

A L^AT_EX Package of utility macros ^{*†}

Arthur Ogawa [‡]

February 25, 2010

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

The distribution point for this work is publish.aps.org/revtex, which contains prebuilt runtime files, documentation, and full source, ready to add to a TDS-compliant T_EX installation.

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 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 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	Change Log	3
1.3	Bill of Materials	3
1.3.1	Primary Source	3
1.3.2	Generated by <code>latex ltxutil.dtx</code>	4
1.3.3	Generated by <code>tex ltxutil.dtx</code>	4
1.3.4	Auxiliary	4
2	Code common to all modules	4

^{*}This file has version number 4.1p, last revised 2010/02/25/18:00:00.

[†]Version 4.1p © 2009 The American Physical Society

[‡]mailto:arthur_ogawa@sbcglobal.net

3	The driver module <code>doc</code>	5
3.1	The Preamble	5
3.1.1	Docstrip and info directives	5
3.2	The “Read Me” File	6
3.3	The Document Body	9
4	Using this package	9
4.1	Invoking the package	9
5	Compatibility with L^AT_EX’s Required Packages	10
5.1	<code>array</code>	10
5.2	<code>longtable</code>	10
6	Implementation of package	10
6.1	Beginning of the <code>package</code> <code>DOCSTRIP</code> module	10
6.2	Banner and beginning of the <code>kernel</code> <code>DOCSTRIP</code> module	10
6.3	Errors and warnings	11
6.4	New Tools	11
6.5	Boolean Control	13
6.6	Begin Document Structure	15
6.7	Class Extensions	17
6.8	Type Tools	19
6.9	Display Math	20
6.10	Footnotes	22
6.11	Floats	27
6.11.1	Usage notes	27
6.11.2	Robustifying fragile commands	28
6.11.3	Preparing for the <code>hyperref</code> package	29
6.11.4	Footnotes within floats, unfloating floats, float font	30
6.11.5	Writing floats out to a file	32
6.12	Counters	36
6.13	Customization of Sections	36
6.14	Patch the <code>tabular</code> and <code>array</code> Environments	40
6.15	Repair other broken parts of L ^A T _E X	62
6.16	Syntax	63
6.17	Auto-indented Contents	63
6.18	Lists	66
6.19	Hypertext capabilities	67
6.20	End of the <code>kernel</code> <code>DOCSTRIP</code> module	70
	Index	71

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.dtx` (Note: do *not* use L^AT_EX for this

task). The typeset documentation that you are now reading is generated from the same file by typesetting it with \LaTeX 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 $\LaTeX 2_\epsilon$ (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 file `00readme`.

Note: you will have to run \LaTeX , then `makeindex -s gind.ist ltxutil.idx`, then `makeindex -s gglo.ist -o ltxutil.gls ltxutil.glo`, then \LaTeX again in order to obtain a valid index and table of contents.

2. Now typeset `ltxutil.dtx` with \TeX (not \LaTeX), thereby generating the package file `ltxutil.sty`.
3. Install the following files into indicated locations within your TDS-compliant `texmf` tree (you may need root access):

- `$TEXMF/tex/latex/revtex/ltxutil.sty`
- `$TEXMF/source/latex/revtex/ltxutil.dtx`
- `$TEXMF/doc/latex/revtex/ltxutil.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 Change Log

1.3 Bill of Materials

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

1.3.1 Primary Source

One single file generates all.

```
%ltxutil.dtx
%
```

1.3.2 Generated by latex ltxutil.dtx

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

```
%00readme ltxutil.pdf
%
```

1.3.3 Generated by tex ltxutil.dtx

Typesetting this file with T_EX generates the package file.

```
%ltxutil.sty
%
```

1.3.4 Auxiliary

The following are auxiliary files generated in the course of running L^AT_EX:

```
%ltxutil.aux ltxutil.idx ltxutil.ind ltxutil.log ltxutil.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^AT_EX 2_ε. An appropriate message is displayed if a different T_EX format is used.

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

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

```
4 %<kernel&!package&!doc>\typeout{%
5 %<*package|doc>
6 \ProvidesFile{%
7 %</package|doc>
8 %<*kernel|package|doc>
9 ltxutil%
10 %</kernel|package|doc>
11 %<*doc>
12 .dtx%
13 %</doc>
14 %<package>.sty%
15 %<*package|doc>
16 }%
17 %</package|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. Give credit where due.

```
18 %<*doc|package|kernel>
19 %<version>
20 [2010/02/25/18:00:00 4.1p utilities package (portions licensed from W. E. Baxter web at supers
21 %</doc|package|kernel>
22 %<kernel&!package&!doc>}%
```

3 The driver module doc

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

```
23 %<*doc>
```

3.1 The Preamble

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

```
24 \documentclass{ltxdoc}
25 \RequirePackage{ltxdocext}%
26 \let\url\undefined
27 \RequirePackage[colorlinks=true,linkcolor=blue]{hyperref}%
28 \expandafter\ifx\csname package@font\endcsname\@undefined\else
29 \expandafter\RequirePackage\expandafter{\csname package@font\endcsname}%
30 \fi
31 \CodelineIndex\EnableCrossrefs % makeindex -s gind.ist ltxutil
32 \RecordChanges % makeindex -s gglo.ist -o ltxutil.gls ltxutil.glo
```

3.1.1 Docstrip and info directives

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

```
33 \setcounter{StandardModuleDepth}{1}
```

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

```
34 \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 out these procedures begin the programmer’s manual.

```
35 \begin{filecontents*}{00readme}
36 \title{%
37 A \LaTeX\ Package of utility macros%
38 \thanks{%
39 This file has version number \fileversion,
40 last revised \filedate.%
41 }%
42 \thanks{%
43 Version \fileversion\ \copyright\ 2009 The American Physical Society
44 }%
45 }%
46 \author{%
47 Arthur Ogawa%
48 \thanks{\texttt{mailto:arthur\_ogawa at sbcglobal.net}}}%
49 }%
50 %\iffalse
51 % For version number and date,
52 % search on "\fileversion" in the .dtx file,
53 % or see the end of the 00readme file.
54 %\fi
55 \maketitle
56
57 This file embodies the \classname{ltxutil} package,
58 the implementation and its user documentation.
59
60 The distribution point for this work is
61 \url{publish.aps.org/revtex},
62 which contains prebuilt runtime files, documentation, and full source,
63 ready to add to a TDS-compliant \TeX\ installation.
64
65 The \classname{ltxutil} package was commissioned by the American Physical Society
66 and is distributed under the terms of the \LaTeX\ Project Public License,
67 the same license under which all the portions of \LaTeX\ itself are distributed.
68 Please see \url{http://ctan.tug.org/macros/latex/base/lppl.txt} for details.
69
70 To use this document class, you must have a working
71 \TeX\ installation equipped with \LaTeXe\
```

72 and possibly pdftex and Adobe Acrobat Reader or equivalent.
73
74 To install, retrieve the distribution,
75 unpack it into a directory on the target computer,
76 and move the file `\file{ltxutil.sty}`
77 into a location in your filesystem where it will be found by `\LaTeX`.
78
79 To use, read the user documentation `\file{ltxutil.pdf}`.
80
81 `\tableofcontents`
82
83 `\section{Processing Instructions}`
84
85 The package file `\file{ltxutil.sty}`
86 is generated from this file, `\file{ltxutil.dtx}`,
87 using the `{\sc docstrip}` facility of `\LaTeX`
88 via `|tex ltxutil.dtx|` (Note: do `\emph{not}` use `\LaTeX` for this task).
89 The typeset documentation that you are now reading is generated from
90 the same file by typesetting it with `\LaTeX` or `pdftex`
91 via `|latex ltxutil.dtx|` or `|pdflatex ltxutil.dtx|`.
92
93 `\subsection{Build Instructions}`
94
95 You may bootstrap this suite of files solely from `\file{ltxutil.dtx}`.
96 Prepare by installing `\LaTeXe` (and either `tex` or `pdftex`) on your computer,
97 then carry out the following steps:
98 `\begin{enumerate}`
99 `\item`
100 Within an otherwise empty directory,
101 typeset `\file{ltxutil.dtx}` with `\LaTeX` or `pdflatex`;
102 you will obtain the typeset documentation you are now reading,
103 along with the file `\file{00readme}`.
104
105 Note: you will have to run `\LaTeX`, then
106 `\file{makeindex} \texttt{-s gind.ist ltxutil.idx}`, then
107 `\file{makeindex} \texttt{-s gglo.ist -o ltxutil.gls ltxutil.glo}`, then
108 `\LaTeX` again in order to obtain a valid index and table of contents.
109 `\item`
110 Now typeset `\file{ltxutil.dtx}` with `\TeX` (not `\LaTeX`),
111 thereby generating the package file `\file{ltxutil.sty}`.
112 `\item`
113 Install the following files into indicated locations within your
114 TDS-compliant `\texttt{texmf}` tree (you may need root access):
115 `\begin{itemize}`
116 `\item`
117 `\file{\$TEXMF/}\file{tex/}\file{latex/}\file{revtex/}\classname{ltxutil.sty}`
118 `\item`
119 `\file{\$TEXMF/}\file{source/}\file{latex/}\file{revtex/}\classname{ltxutil.dtx}`
120 `\item`
121 `\file{\$TEXMF/}\file{doc/}\file{latex/}\file{revtex/}\classname{ltxutil.pdf}`

