcontents</code>	110
<code>\TeX</code>	90
<code>text</code> , argument	10
<code>\textbar</code>	482
<code>\textbullet</code>	483
<code>\textcopyright</code>	484
<code>\textit</code>	296, 297
<code>\textparagraph</code>	488
<code>\textregistered</code>	485
<code>\textsection</code>	20
<code>\textsection</code>	488
<code>\texttt</code>	77
<code>\thanks</code>	68
<code>theindex</code> environment	14
<code>\thispagestyle</code>	307
<code>times.sty</code>	18
<code>\title</code>	66
<code>\toprule</code>	10, 16
<code>\toprule</code>	356
<code>\ttfamily</code>	174, 220, 225, 226, 241, 252, 254, 256, 263, 271, 300
<code>\ttt</code>	221, 228, 229, 257, 264, 299, 300
<code>\twocolumn</code>	303
U	
<code>unnumtable</code> (environment)	342
<code>unnumtable</code> environment	10
<code>\unskip</code>	166, 350
<code>\url</code>	10
<code>\url</code>	86, 268, 272
<code>\url@break</code>	271, 286, 299
<code>\url@char</code>	272, 287
<code>\url@ltxdocext</code>	266, 268
V	
<code>\vbox</code>	22
<code>\verb</code>	9, 12
<code>\verbatim</code>	164, 165
<code>verbatim</code> environment	2, 9, 12
<code>\verbatim@font</code>	173
<code>\verbatim@nolig@list</code>	201
<code>\VERTBAR</code>	479, 482
W	
<code>\write</code>	407, 423
Z	
<code>zpsycmrv</code>	19
<code>zpsynocmrv</code>	19
<code>zptmcmr</code>	18
<code>zptmcmrm</code>	18
<code>zptmnocmr</code>	18
<code>zptmnocmrm</code>	18
<code>zpzccmry</code>	19
<code>zpzcnocmry</code>	19