```

122 \end{itemize}
123 where \file{$TEXMF/} stands for \file{texmf-local/}, or some other \texttt{texmf} tree
124 in your installation.
125 \item
126 Run \texttt{mktexlsr} on \file{$TEXMF/} (you may need root access).
127 \item
128 Build and installation are now complete;
129 now put a \cmd\usepackage\texttt{\{ltxutil\}} in your document preamble!
130 \end{enumerate}
131
132 \subsection{Change Log}
133 \changes{4.0b}{1999/06/20}{AO: Fixed spurious \texttt{CR} and (return) characters in output fil
134 \changes{4.0b}{1999/06/20}{AO: Removed superfluous \cs{def}s, changed to using \cs{floats@sw} a
135 \changes{4.0b}{1999/06/20}{only execute if there really were floats of the given type}
136 \changes{4.0b}{1999/06/20}{Support the hack with \cs{prepdef}, and delay until \cs{AtBeginDocum
137 \changes{4.0c}{1999/11/13}{(AO, 110) Install hooks for endfloats processing}
138 \changes{4.0c}{1999/11/13}{(AO, 116) Hyperref compatibility}
139 \changes{4.0c}{1999/11/13}{(AO, 130) Interference from array package}
140 \changes{4.0c}{1999/11/13}{*-form mandates pagebreak at each float; only print section head if
141 \changes{4.0d}{2000/04/10}{(AO, 127) Floats placed [h] to allow page breaks}
142 \changes{4.0d}{2000/04/10}{(AO, 174) kernel fix}
143 \changes{4.0d}{2000/05/19}{(AO, 224) Hyperref compatibility.}
144 \changes{4.0d}{2000/05/23}{Allow things to break over pages by setting array@default.}
145 \changes{4.0e}{2000/11/16}{(AO, 221) Remove samepage command from @xfloat@prep: If the float ca
146 \changes{4.0f}{2001/07/13}{(AO, 404) Hyperref compatibility}
147 \changes{4.1a}{2008/01/19}{(AO, 459) do not assume \cs{classname} is defined}%
148 \changes{4.1a}{2008/01/19}{(AO, 461) Change the csname from \cs{@dotsep} to \cs{ltxu@dotsep}. T
149 \changes{4.1a}{2008/01/19}{(AO, 475) I had not properly reproduced the LaTeX macro \cs{eqnarray
150 \changes{4.1a}{2008/01/19}{(AO, 479) Per: Dylan Thurston<dpt at math.harvard.edu>}%
151 \changes{4.1a}{2008/06/30}{(AO) Make \cs{addtocontents} a \cs{long} \cs{def}; gobble up \cs{foo
152 \changes{4.1a}{2008/06/30}{(AO) Remove code that avoided changes to \cs{@xfootnotemark}}%
153 \changes{4.1a}{2008/06/30}{(AO, 438) Complete rewrite of footnote macros.}
154 \changes{4.1a}{2008/07/07}{\cs{@xfloat@prep} calls \cs{ltxu@footnote@pop} to restore the origina
155 \changes{4.1a}{2008/08/12}{\cs{class@documenthook} is the last \cs{AtBeginDocument} token now}
156 \changes{4.1a}{2008/08/12}{Class extension mechanism \cs{@pushfilename@ltx} and \cs{@p@pfilenam
157 \changes{4.1a}{2008/08/12}{Class extension mechanism \cs{class@extension}, \cs{class@extensionf
158 \changes{4.1a}{2008/08/12}{Get rid of \cs{set@typesize@hook} \cs{set@pica@hook} and the \cs{nor
159 \changes{4.1b}{2008/08/12}{(AO, 487) Support for video figures and the \cs{setfloatlink} comman
160 \changes{4.1b}{2008/08/12}{(AO, 505) try to accommodate \classname{colortbl}.}
161 \changes{4.1b}{2008/08/12}{Acquire \classname{hyperref} saveoire}
162 \changes{4.1b}{2008/08/12}{Default assignment of \cs{float@sw} now, not at \cs{AtBeginDocument}
163 \changes{4.1b}{2008/08/12}{If class option \classoption{lengthcheck} is in effect, log the heig
164 \changes{4.1b}{2008/08/12}{No need to protect against undefined \cs{float@sw}}
165 \changes{4.1b}{2008/08/12}{Patch the array package even later: after all package patches go in.
166 \changes{4.1b}{2008/08/12}{Refine toc processing: provide default.}%
167 \changes{4.1b}{2008/08/12}{Tally and log the height of a float class}
168 \changes{4.1d}{2009/03/27}{(AO, 511) Compatability with lineno.sty's erroneous way of detecting
169 \changes{4.1f}{2009/07/07}{(AO, 515) Hook for setting the font of a footnote}
170 \changes{4.1f}{2009/07/10}{(AO, 518) Tally register overflow when locument is long}
171 \changes{4.1g}{2009/10/06}{(AO, 532) Both arguments of \cs{href} get sanitized}%

```



```

172 \changes{4.1g}{2009/10/07}{(AO, 525) Remove phantom paragraph above display math that is given
173 \changes{4.1g}{2009/10/07}{(AO, 539) Use of double-backslash in argument of \cs{section} gives
174 \changes{4.1n}{2009/12/05}{(AO, 569) Use of \classname{hyperref} interferes with column balanci
175 \changes{4.1n}{2009/12/06}{(AO) Incorporate change to ltmiscen.dtx v1.1i 2000/05/19}%
176 \changes{4.1n}{2009/12/09}{(AO, 569) execute \classname{atveryend}'s \cs{Call@AfterLastShipout}
177 \changes{4.1n}{2009/12/13}{(AO, 574) protect against \classname{lineno.sty}, which forces a vis
178 \changes{4.1n}{2010/01/02}{(AO, 571) Interface \cs{set@footnotewidth} for determining the set w
179 \changes{4.1n}{2010/01/02}{(AO, 571) allow split after last line of footnote}%
180 \changes{4.1n}{2010/01/06}{(AO, 572) title block footnotes numbered independently from body foo
181 \changes{4.1p}{2010/02/24}{(AO, 582) A patch of \classname{hyperref.sty} to provide backward co
182
183
184 \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.

```

185 \begin{document}%
186 \expandafter\DocInput\expandafter{\jobname.dtx}%
187 \end{document}
188 %</doc>

```

4 Using this package

Once this package is installed on your filesystem, you can employ it in adding functionality to \LaTeX by invoking it in your document or document class.

4.1 Invoking the package

In your document, you can simply call it up in your preamble:

```

%\documentclass{book}%
%\usepackage{ltxutil}%
%\begin{document}
%your document here
%\end{document}

```

However, the preferred way is to invoke this package from within your customized document class:

```

%\NeedsTeXFormat{LaTeX2e}[1995/12/01]%
%\ProvidesClass{myclass}%
%\RequirePackage{ltxutil}%
%\LoadClass{book}%
%class customization commands
%\endinput

```

Once loaded, the package gives you access to certain procedures, usually to be invoked by a \LaTeX command or environment, but not at the document level.

5 Compatibility with L^AT_EX’s Required Packages

Certain packages, usually ones written by members of the L^AT_EX Project itself, have been designated “required” and are distributed as part of standard L^AT_EX. These packages have been placed in a privileged position vis á vis the L^AT_EX kernel in that they override the definitions of certain kernel macros.

The `ltxutil` package will be incompatible with any package that redefines any of the kernel macros that `ltxutil` patches—if that package is loaded *after* `ltxutil`. This means that for greatest compatibility, `ltxutil` should be loaded *after*, say, `ftnright`, which overwrites L^AT_EX’s kernel procedures `\@outputdblcol`, `\@startcolumn`, and `\@makecol`.

Hereinafter follows some notes on specific L^AT_EX packages.

5.1 array

This package alters the way tabular environments are done, therefore it could run afoul of the L^AT_EX “required” package `array` or any package that calls for it to be loaded. However, this package has provisions for remaining compatible with `array`. So long as the version of `array` that is used with this package has the appropriate meanings for the procedures it overwrites, all should be well.

5.2 longtable

David Carlisle’s `longtable` package modifies both the L^AT_EX kernel and the `array` package. This package must therefore alter `\LT@array`. For now, that job is handled by `ltxgrid`.

6 Implementation of package

Special acknowledgment: this package uses concepts pioneered and first realized by William Baxter (<mailto:web@superscript.com>) in his SuperScript line of commercial typesetting tools, and which are used here with his permission.

6.1 Beginning of the package DOCSTRIP module

```
189 %<*package>
190 \def\package@name{ltxutil}%
191 \expandafter\PackageInfo\expandafter{\package@name}{%
192 Utility macros for \protect\LaTeXe,
193 by A. Ogawa (arthur_ogawa at sbcglobal.net)%
194 }%
195 %</package>
```

6.2 Banner and beginning of the kernel DOCSTRIP module

```
196 %<*kernel>
```

6.3 Errors and warnings

```
\class@err A few shorthands for Class messages. Your document class should define
\class@warn \class@name.
\class@info 197 \def\class@err#1{\ClassError{\class@name}{#1}\@eha}%
198 \def\class@warn#1{\ClassWarningNoLine{\class@name}{#1}}%
199 \def\class@info#1{\ClassInfo{\class@name}{#1}}%
200 \def\obsolete@command#1{%
201 \class@warn@end{Command \string#1\space is obsolete.^^JPlease remove from your document}%
202 \global\let#1\@empty
203 #1%
204 }%
205 \def\replace@command#1#2{%
206 \class@warn@end{Command \string#1\space is obsolete;^^JUse \string#2\space instead}%
207 \global\let#1#2%
208 #1%
209 }%
210 \def\replace@environment#1#2{%
211 \class@warn@end{Environment #1 is obsolete;^^JUse #2 instead}%
212 \glet@environment{#1}{#2}%
213 \@nameuse{#1}%
214 }%
215 \def\incompatible@package#1{%
216 \@ifpackageloaded{#1}{%
217 \def\@tempa[I cannot continue. You must remove the \string\usepackage\ statement that caused
218 \ClassError{\class@name}{The #1 package cannot be used with \class@name}%
219 \@tempa\stop
220 }{%
221 \class@info{#1 was not loaded (OK!)}%
222 }%
223 }%
224 \def\class@warn@end#1{%
225 \gapdef\class@enddocumenthook{\class@warn{#1}}%
226 }%

Give \class@name a meaning if it does not already have one.
227 \ifx\undefined\class@name
228 \def\class@name{ltxutil}%
229 \class@warn{You should define the class name before reading in this package. Using default}%
230 \fi
```

6.4 New Tools

```
\t@
231 \def\t@{to}%

\dimen@iii
232 \dimendef\dimen@iii\thr@@
```

```

\halign@
233 \def\halign@{\halign\t@}%

\fur Analogous to \@ne, \tw@, and \thr@.
234 \chardef\fur=4\relax
235 \chardef\cat@letter=11\relax
236 \chardef\other=12\relax

\let@environment The directive \let@environment takes care of a common programming idiom
\glet@environment whereby one environment is made a synonym for another.
237 \def\let@environment#1#2{%
238 \expandafter\let
239 \csname#1\expandafter\endcsname\csname#2\endcsname
240 \expandafter\let
241 \csname end#1\expandafter\endcsname\csname end#2\endcsname
242 }%
243 \def\glet@environment#1#2{%
244 \global\expandafter\let
245 \csname#1\expandafter\endcsname\csname#2\endcsname
246 \global\expandafter\let
247 \csname end#1\expandafter\endcsname\csname end#2\endcsname
248 }%

\tracingplain The command \tracingplain causes TEX's tracing parameters to return to the
values set by default. This command is sometimes useful when you have said
\tracingall somewhere and want to restore. The \traceoutput command
causes \tracingoutput diagnostics upon \shipout.
249 \newcommand\tracingplain{%
250 \tracingonline\z@\tracingcommands\z@\tracingstats\z@
251 \tracingpages\z@\tracingoutput\z@\tracinglostchars\@ne
252 \tracingmacros\z@\tracingparagraphs\z@\tracingrestores\z@
253 \showboxbreadth5\showboxdepth3\relax %\errorstopmode
254 }%
255 \newcommand\traceoutput{%
256 \appdef\@resetactivechars{\showoutput}%
257 }%

\say The commands \say and \saythe cause diagnostic messages in the TEX log that
\saythe give the value of a control sequence name or a register respectively.
258 \newcommand\say[1]{\typeout{<\noexpand#1=\meaning#1>}}%
259 \newcommand\saythe[1]{\typeout{<\noexpand#1=\the#1>}}%

\fullinterlineskip Resets the \prevdepth so that the full amount of \baselineskip glue will be
inserted by the \baselineskip mechanism. Can be invoked just after a \hrule
to undo its default suppression of base line skip.
260 \def\fullinterlineskip{\prevdepth\z@}%

\count@i
\count@ii 261 \countdef\count@i\@ne
262 \countdef\count@ii\tw@

```

6.5 Boolean Control

We introduce just enough of the Boolean calculus for \TeX . Alan Jeffrey was the pioneer here, with an article in TUGboat (Vol. 11, No. 2, page 237). This implementation owes a debt to William Baxter (web at superscript.com). See articles by Baxter and Ogawa in the proceedings of the 1994 TUG meeting, TUGboat Vol. 15, No. 3.

```

\prepdef Provide the capability of performing head- and tail patches. The procedure
\appdef \prepdef prepends to the given macro the tokens specified in its second argument.
\gappdef Likewise for \appdef, except that it appends. Note that the first 10 toks registers
are utility registers, and we simply make a control sequence name, \toks@ii, for
one of them.

263 \long\def\prepdef#1#2{%
264 \@ifxundefined#1{\toks@{}}{\toks@\expandafter{#1}}%
265 \toks@ii{#2}%
266 \edef#1{\the\toks@ii\the\toks@}%
267 }%
268 \long\def\appdef#1#2{%
269 \@ifxundefined#1{\toks@{}}{\toks@\expandafter{#1}}%
270 \toks@ii{#2}%
271 \edef#1{\the\toks@\the\toks@ii}%
272 }%
273 \long\def\gappdef#1#2{%
274 \@ifxundefined#1{\toks@{}}{\toks@\expandafter{#1}}%
275 \toks@ii{#2}%
276 \global\edef#1{\the\toks@\the\toks@ii}%
277 }%
278 \long\def\appdef@val#1#2{%
279 \appdef#1{{#2}}%
280 }%
281 \long\def\appdef@e#1#2{%
282 \expandafter\appdef
283 \expandafter#1%
284 \expandafter{#2}%
285 }%
286 \long\def\appdef@eval#1#2{%
287 \expandafter\appdef@val
288 \expandafter#1%
289 \expandafter{#2}%
290 }%
291 \toksdef\toks@ii=\tw@

\@ifxundefined Certain utility procedures use \@ifxundefined, which is defined here in terms
\@ifnotrelax of \@ifx. Others use \@ifnotrelax, namely when the control sequence name is
\@argswap manufactured by the use of \csname.
\@argswap@val The procedures \@argswap and \@argswap@val are used to facilitate control of
expansion.

292 \long\def\@ifxundefined#1{\@ifx{\undefined#1}}%

```

```

293 \long\def\@ifnotrelax#1#2#3{\@ifx{\relax#1}{#3}{#2}}%
294 \long\def\@argswap#1#2{#2#1}%
295 \long\def\@argswap@val#1#2{#2{#1}}%
296 \def\@ifxundefined@cs#1{\expandafter\@ifx\expandafter{\csname#1\endcsname\relax}}%

\@boolean In order to define \@ifx, we first must create the “defining word” (term taken from
\@boole@def our Forth vocabulary) \@boole@def, which employs \@boolean to do its job.

297 \def\@boolean#1#2{%
298   \long\def#1{%
299     #2% \if<something>
300     \expandafter\true@sw
301   \else
302     \expandafter\false@sw
303   \fi
304 }%
305 }%
306 \def\@boole@def#1#\@boolean{#1}}% Implicit #2

\@booleantrue The procedures \@booleantrue and \@booleanfalse are assignment operators
\@booleanfalse for Boolean flags.

307 \def\@booleantrue#1{\let#1\true@sw}%
308 \def\@booleanfalse#1{\let#1\false@sw}%

\@ifx We can now invoke the defining word to create the procedures \@ifx and friends.
\@ifx@empty Compatibility Note: earlier versions of this package defined a procedure
\@if@empty \@ifempty. However, for compatibility with AMSLATEX, we must avoid the fol-
\@ifcat lowing three names: \@ifempty, \@xifempty, and \@ifnotempty.

\@ifdim 309 \@boole@def\@ifx#1{\ifx#1}%
\@ifeof 310 \@boole@def\@ifx@empty#1{\ifx\@empty#1}%
\@ifhbox 311 \@boole@def\@if@empty#1{\if!#1!}%
\@ifhmode 312 %\@boole@def\@if@sw#1{\csname if#1\endcsname}%
\@ifinner 313 \def\@if@sw#1#2{#1\expandafter\true@sw\else\expandafter\false@sw#2}%
\@ifmmode 314 \@boole@def\@ifdim#1{\ifdim#1}%
\@ifnum 315 \@boole@def\@ifeof#1{\ifeof#1}%
\@ifodd 316 \@boole@def\@ifhbox#1{\ifhbox#1}%
\@ifvbox 317 \@boole@def\@ifhmode{\ifhmode}%
\@ifvmode 318 \@boole@def\@ifinner{\ifinner}%
\@ifvoid 319 \@boole@def\@ifmmode{\ifmmode}%
320 \@boole@def\@ifnum#1{\ifnum#1}%
321 \@boole@def\@ifodd#1{\ifodd#1}%
322 \@boole@def\@ifvbox#1{\ifvbox#1}%
323 \@boole@def\@ifvmode{\ifvmode}%
324 \@boole@def\@ifvoid#1{\ifvoid#1}%

\true@sw Note that when a Boolean operator expands, it employs two macros that act as
\false@sw selectors, defined here.

325 \long\def\true@sw#1#2{#1}%
326 \long\def\false@sw#1#2{#2}%

```

`\loopuntil` Loop control using the Boolean idiom. Superior to `\loop... \repeat` because these
`\loopwhile` can be nested. The tail of the argument must have a Boolean predicate.

```
327 \long\def\loopuntil#1{#1}{\loopuntil{#1}}%
328 \long\def\loopwhile#1{#1{\loopwhile{#1}}{}}%
```

`\@provide` A defining word that refuses to clobber a prior meaning.

```
329 \def\@provide#1{%
330 \ifx{\undefined#1}{\true@sw}{\ifx{\relax#1}{\true@sw}{\false@sw}}%
331 {\def#1}{\def\j@nk}%
332 }%
```

6.6 Begin Document Structure

The standard L^AT_EX mechanism `\AtBeginDocument` is inadequate because the `\vsize` is bound much too early. We supply here a mechanism whereby decisions about the page layout can be deferred until `\AtBeginDocument` time.

The problem we are working around is that the `\AtBeginDocument` hook in `\document` appears long after the calculation of `\vsize` and `\hsize`, that is, L^AT_EX provides no mechanism for deferring the decision about the page grid until `\AtBeginDocument` time. We fix things by prepending a hook at the very beginning of `\document`.

As it turns out, though, it appears feasible to simply invoke the desired column grid command at `\AtBeginDocument` time, since the MVL has nothing in it at that time that would be problematical.

The facility depends on the stability of this part of L^AT_EX's kernel code (the first token of `\document`), which could change, you see. But considering that L^AT_EX is at this point essentially stagnant once more, we risk it.

`\document` We begin by installing hooks into `\document` that we will manage ourselves. First, we do as `\document` does: end the group begun by `\begin`. Last, we conclude our shenanigans by absorbing the first token of the expansion of `\document`, which we assume to be `\endgroup`.

```
333 \prepdef\document{%
334 \endgroup
335 \document@inithook
336 \true@sw}%
337 }%
```

`\document@inithook` To use, simply `\appdef\document@inithook{<your tokens here>}`.

```
338 \let\document@inithook\@empty
```

`\class@documenthook` We install the last `\AtBeginDocument` hook, namely the procedure `\class@documenthook`.
`\class@enddocumenthook` Within the document class, we will use this hook exclusively, so as to avoid interference from other packages. Similarly with `\class@enddocumenthook`, installed via `\AtEndDocument`.

A document class using this package should do as this package does and just say, `\appdef \class@documenthook` instead of `\AtBeginDocument`, and `\appdef \class@enddocumenthook` instead of `\AtEndDocument`.

```

339 \appdef\document@inithook{%
340 \AtBeginDocument{\class@documenthook}%
341 }%
342 \AtEndDocument{%
343 \class@enddocumenthook
344 }%
345 \let\class@documenthook\@empty
346 \let\class@enddocumenthook\@empty

```

`\enddocument` The standard L^AT_EX `\end{document}` processing is a potential problem, particularly when the output routine has been changed by `ltxgrid`. We separate out the procedure that checks the auxiliary file at the end of the job so that later it can be called from the safety of the output routine. We will do this to ensure that the `\@mainaux` stream is not closed until the last page of the job is shipped out, and that can only be done by coordinating with the output routine.

```

347 \def\enddocument{%

```

The following line from `ltxutil.dtxltxmiscen.dtx` ‘resets `\AtEndDocument` for latex/3060’.

```

348 \let\AtEndDocument\@firstofone
349 \@enddocumenthook
350 \@checkend{document}%

```

The `\clear@document` statement ends the current page (we must guarantee no further shipouts), then executes all cleanup procedures that must occur only after the last shipout. Clients will queue up their procedures via `\AfterLastShipout`, if it exists, otherwise by doing `\appdef\clear@document`.

```

351 \clear@document

```

We are very close to ending the T_EX run, now.

```

352 \check@aux
353 \deadcycles\z@
354 \@@end
355 }%
356 \def\check@aux{\do@check@aux}%
357 \def\do@check@aux{%
358 \@ifsw@if@filesw\fi{%
359 \immediate\closeout\@mainaux
360 \let\@setckpt\@gobbletwo
361 \let\@newl@bel\@testdef
362 \@tempwafalse
363 \makeatletter
364 \input\jobname.aux\relax
365 }{}%
366 \@dofilelist
367 \@ifdim{\font@submax >\fontsubfuzz\relax}{%

```



```

368 \@font@warning{%
369   Size substitutions with differences\MessageBreak
370   up to \font@submax\space have occurred.\@gobbletwo
371 }%
372 }{}%
373 \@defaultsubs
374 \@refundefined
375 \@if@sw@if@filesw\fi{%
376   \@ifx{\@multiplelabels\relax}{%
377     \@if@sw@if@tempswa\fi{%
378       \@latex@warning@no@line{%
379         Label(s) may have changed.
380         Rerun to get cross-references right
381       }%
382     }{}%
383   }{%
384     \@multiplelabels
385   }%
386 }{}%
387 }%

```

`\clear@document` The procedure `\clear@document` is responsible for flushing out the last page of the document, if not already done. The procedure then executes those procedures that must wait for execution until after the last page is shipped out. Clients of `ltxutil`, such as `ltxgrid` and `revtex4` will queue these procedures up via `\AfterLastShipout`, if it exists, otherwise by doing `\appdef\clear@document`.

The command `\Call@AfterLastShipout` is provided by Heiko Oberdiek's `atveryend` package. This package is compatible with `ltxutil`.

Note on compatibility with `atveryend`: we arrange for `\Call@AfterLastShipout` to be called from the safety of the output routine, thereby ensuring that all of the procedures queued up by that package's `\AfterLastShipout` are executed at the right time. We also ensure that `\Call@AfterLastShipout` has a default definition, in case the package was never loaded.

```

388 \def\clear@document{%
389   \clearpage
390   \do@output@cc1v{%
391     \Call@AfterLastShipout
392   }%
393 }%
394 \appdef\class@documenthook{%
395   \providecommand\Call@AfterLastShipout{}%
396 }%

```

6.7 Class Extensions

The \LaTeX procedure `\@onefilewithoptions` is the vehicle for reading in a \LaTeX class or package. The APS RevTeX class implements the use of what are called

“substyles”, actually extensions to the class itself. Any document class can do likewise.

`\class@extension` A procedure similar to L^AT_EX’s `\@onefilewithoptions`, but as an extension to
`\class@extensionfile` the current document class.
`\class@ext@hook` Read in the given file as if it were a document class file. Usage: `\class@extensionfile`
`{\class} \@extension`, where `\class` is a file (similar to `aps.rtx`) and where
`\@extension` is the file extension for `\class`. For instance, to read in the file
`aps.rtx`, do `\class@extensionfile {aps} \substyle@ext`, where the latter has
been define to expand to `.rtx`.

Features supported include passing existing class options on to the class extension, `\AtEndOfClass` processing, a stack that restores `\@currname`, `\@currentx`, `\@clsextension`, and the `\catcode` of ‘@’, fall-back to a control sequence name (with leading ‘rtx@’) if no file exists.

Note that `\LoadClass` gives one the ability to write a class that calls in another class as a (sort of) module: this scheme is like `\LoadClass`, but turned inside out.

```

397 \def\class@extension#1#2{%
398 \IfFileExists{#1.#2}{%
399 \expandafter\class@extensionfile\csname ver@\@currname.\@currentx\endcsname{#1}#2%
400 }{%
401 \csname rtx@#1\endcsname
402 }%
403 }%
404 \def\class@extensionfile#1#2#3{%
405 \@pass@options#3\@unusedoptionlist{#2}%
406 \global\let\@unusedoptionlist\@empty
407 \expandafter\class@ext@hook\csname#2.#3-h@k\endcsname#1{#2}#3%
408 }%
409 \def\class@ext@hook#1#2#3#4{%
410 \@pushfilename@ltx
411 \makeatletter
412 \let\CurrentOption\@empty
413 \@reset@ptions
414 \let#1\@empty
415 \xdef\@currname{#3}%
416 \global\let\@currentx#4%
417 \global\let\@clsextension\@currentx
418 \input{#3.#4}%
419 \@ifl@ter#4{#3}#2{%
420 \class@info{Class extension later than: #2}%
421 }{%
422 \class@info{Class extension earlier: #2}%
423 \@@end
424 }%
425 #1%
426 \let#1\@undefined
427 \expandafter\@ppfilename@ltx\@currnamestack@ltx\@nil
428 \@reset@ptions
429 }%

```

`\@pushfilename` But! L^AT_EX does not provide for a class extension other than `.cls`, therefore we must extend L^AT_EX's file name stack with the file extension of a class extension. This way, procedures like `\ProvidesPackage`, `\OptionNotUsed`, `\ProcessOptions`, `\@reset@options` will still work properly.

```

430 \def\@pushfilename@ltx{%
431 \xdef\@currnamestack@ltx{%
432   {\@currname}%
433   {\@currentx}%
434   {\@clsextension}%
435   {\the\catcode'\@}%
436   \@currnamestack@ltx
437 }%
438 }%
439 \def\@p@pfilename@ltx#1#2#3#4#5\@nil{%
440 \gdef\@currname{#1}%
441 \gdef\@currentx{#2}%
442 \gdef\@clsextension{#3}%
443 \catcode'\@#4\relax
444 \gdef\@currnamestack@ltx{#5}%
445 }%
446 \global\let\@currnamestack@ltx\@empty

```

We carefully patch L^AT_EX so that the current value of `\@clsextension` can be restored after reading in a class file.

6.8 Type Tools

`\flushing` Undoes `\centering`. Should also undo `\raggedleft` and `\raggedright`.

```

447 \def\flushing{%
448   \let\\\@normalcr
449   \leftskip\z@skip
450   \rightskip\z@skip
451   \@rightskip\z@skip
452   \parfillskip\@flushglue
453 }%

```

`\@centercr` The `\@centercr` command is the replacement for `\@normalcr` when setting type centered or ragged. Normally, the meaning of `\\` is `\@normalcr`, which L^AT_EX defines via `\DeclareRobustCommand`. In centered or ragged typesetting, the meaning of `\\` is `\@centercr`, therefore it ought to be defined via `\DeclareRobustCommand` (but unfortunately is not). The fact that it is not is yet another of L^AT_EX's early failures that will never get fixed.

The following exemplar fails under L^AT_EX version 2005/12/01, package `textcase` 2004/10/07 v0.07:

```

%\documentclass{article}%
%\usepackage[overload]{textcase}
%\begin{document}

```

```

%\centering
%\section{MakeTextUppercase{Section\\title}}
%Text
%\end{document}
%
```

The solution is to promote `\centercr` to a robust command, just the same as `\.`. We do that here without needing to know the meaning of the command.

```
454 \expandafter\DeclareRobustCommand\expandafter\centercr\expandafter{\centercr}%
```

6.9 Display Math

`\eqnarray@LaTeX` Team L^AT_EX has stated they will never repair Leslie's broken definition of `\eqnarray`. Let us be bold...

Note on `hyperref` package compatibility: that package overrides `\eqnarray` by wrapping it up in a larger procedure, so its changes are compatible with this package's changes.

```

455 \def\eqnarray@LaTeX{%
456   \stepcounter{equation}%
457   \def\@currentlabel{\p@equation\theequation}%
458   \global\@eqnswtrue
459   \m@th
460   \global\@eqcnt\z@
461   \tabskip\@centering
462   \let\@@@eqnocr
463   $$\everycr{}\halign to\displaywidth\bgroup
464     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnrel
465     &\global\@eqcnt\@ne\hskip \tw@\arraycolsep \hfil${##}$\hfil
466     &\global\@eqcnt\tw@ \hskip \tw@\arraycolsep
467     $\displaystyle{##}$\hfil\tabskip\@centering
468     &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
469     \tabskip\z@skip
470   \cr
471 }
472 \long\def\eqnarray@fleqn@fixed{%
473   \stepcounter{equation}\def\@currentlabel{\p@equation\theequation}%
474   \global\@eqnswtrue\m@th\global\@eqcnt\z@
475   \tabskip\ltx@mathindent
476   \let\@@@eqnocr
477   \setlength\abovedisplayskip{\topsep}%
478   \ifvmode\addtolength\abovedisplayskip{\partopsep}\fi
479   \addtolength\abovedisplayskip{\parskip}%
480   \setlength\belowdisplayskip{\abovedisplayskip}%
481   \setlength\belowdisplayshortskip{\abovedisplayskip}%
482   \setlength\abovedisplayshortskip{\abovedisplayskip}%
483   $$$
484   \everycr{}%
485   \halign@\linewidth\bgroup
486   \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnrel
```

```

487 &\global\@eqcnt\@ne
488 \hskip\tw@eqncolsep
489 \hfil${}\##{}}$\hfil
490 &\global\@eqcnt\tw@
491 \hskip\tw@eqncolsep
492 $\displaystyle{##}$\hfil\tabskip\@centering
493 &\global\@eqcnt\thr@@\hb@xt@\z@\bgroup\hss##\egroup
494 \tabskip\z@skip
495 \cr
496 }%
497 \@ifx{\eqnarray\eqnarray@LaTeX}{%
498 \class@info{Repairing broken LaTeX eqnarray}%
499 \let\eqnarray\eqnarray@fleqn@fixed
500 \newlength\eqncolsep
501 \setlength\eqncolsep\z@
502 \let\eqnarray@LaTeX\relax
503 \let\eqnarray@fleqn@fixed\relax
504 }{}%

```

The macro `\ltx@mathindent` is assigned to the `\tabskip` glue just before the alignment preamble is expanded, the value therefore applying at the left of the first column.

The below value specifies the display math to be set centered, as is common practice. Alternatively, `\tabskip` can be set to a different glue value, accomplishing flush-left display math.

Note that the `ltxutil.dtxfleqn.clo` package provides its own meaning for the `eqnarray` environment, which is also broken. We do not patch that package, however.

Bug note: The `ltxutil.dtxlineno.sty` package detects `ltxutil.dtxfleqn.clo` by testing whether `\mathindent` is defined, instead of using correct L^AT_EX 2_ε means. Even though our `eqnarray` environment is modelled after `ltxutil.dtxfleqn.clo`, we must program defensively here.

```

505 \def\ltx@mathindent{\@centering}%
506 \def\set@eqnarray@skips{}%

```

`\prep@math` If we are in vertical mode when display math mode is entered (via `$$`), T_EX will first enter horizontal mode, then display math mode; this results in a phantom paragraph containing a single `\hbox` consisting of the `\parindent` box followed by the `\parskipfilllskip` glue. Of course, that `\hbox` is accompanied by `\parskip` glue and `\baselineskip` glue.

The `\prep@math` procedure removes the `\parindent` box, thereby (magically) eliminating the phantom paragraph. The `\prep@math@patch` procedure head-patches the `equation` and `eqnarray` environments to accomplish this removal of the phantom paragraph.

Note that there are three remaining ways to enter display math mode that we do not treat: the `displaymath` environment (equivalent to `\[/\]`), and the primitive the `$$` markup. I refrain from treating the first case because `displaymath` already detects the case where it is entered from vertical mode: I do not wish to

engage in the dubious enterprise of attempting to correct a procedure that is ill conceived from the outset. As to the primitive `$$`, there is no help for users who insist upon employing procedural markup in their documents. in their documents.

```
507 \def\prep@math{%
508 \@ifvmode{\everypar{\setbox\z@\lastbox}}{}}%
509 }%
510 \def\prep@math@patch{%
511 \prepdef\equation{\prep@math}%
512 \prepdef\eqnarray{\prep@math}%
513 }%
```

A document class may invoke `\prep@math@patch` at any point it wishes to prevent the appearance of the phantom paragraph: it may be a global declaration or a local one.

We fail to patch `\[`, `\equation`, however.

6.10 Footnotes

<code>\footnotemark</code>	We repair an error in the L ^A T _E X kernel (see <code>lfloat.dtx</code>) involving footnotes.
<code>\footnotetest</code>	The symptom is that the <code>\footnotemark</code> command does not work properly
<code>\ltx@xfootnote</code>	within a <code>minipage</code> environment. The source of the problem is in the way the
<code>\ltx@footmark</code>	<code>\footnotemark</code> and <code>\@xfootnotemark</code> procedures are defined: they do not share
<code>\ltx@foottext</code>	the method, used by <code>\footnote</code> and other procedures, that allows a context switch
<code>\ltx@make@current@footnote</code>	to change the way footnotes behave within a <code>minipage</code> environment. This is a

L^AT_EX bug of long standing; our fix dates to 1987.

While we are at it, we rewrite both the `\footnote`, `\footnotemark` and `\footnotetest` procedures, achieving a cleaner separation of syntax and semantics. Note that the `\@footnotetest` procedure is not involved in context switching; `hyperref` will take over that procedure, substituting its own processing around its argument and passing this to `\H@@footnotetest`. We anticipate this, and do our context switching on `\H@@footnotetest`.

The `\@makefnmark` continues as the method of formatting the footnote mark.

A note about the context switch mentioned above: the `minipage` environment executes the following in order to alter the way footnotes behave:

```
%\def\@mpfn{mpfootnote}%
%\def\@thempfn{\thempfootnote}%
%\let\@footnotetest\@mpfootnotetest
%\let\@makefnmark\@mpmakefnmark
%\c@mpfootnote\z@
```

This code changes the counter used in autonumbered footnotes, the choice of footnote marker, and the procedure used on the footnote text. Changing the counter is needed because `minipage` footnotes are in their own sequence, and the footnote marker is customarily different within a `minipage`. The procedure that works on the footnote text must be different because the footnotes are placed at the bottom of the `minipage`, not the bottom of the text column.

Note that L^AT_EX initially defines `\@mpfn` as `footnote` and `\thempfn` as `\thefootnote`, so we are initially doing general footnotes.

Any procedure that establishes a minipage-like context (e.g., floats) can do the same as the minipage context switch illustrated above.

Three user-level command, `\footnote`, `\footnotemark`, and `\footnotetext` are defined (see the L^AT_EX manual for user-level details).

`\footnote` The first user-level command is `\footnote`. A simple way to look at this command is to think of it as `\footnotemark [number] \footnotetext [number] {text}`, where the optional argument is the same in both calls. We also define a syntactical helper procedure `\ltx@xfootnote`.

We employ the procedures `\ltx@stp@footproc` and `\ltx@def@footproc`, passing in the procedure to execute, in this case `\ltx@footmark`, which sets the footnote mark. In any case, we end on the procedure `\ltx@foottext`, which sets the footnote text.

```
514 \def\footnote{\@ifnextchar[\ltx@xfootnote\ltx@yfootnote]%
515 \def\ltx@xfootnote[#1]{%
516   \ltx@def@footproc\ltx@footmark[#1]%
517   \expandafter\ltx@foottext\expandafter{\the\csname c@\@mpfn\endcsname}%
518 }%
519 \def\ltx@yfootnote{%
520   \ltx@stp@footproc\ltx@footmark
521   \expandafter\ltx@foottext\expandafter{\the\csname c@\@mpfn\endcsname}%
522 }%
```

The `\footmark` user-level command is next. Here we use the procedures `\ltx@stp@footproc` and `\ltx@def@footproc` again, but unlike `\footnote`, we do not set the footnote text.

```
523 \def\footnotemark{\@ifnextchar[\ltx@xfootmark\ltx@yfootmark]%
524 \def\ltx@xfootmark{\ltx@def@footproc\ltx@footmark}%
525 \def\ltx@yfootmark{\ltx@stp@footproc\ltx@footmark}%
526 \def\ltx@footmark#1{%
527   \leavevmode
528   \ifhmode\edef\@x@sf{\the\spacefactor}\nobreak\fi
529   \begingroup
530     \expandafter\ltx@make@current@footnote\expandafter{\@mpfn}{#1}%
531     \expandafter\@argswap@val\expandafter{\Hy@footnote@currentHref}{\hyper@linkstart {link}}%
532     \@makefnmark
533     \hyper@linkend
534   \endgroup
535   \ifhmode\spacefactor\@x@sf\fi
536   \relax
537 }%
```

The third user-level command is `\footnotetext`. As with `\footnotemark`, we use the procedures `\ltx@stp@footproc` and `\ltx@def@footproc`, this time passing in the procedure `\ltx@foottext`, which sets the footnote text.

```
538 \def\footnotetext{\@ifnextchar[\ltx@xfoottext\ltx@yfoottext]%
539 \def\ltx@xfoottext{\ltx@def@footproc\ltx@foottext}%

```

```

540 \def\ltx@yfoottext{\ltx@stp@footproc\ltx@foottext}%
541 \long\def\ltx@foottext#1#2{%
542   \begingroup
543   \expandafter\ltx@make@current@footnote\expandafter{\@mpfn}{#1}%
544   \@footnotetext{#2}%
545   \endgroup
546 }%

```

Here are the definitions of the procedures `\ltx@stp@footproc` and `\ltx@def@footproc`. The require argument is the procedure to execute afterwards, and `\ltx@def@footproc` parses a bracket-delimited argument (it is not optional). In each case the given procedure is executed with an argument prepared for it: the value of the footnote counter.

```

547 \def\ltx@def@footproc#1[#2]{%
548   \begingroup
549   \csname c@\@mpfn\endcsname #2\relax
550   \unrestored@protected@xdef\@thefnmark{\thempfn}%
551   \expandafter\endgroup
552   \expandafter#1%
553   \expandafter{\the\csname c@\@mpfn\endcsname}%
554 }%
555 \def\ltx@stp@footproc#1{%
556   \expandafter\stepcounter\expandafter{\@mpfn}%
557   \protected@xdef\@thefnmark{\thempfn}%
558   \expandafter#1%
559   \expandafter{\the\csname c@\@mpfn\endcsname}%
560 }%

```

Here we provide for our good friend `hyperref` to enter in like a bull in a china shop. If it is not loaded, we do what it would have done, but gentlier and without hypertext functionality.

```

561 \appdef\class@documenthook{%
562   \let\footnote@latex\footnote
563   \@ifpackageloaded{hyperref}{}{%
564     \let\H@@footnotetext\@footnotetext
565     \def\@footnotetext{\H@@footnotetext}%
566     \let\H@@mpfootnotetext\@mpfootnotetext
567     \def\@mpfootnotetext{\H@@mpfootnotetext}%
568   }%
569 }%

```

In the following, we must use L^AT_EX's rococco equipment in the form of `\protected@edef`, because of the presence of a font switch in the meaning of `\thempfootnote`. But, really, isn't this a sloppy conflation of semantics and presentation?

```

570 \def\ltx@make@current@footnote#1#2{%
571   \csname c#1\endcsname#2\relax
572   \protected@edef\Hy@footnote@currentHref{\@currentHref-#1.\csname the#1\endcsname}%
573 }%
574 \def\thempfootnote@latex{\itshape \@alph \c@mpfootnote }%

```



```

575 \def\ltx@thempfootnote{\@alph@c@mpfootnote}%
576 \ifx{\thempfootnote\thempfootnote@latex}{%
577 \class@info{Repairing hyperref-unfriendly LaTeX definition of \string\mpfootnote}%
578 \let\thempfootnote\ltx@thempfootnote
579 }{}%

```

Note on `hyperref` compatibility: In its “Automated L^AT_EX hypertext cross-references”, the `hyperref` package alters footnote processing, but it does nothing to address the several issues of concern to us.

The `hyperref` package takes over the `\@mpfootnotetext` and `\@footnotetext` procedures, wrapping the argument in its own code. It also rewrites `\@footnotemark`, making it a hyperlink, and `\@xfootnotenext`, removing from it all hypertext capabilities.

However, if the `\footnotemark` command has been supplied with an optional argument, `hyperref`’s changes do not apply: it punts in this case.

At the same time, it attempts to turn off its changes during `\maketitle` processing, destroying one of the capabilities we desire.

We make ourself `hyperref` savvy: we re-implement footnote processing, using `hyperref` capabilities if that package has been loaded.

Any other package that rewrites L^AT_EX’s footnote macros will be incompatible with this package.

Two thoughts about `hyperref`: what for does it define `\realfootnote`? Apparently even SR himself cannot remember.

Also: a document class that desires high hypertext capabilities might well wish to reimplement `\maketitle` so that footnotes called out from there are hypertext links: the `hyperref` package’s “Automated L^AT_EX hypertext cross-references” does not do any of this:

But the special footnotes in `\maketitle` are much too hard to deal with properly. Let them revert to plain behaviour.

Note that the document class, in reimplementing `\maketitle`, must ensure that the `hyperref` package does not clobber its own definition!

```

\@footnotetext The two procedures \@footnotetext and \@mpfootnotetext share code. We
\@mpfootnotetext make that explicit here.
\@tpfootnotetext Note that the procedure calling \make@footnotetext will open a group with
\make@footnotetext \bgroup which is then closed by \minipagefootnote@drop.
\set@footnotewidth Difference from LATEX: here we do not set \floatingpenalty to infinity. Doing
this must date back to a time when LATEX could not accomodate split insertions
(footnotes). I cannot think of any other reason to do have done this. At any rate,
with the ltxgrid package, split insertions are properly taken care of, so we allow
it.

```

We provide the hook `\set@footnotewidth` that sets the footnote on a particular measure. Some page grids are such as to set a footnote in a context where `\columnwidth` is not the right parameter to use for the set width of a footnote. In such a case, for the applicable scope, you should define `\set@footnotewidth` to perform this job correctly.

If we are setting type on multiple page grids, we must still ensure that all footnotes that find their way into the `\footins` insert register are set on the same width. This implies the need for a document to have an “overall” page grid, which determines the set width of all footnotes with the exception of minipage footnotes.

In general, remember that footnotes, like all insertions (including floats), are a step outside of the galley context, and all aspects of insertions need to be properly handled, including the set width.

```

580 \def\@makefnmark{%
581   \hbox{%
582     \@textsuperscript{%
583       \normalfont\itshape\@thefnmark
584     }%
585   }%
586 }%

587 \long\def\@footnotetext{%
588   \insert\footins\bgroup
589   \make@footnotetext
590 }%

591 \long\def\@mpfootnotetext{%
592   \minipagefootnote@pick
593   \make@footnotetext
594 }%

```

Procedure `\make@footnotetext` sets the footnote #1 into type, with the proper font, color, leading, width, and label in effect. It also establishes a strut and null glue at the end of the last paragraph of the footnote; The strut helps compensate for the lack of `\interlineskip` glue between `\inserts`; the glue establishes a feasible `\vsplit` point between footnotes.

Note that in the title block (`ltxfront`), the alternative definition, under the name `\frontmatter@footnotetext`, is used. The only material difference there is the reference to `\frontmatter@makefnntext` instead of `\@makefnntext`.

Dependency note: the `\@makefnntext` procedure is used to further process the footnote text and to execute the `\@makefnmark` procedure to produce the footnote mark. The definition of the former is customarily found in the document class (hereunder that of `ltxutil.dtxarticle.cls`), the latter in `ltxutil.dtxlatex.ltx`. They are as follows:

```

%\newcommand\@makefnntext[1]{%
% \parindent 1em\noindent
% \hb@xt@1.8em{\hss\@makefnmark}%
% #1%
%}%
%\def\@makefnmark{%
% \hbox{\@textsuperscript{\normalfont\@thefnmark}}%
%}%
%

595 \long\def\make@footnotetext#1{%

```

```
596 \set@footnotefont
```

As noted above, we do *not* do `\floatingpenalty \@MM`, as in standard L^AT_EX.

```
597 \set@footnotewidth
598 \@parboxrestore
599 \protected@edef\@currentlabel{%
```

Note that we employ `\@mpfn` as a level of redirection for the footnotecounter.

```
600 \csname p@\@mpfn\endcsname\@thefnmark
601 }%
602 \color@begingroup
603 \makefnmark{
604 \rule\z@\footnotesep\ignorespaces#1%
```

The following strut and glue are for spacing and splitting, as mentioned above.

```
605 \finalstrut\strutbox\vadjust{\vskip\z@skip}%
606 }%
607 \color@endgroup
608 \minipagefootnote@drop
609 }%
```

`\set@footnotefont` is the procedure for setting the font of a footnote. Other aspects of the environment may be set using this hook.

```
610 \def\set@footnotefont{
611 \reset@font\footnotesize
612 \interlinepenalty\interfootnotelinepenalty
613 \splittopskip\footnotesep
614 \splitmaxdepth\dp\strutbox
615 }%
```

`\set@footnotewidth` is the procedure for setting the width of a footnote. The default page grid, a single, full-width column, sets footnotes on the width of the text.

```
616 \def\set@footnotewidth{\set@footnotewidth@one}%
```

6.11 Floats

6.11.1 Usage notes

We extend the L^AT_EX kernel for three purposes:

1. When the `\footnote` command is used within the scope of a float, we do as `minipage` does.
2. We provide a mechanism to write floats out to an external stream for temporary storage (deferred floats).
3. We provide mechanism for placing a float **here** invariably, that is, floats are unfloat. This mechanism is used to read the external stream mentioned above.

To use these mechanisms, the document class should define a float, say, `figure` as per usual, and in addition:

1. Optionally define an alternative, say `figure@write` as follows:

```
\newenvironment{figure@write}{%
% \write@float{figure}%
%}{%
% \endwrite@float
%}
```

That is, the alternative environment executes `\write@float` instead of `\@float`. Note that this step is not needed if the float environment is defined in the simple way of `classes.dtx`. However, an environment like `longtable` will require it.

2. Install into `\AtBeginDocument` a call to `\do@if@floats`, with the float name and an appropriate file extension as its arguments.

```
\appdef\class@documenthook{\do@if@floats{figure}{.fgx}}
```

3. Optionally define a text entity `\figuresname` that will be the text of the head that is set over the deferred floats. If not defined, there will be no head.
4. Optionally define a user-level command to allow the document to determine where the figures are printed out (default is to print at end of document). E.g.,

```
\newcommand\printfigures{\print@float{figure}}
```

5. Install into `\appdef\class@enddocumenthook` a call to `\printfigures`, or, if the latter is not defined, as follows:

```
\appdef\class@enddocumenthook{\print@float{figure}}
```

Note that installing this command into `\AtBeginDocument` is best done earlier than calls that assume the last page of the document is at hand.

6.11.2 Robustifying fragile commands

Certain of L^AT_EX's commands cannot be written out to a file or appear within a `\mark` command argument because they do calculations during expansion. We provide for a little help, but without changing the meanings of these commands.

```

\addtocontents
\robustify@contents 617 \def\robustify@contents{%
618 \let \label \@gobble
619 \let \index \@gobble
620 \let \glossary \@gobble
621 \let\footnote \@gobble
622 \def\({\string\}%
623 \def\)\string\}%
624 \def\{\string\}%
625 }%
626 \long\def\addtocontents#1#2{%
627 \protected@write\@auxout{\robustify@contents}{\string \@writefile {#1}{#2}}%
628 }%

```

6.11.3 Preparing for the hyperref package

`\addcontentsline` The `hyperref` package assumes that the `\contentsline` command will be given four arguments. Therefore it cannot successfully process a `ltxutil.dtx.toc` file that had been written by standard L^AT_EX. We fix things up by always writing that fourth argument and by supplying a `\contentsline` command that can read them.

We also give the `\newlabel` command's second argument five tokens.

Finally, we wrap L^AT_EX's `\contentsline` command with code to detect the case where the expected procedure is not defined, and we give it a syntax with no semantics.

We switch over to this new definition only after `hyperref` has loaded.

```

629 \def\addcontentsline#1#2#3{%
630 \addtocontents{#1}{%
631 \protect\contentsline{#2}{#3}{\thepage}{}%
632 }%
633 }%
634 \def\label#1{%
635 \@bsphack
636 \protected@write\@auxout{%
637 \string\newlabel{#1}{\@currentlabel}{\thepage}{}}%
638 }%
639 \@esphack
640 }%
641 \def\ltx@contentsline#1{%
642 \expandafter\ifnotrelax\csname l@#1\endcsname}{%
643 \expandafter\let\csname l@#1\endcsname\@gobbletwo
644 }%
645 \contentsline@latex{#1}%
646 }%
647 \appdef\document@inithook{%
648 \let\contentsline@latex\contentsline
649 \let\contentsline\ltx@contentsline
650 }%

```

6.11.4 Footnotes within floats, unfloating floats, float font

`\caption` DPC: Er a bit of a hack, but seems best way of supporting normal L^AT_EX syntax at this point: If a caption is used below a table, then put out the footnotes before the caption.

```
651 \appdef\class@documenthook{%
652 \prepdef\caption{\minipagefootnote@here}%
653 }%
```

Note on `hyperref` compatibility: this change to the `\caption` command is compatible with the “Automated L^AT_EX hypertext cross-references” patches of that package.

All the same, I think Sebastian’s changes to `\caption` and `\@caption` could bear with some improvement. The following implementation requires knowing only the pattern part of the `\@caption` macro:

```
%\def\caption{%
% \H@refstepcounter\@capttype
% \hyper@makecurrent{\@capttype}%
% \@dblarg{\H@caption\@capttype}%
%}%
%\def\H@caption#1[#2]#3{%
% \@caption{#1}[#2]{%
% \ifHy@nesting
% \hyper@@anchor{\@currentHref}{#3}%
% \else
% \hyper@@anchor{\@currentHref}{\relax}#3%
% \fi
% }%
%}
```

`\minipagefootnote@init` Procedure to deal with footnotes accumulated within a minipage environment.

`\minipagefootnote@here` These procedures encapsulate all uses of the `\@mpfootins` box.

`\minipagefootnote@foot` Note: `\minipagefootnote@here` must *not* be executed within the MVL!

```
\minipagefootnote@pick 654 \def\minipagefootnote@init{%
\minipagefootnote@drop 655 \setbox\@mpfootins\box\voidb@x
656 }%
657 \def\minipagefootnote@pick{%
658 \global\setbox\@mpfootins\vbox\bgroup
659 \unvbox\@mpfootins
660 }%
661 \def\minipagefootnote@drop{%
662 \egroup
663 }%
664 \def\minipagefootnote@here{%
665 \par
666 \@ifvoid\@mpfootins{}{%
667 \vskip\skip\@mpfootins
668 \fullinterlineskip
669 \@ifinner{%
```

```

670      \vtop{\unvcopy\@mpfootins}%
671      {\setbox\z@\lastbox}%
672      }-}%
673      \unvbox\@mpfootins
674  }%
675 }%
676 \def\minipagefootnote@foot{%
677 \@ifvoid\@mpfootins{}-}%
678 \insert\footins\bgroup\unvbox\@mpfootins\egroup
679 }%
680 }%
681 \def\endminipage{%
682   \par
683   \unskip
684   \minipagefootnote@here
685   \@minipagefalse   %% added 24 May 89
686   \color@endgroup
687   \egroup
688   \expandafter\@iiiparbox\@mpargs{\unvbox\@tempboxa}%
689 }%

```

`\floats@sw` The Boolean `\floats@sw` signifies that floats are to be floated; if false, that floats are to be deferred to the end of the document. Note that the assignment of this Boolean is to be overridden by the document class in response to user-selected options.

```
690 \@booleantrue\floats@sw
```

`\@xfloat` The float start-code is redefined to set up footnotes in the style of minipage. Also, `\@mpmakefntext` the `\floats@sw` Boolean informs us that floats are to be all placed **here**. Note that, to protect against the Boolean being undefined at this late hour, we default it globally to true.

```

691 \let\@xfloat@LaTeX\@xfloat
692 \def\@xfloat#1[#2]{%
693   \@xfloat@prep
694   \@nameuse{fp@proc#2}%
695   \floats@sw{\@xfloat@LaTeX{#1}[#2]}{\@xfloat@anchored{#1}[]}%
696 }%
697 \def\@xfloat@prep{%
698   \ltx@footnote@pop
699   \def\@mpfn{mpfootnote}%
700   \def\thempfn{\thempfootnote}%
701   \c@mpfootnote\z@
702   \let\H@footnotetext\H@mpfootnotetext
703 }%
704 \let\ltx@footnote@pop\@empty
705 \def\@xfloat@anchored#1[#2]{%
706   \def\@capttype{#1}%
707   \begin@float@pagebreak
708   \let\end@float\end@float@anchored

```

```

709 \let\end@dblfloat\end@float@anchored
710     \hsize\columnwidth
711     \@parboxrestore
712     \@floatboxreset
713 \minipagefootnote@init
714 }%
715 \def\end@float@anchored{%
716 \minipagefootnote@here
717 \par\vskip\z@skip
718 \par
719 \end@float@pagebreak
720 }%
721 \def\begin@float@pagebreak{\par\addvspace\intextsep}%
722 \def\end@float@pagebreak{\par\addvspace\intextsep}%
723 \def\@mpmakefntext#1{%
724 \parindent=1em
725 \noindent
726 \hb@xt@1em{\hss\@makefnmark}}%
727 #1%
728 }%

```

6.11.5 Writing floats out to a file

`\do@if@floats` The procedure `\do@if@floats` should be executed at `\class@documenthook` time: it arranges to write out the floats of the given class to a temporary file, to be read back later (deferred floats), given that `\floats@sw` is false. Note that, to protect against the Boolean being undefined at this late hour, we default it globally to true.

```

729 \def\do@if@floats#1#2{%
730 \floats@sw}{-%

```

Open the stream to save out the document's floats of this class.

```

731 \expandafter\newwrite
732     \csname#1write\endcsname
733 \expandafter\def
734     \csname#1@stream\endcsname{\jobname#2}%
735 \expandafter\immediate
736 \expandafter\openout
737     \csname#1write\endcsname
738     \csname#1@stream\endcsname\relax

```

Swap environments. If the class writer has defined, e.g., `figure@write`, then we use this as the procedure to execute for writing the float out to the external stream. Otherwise, the replacement of `\@float` by `\write@float` should do the right thing for float environments defined in the simple way of `classes.dtx`.

```

739 \@ifxundefined\@float@LaTeX{%
740 \let\@float@LaTeX\@float
741 \let\@dblfloat@LaTeX\@dblfloat
742 \let\@float\write@float
743 \let\@dblfloat\write@floats

```



```

744 }{}%
745 \let@environment{#1@float}{#1}%
746 \let@environment{#1@floats}{#1*}%
747 \@ifxundefined@cs{#1@write}{-}%
748 \let@environment{#1}{#1@write}%
749 }%
750 }%
751 }%

```

`\print@float` The procedure `\print@float` prints out the deferred floats.

Here, we make use of the `\floats@sw` Boolean to select the non-floating type of processing.

```

752 \def\triggerpar{\leavevmode\@par}%
753 \def\onepage{\def\begin@float@pagebreak{\newpage}\def@end@float@pagebreak{\newpage}}%
754 \def\print@float#1#2{%
755 \lengthcheck@sw{%
756 \total@float{#1}%
757 }{}%
758 \@ifxundefined@cs{#1write}{-}%
759 \begingroup
760 \@booleanfalse\floats@sw
761 #2%
762 \raggedbottom
763 \def\array@default{v}% floats must
764 \let@float\@float@LaTeX
765 \let@dblfloat\@dblfloat@LaTeX
766 \let\trigger@float@par\triggerpar
767 \let@environment{#1}{#1@float}%
768 \let@environment{#1*}{#1@floats}%
769 \expandafter\prepdef\csname#1\endcsname{\trigger@float@par}%
770 \expandafter\prepdef\csname#1*\endcsname{\trigger@float@par}%
771 \@namedef{fps#1}{h!}%
772 \expandafter\immediate
773 \expandafter\closeout
774 \csname#1write\endcsname
775 \everypar{%
776 \global\let\trigger@float@par\relax
777 \global\everypar{}\setbox\z@\lastbox
778 \@ifxundefined@cs{#1sname}{-}%
779 \begin@float@pagebreak
780 \expandafter\section
781 \expandafter*%
782 \expandafter{%
783 \csname#1sname\endcsname
784 }%
785 }%
786 }%
787 \input{\csname#1@stream\endcsname}%
788 \endgroup
789 \global\expandafter\let\csname#1write\endcsname\relax

```

790 }%
791 }%

`\tally@float` If we are tallying column inches, `\tally@float` tallies a contribution to `\ftype@`
`\total@float` `\@capytype`, depending upon the width of `\@currbox`. In effect, each float class is tallied in two sections, one for narrow, one for wide floats.

If statistics are wanted, `\total@float` logs the tally for the given float class. The quantity `\@twopowerfourteen` is 2^{14} , `\@twopowertwo` is 2^2 .

```
792 \chardef\@xvi=16\relax
793 \mathchardef\@twopowerfourteen="4000
794 \mathchardef\@twopowertwo="4
795 \def\tally@float#1{%
796 \begingroup
```

We strip all but the least significant 5 bits from `\count \@currbox`, and put them into `\@tempcnta`. We then subtract 16 from `\count \@currbox` (unless this would make it negative), effectively reversing the process carried out in `\@float`.

```
797 \@tempcnta\count\@currbox
798 \divide\@tempcnta\@xxxii
799 \multiply\@tempcnta\@xxxii
800 \advance\count\@currbox-\@tempcnta
801 \divide\@tempcnta\@xxxii
802 \@ifnum{\count\@currbox>\@xvi}{%
803 \advance\count\@currbox-\@xvi\@booleantrue\@temp@sw
804 }{%
805 \@booleanfalse\@temp@sw
806 }%
```

If so desired, we log the characteristics of this float object: float class and float placement parameters, height, depth, and width.

```
807 \show@box@size@sw{%
808 \class@info{Float #1
809 (\the\@tempcnta)[\@temp@sw{16+}]}\the\count\@currbox]^^J%
810 (\the\ht\@currbox+\the\dp\@currbox)X\the\wd\@currbox
811 }%
812 }{%
813 \endgroup
```

Here we tally the height of this float object.

```
814 \expandafter\let
815 \expandafter\@tempa
816 \csname fbox@\csname ftype@#1\endcsname\endcsname
817 \@ifnotrelax\@tempa{%
818 \@ifhbox\@tempa{%
819 \setbox\@tempboxa\vbox{\unvcopy\@currbox\hrule}%
820 \dimen@ht\@tempboxa
821 \divide\dimen@\@twopowerfourteen
822 \@ifdim{\wd\@tempboxa<\textwidth}{%
823 \advance\dimen@\ht\@tempa
824 \global\ht\@tempa\dimen@
```

```

825 }{%
826   \advance\dimen@\dp\@tempa
827   \global\dp\@tempa\dimen@
828 }%
829 }{%}
830 }{%}
831 }%
832 \def\total@float#1{%
833   \expandafter\let
834   \expandafter\@tempa
835     \csname fbox@\csname ftype#1\endcsname\endcsname
836   \@ifnotrelax\@tempa{%
837     \@ifhbox\@tempa{%
838       \@tempdima\the\ht\@tempa\divide\@tempdima\@twopowertwo\@tempcnta\@tempdima
839       \@tempdimb\the\dp\@tempa\divide\@tempdimb\@twopowertwo\@tempcntb\@tempdimb
840       \class@info{Total #1: Column(\the\@tempcnta pt), Page(\the\@tempcnta pt)}%
841     }{%}
842   }{%}
843 }%

```

`\write@float` Handles the case where the name of the float is the same as that of the stream.
`\write@floats` Note that `longtable` does *not* fit this case. Note also: `\write@float` is *not* a
`\write@@float` user-level environment, therefore it is properly not defined with `\newenvironment`.

```

844 \def\write@float#1{\write@@float{#1}{#1}}%
845 \def\endwrite@float{\@Esphack}%
846 \def\write@floats#1{\write@float{#1*}{#1}}%
847 \def\endwrite@floats{\@Esphack}%

```

`\write@@float`

```

848 \def\write@@float#1#2{%
849   \ifhmode
850     \@bsphack
851   \fi
852   \chardef\@tempc\csname#2write\endcsname
853   \toks@{\begin{#1}}%
854   \def\@tempb{#1}%
855   \expandafter\let\csname end#1\endcsname\endwrite@float
856   \catcode'\^M\active
857   \@makeother\{\@makeother\}\@makeother\%
858   \write@floatline
859 }%

```

`\write@floatline` The procedure `\write@floatline` only parses; it passes its result to `\@write@floatline`,
`\@write@floatline` which writes the line to output, then tests the line for the `\end{float}` tokens
`\float@end@tag` with aid of the `\float@end@tag` procedure.

```

860 \begingroup
861 \catcode'\[\the\catcode'\{\catcode'\}\the\catcode'\}\@makeother\{\@makeother\}%
862 \gdef\float@end@tag#1\end{#2}#3\nul[%
863   \def\@tempa[#2]%

```

```

864 \@ifx[\@tempa\@tempb][\end[#2]][\write@floatline]%
865 ]%
866 \obeylines%
867 \gdef\write@floatline#1^~M[%
868 \begingroup%
869 \newlinechar‘^~M%
870 \toks@\expandafter[\the\toks@#1]\immediate\write\@tempc[\the\toks@]%
871 \endgroup%
872 \toks@[]%
873 \float@end@tag#1\end{}}\@nul%
874 ]%
875 \endgroup

```

6.12 Counters

The following definitions override those of the L^AT_EX kernel, providing for a greater range of inputs.

```

876 \def\@alph#1{\ifcase#1\or a\or b\or c\or d\else\@ialph{#1}\fi}
877 \def\@ialph#1{\ifcase#1\or \or \or \or \or e\or f\or g\or h\or i\or j\or
878 k\or l\or m\or n\or o\or p\or q\or r\or s\or t\or u\or v\or w\or x\or
879 y\or z\or aa\or bb\or cc\or dd\or ee\or ff\or gg\or hh\or ii\or jj\or
880 kk\or ll\or mm\or nn\or oo\or pp\or qq\or rr\or ss\or tt\or uu\or
881 vv\or ww\or xx\or yy\or zz\else\@ctrerr\fi}

```

6.13 Customization of Sections

Patch the standard L^AT_EX sectioning procedure to:

- Allow a sectioning command to trigger the title page, or more generally to recognize that it is the first object in the document, so we headpatch `\@startsection`.
- Allow a tail command in #6 to uppercase the title, so we retain DPC’s braces.
- Allow each type of sectioning command to format its number differently, so we generalize `\@secntformat`.
- Allow each type of sectioning command to format its argument differently, so we generalize `\@hangfrom`.
- Allow the starred form of the command to mark (the running head) and make an entry in the TOC, so we put `\@ssect` on the same footing as `\@sect`.

Note that the tokens passed to the TOC now are *not* the optional argument of the command, but the required. This means that the user can no longer use the former to put variant content in to the TOC as the Manual says.

Instead, the optional argument is used to put an alternative title into the running headers, a better choice.

`\@startsection` Patch a head hook into the basic sectioning command. Treat `\@sect` and `\@ssect` on an equal footing; now their pattern parts are identical.

```

882 \def\@startsection#1#2#3#4#5#6{%
883 \@startsection@hook
884 \ifnoskipsec \leavevmode \fi
885 \par
886 \@tempskipa #4\relax
887 \@afterindenttrue
888 \ifdim \@tempskipa <\z@
889 \@tempskipa -\@tempskipa \@afterindentfalse
890 \fi
891 \if@nobreak
892 \everypar{}%
893 \else
894 \addpenalty\@secpenalty\addvspace\@tempskipa
895 \fi
896 \@ifstar
897 {\@dblarg{\@ssect@ltx{#1}{#2}{#3}{#4}{#5}{#6}}}%
898 {\@dblarg{\@sect@ltx {#1}{#2}{#3}{#4}{#5}{#6}}}%
899 }%
900 \def\@startsection@hook{}%

```

`\@sect` When defining `\@svsec`, do not expand `\@secntformat`. Put brace characters back where they were before David Carlisle got at them (i.e., as if `\@changfrom` had two arguments). Protect the mark mechanism from an undefined meaning. Pass #8 to the TOC instead of #7. Remove `\relax` from the replacement part of `\@svsec`.

The procedure `\@changfrom` and `\@runin@to` can be used to process the argument of the head. The head can define, e.g., `\@changfrom@section`, to do its own processing.

In using `\H@refstepcounter` in place of `\refstepcounter` we rely on either loading before any package that patches the latter, or the convention that the former is the original L^AT_EX procedure.

```

901 \class@info{Repairing broken LaTeX \string\@sect}%
902 \def\@sect@ltx#1#2#3#4#5#6[#7]#8{%
903   \@ifnum{#2}>\c@secnumdepth}{%
904     \def\H@svsec{\phantomsection}%
905     \let\@svsec\@empty
906   }{%
907     \H@refstepcounter{#1}%
908     \def\H@svsec{%
909       \phantomsection
910     }%
911     \protected@edef\@svsec{{#1}}%
912     \@ifundefined{@#1cntformat}{%
913       \prepdef\@svsec\@secntformat
914     }{%
915       \expandafter\prepdef

```

```

916     \expandafter\@svsec
917         \csname @#1cntformat\endcsname
918     }%
919 }%
920 \@tempskipa #5\relax
921 \@ifdim{\@tempskipa>\z@}{%
922     \begingroup
923     \interlinepenalty \@M
924     #6{%
925     \ifundefined{@hangfrom@#1}{\@hang@from}{\csname @hangfrom@#1\endcsname}%
926     {\hskip#3\relax\H@svsec}{\@svsec}{#8}%
927     }%
928     \@@par
929     \endgroup
930     \@ifundefined{#1mark}{\@gobble}{\csname #1mark\endcsname}{#7}%
931     \addcontentsline{toc}{#1}{%
932     \@ifnum{#2>\c@secnumdepth}{%
933     \protect\numberline{}%
934     }{%
935     \protect\numberline{\csname the#1\endcsname}%
936     }%
937     #8}%
938 }{%
939     \def\@svsechd{%
940     #6{%
941     \ifundefined{@runin@to@#1}{\@runin@to}{\csname @runin@to@#1\endcsname}%
942     {\hskip#3\relax\H@svsec}{\@svsec}{#8}%
943     }%
944     \@ifundefined{#1mark}{\@gobble}{\csname #1mark\endcsname}{#7}%
945     \addcontentsline{toc}{#1}{%
946     \@ifnum{#2>\c@secnumdepth}{%
947     \protect\numberline{}%
948     }{%
949     \protect\numberline{\csname the#1\endcsname}%
950     }%
951     #8}%
952     }%
953 }%
954 \@xsect{#5}%
955 }%
956 \def\@hang@from#1#2#3{\@hang@from{#1#2}#3}%
957 \def\@runin@to #1#2#3{#1#2#3}%

```

`\@ssect` Put brace characters back where they were before David Carlisle got at them (as if `\@hangfrom` has two arguments). Possibly set a mark. Make a TOC entry.

Note that, for compatibility with the `hyperref` package, we need to provide the interface required by that package (actually required by `pdfmark.def` and `nameref.sty`), namely the definition of `\@currentlabelname` (but now removed), the insertion of the procedure `\Sectionformat` (but why is this needed?), and the

call to `\phantomsection` (which must precede the call to `\addcontentsline`). We also have to sidestep the patch to `\ssect` in that same file, therefore we use a different control sequence name in the call from `\startsection`.

```

958 \def\@ssect@ltx#1#2#3#4#5#6[#7]#8{%
  Removed \def\@currentlabelname{#8}
959 \def\H@svsec{\phantomsection}%
960 \@tempskipa #5\relax
961 \@ifdim{\@tempskipa>\z@}{%
962   \begingroup
963   \interlinepenalty \@M
964   #6{%
965     \ifundefined{@hangfroms@#1}{\@hang@froms}{\csname @hangfroms@#1\endcsname}%
  Removed {\hskip#3\relax\H@svsec}{\Sectionformat{#8}{#1}}
966     {\hskip#3\relax\H@svsec}{#8}%
967   }%
968   \@@par
969   \endgroup
970   \@ifundefined{#1smark}{\@gobble}{\csname #1smark\endcsname}{#7}%
971   \addcontentsline{toc}{#1}{\protect\numberline{#8}}%
972 }{%
973   \def\@svsechd{%
974     #6{%
975       \ifundefined{@runin@tos@#1}{@runin@tos}{\csname @runin@tos@#1\endcsname}%
  Removed {\hskip#3\relax\H@svsec}{\Sectionformat{#8}{#1}}
976     {\hskip#3\relax\H@svsec}{#8}%
977   }%
978   \@ifundefined{#1smark}{\@gobble}{\csname #1smark\endcsname}{#7}%
979   \addcontentsline{toc}{#1}{\protect\numberline{#8}}%
980 }%
981 }%
982 \@xsect{#5}%
983 }%
984 \def\@hang@froms#1#2{#1#2}%
985 \def\@runin@tos #1#2{#1#2}%

```

`\init@hyperref` Document classes that incorporate this package will be `hyperref-savvy`. (To accomplish this, we ensure that `\hyperanchor` and `\hyper@last` are both defined.) Being `hyperref-savvy` levels some requirements on us, but the benefits are many.

One is that the TOC will not get amnesia and require a full set of three typesetting runs before its formatting is stable. Instead, only two runs are required: the first updates the auxiliary file, the second the TOC. However, the formatting of the document does not change.

Another aspect of being `hyperref-savvy` is that the syntax of commands in the `.aux` file will not change if `hyperref` is turned on or off.

Note that `\hyper@anchorstart` and `\hyper@anchorend` constitute the programming interface for a hypertext anchor (the target of a hypertext link); `\hyper@linkstart` and `\hyper@linkend` are the interface for a hypertext link.

```

986 \def\init@hyperref{%
987 \providecommand\phantomsection{}%
988 \providecommand\hyper@makecurrent[1]{}%
989 \providecommand\Hy@raisedlink[1]{}%
990 \providecommand\hyper@anchorstart[1]{}%
991 \providecommand\hyper@anchorend{}%
992 \providecommand\hyper@linkstart[2]{}%
993 \providecommand\hyper@linkend{}%
994 \providecommand\@currentHref{}%
995 }%
996 \let\H@refstepcounter\refstepcounter
997 \appdef\document@inithook{%
998 \init@hyperref
999 }%

```

`\sec@upcase` Upper case for sections (optional upper case items). These are created so that some headings can be toggled between mixed case and upper case readily. Headings that might be changed can be wrapped in the style file in `\sec@upcase{<text>}` constructs; the expansion of `\sec@upcase` is controlled here. It is `\relax` by default (mixed case heads), and can easily be changed to `\uppercase` if desired. If mixed-case headings are wanted by the editor, authors *must* supply mixed case text, although this is what authors should be doing anyway. (Mixed can be converted to upper, but the reverse transformation cannot be automated.)

The following setting gives the L^AT_EX default.

```
1000 \def\sec@upcase#1{\relax{#1}}%
```

6.14 Patch the tabular and array Environments

`\endtabular` We headpatch the begin processing and tailpatch the end processing of the `tabular` and `array` environments. A document class can define these hooks as needed.

We proceed with care to make further patches to support tabulars that break over pages. Our patches will not necessarily be effective for other packages that replace the L^AT_EX `array` and `tabular` environments. I know of none that do so.

```

1001 \appdef\document@inithook{%
1002 \ifpackageloaded{array}{\switch@array}{\switch@tabular}%
1003 \prepdef\endtabular{\endtabular@hook}%
1004 \@provide\endtabular@hook{}%
1005 \prepdef\endarray{\endarray@hook}%
1006 \@provide\endarray@hook{}%
1007 \providecommand\array@hook{}%

```

Install, effectively, a head patch to `\tabular`. In order to avoid interference from, e.g., the `array` package, we must perform this patch only *after* packages load.

```

1008 \prepdef\@tabular{\tabular@hook}%
1009 \@provide\tabular@hook{}%
1010 }%

```


`\switch@tabular` The two procedures `\switch@tabular` and `\switch@array` apply needed patches to the various tabular procedures, the former applying to the L^AT_EX kernel, the latter to the required `array` package (and to the number of other required packages that load it).

```

1011 \def\switch@tabular{%
1012 \let\@array@sw\@array@sw@array
1013 \@ifx{\@array\@array@LaTeX}{%
1014 \@ifx{\multicolumn\multicolumn@LaTeX}{%
1015 \@ifx{\@tabular\@tabular@LaTeX}{%
1016 \@ifx{\@tabarray\@tabarray@LaTeX}{%
1017 \@ifx{\array\array@LaTeX}{%
1018 \@ifx{\endarray\endarray@LaTeX}{%
1019 \@ifx{\endtabular\endtabular@LaTeX}{%
1020 \@ifx{\@mkpream\@mkpream@LaTeX}{%
1021 \@ifx{\@addamp\@addamp@LaTeX}{%
1022 \@ifx{\@arrayacol\@arrayacol@LaTeX}{%
1023 \@ifx{\@tabacol\@tabacol@LaTeX}{%
1024 \@ifx{\@arrayclassz\@arrayclassz@LaTeX}{%
1025 \@ifx{\@tabclassiv\@tabclassiv@LaTeX}{%
1026 \@ifx{\@arrayclassiv\@arrayclassiv@LaTeX}{%
1027 \@ifx{\@tabclassz\@tabclassz@LaTeX}{%
1028 \@ifx{\@classv\@classv@LaTeX}{%
1029 \@ifx{\hline\hline@LaTeX}{%
1030 \@ifx{\@tabularcr\@tabularcr@LaTeX}{%
1031 \@ifx{\@xtabularcr\@xtabularcr@LaTeX}{%
1032 \@ifx{\@xargarraycr\@xargarraycr@LaTeX}{%
1033 \@ifx{\@yargarraycr\@yargarraycr@LaTeX}{%
1034 \true@sw
1035 }{%
1036 \false@sw
1037 }%
1038 }{%
1039 \false@sw
1040 }%
1041 }{%
1042 \false@sw
1043 }%
1044 }{%
1045 \false@sw
1046 }%
1047 }{%
1048 \false@sw
1049 }%
1050 }{%
1051 \false@sw
1052 }%
1053 }{%
1054 \false@sw
1055 }%

```

```

1056         }{%
1057         \false@sw
1058         }%
1059     }{%
1060     \false@sw
1061     }%
1062 }{%
1063 \false@sw
1064 }%
1065 }{%
1066 \false@sw
1067 }%
1068 }{%
1069 \false@sw
1070 }%
1071 }{%
1072 \false@sw
1073 }%
1074 }{%
1075 \false@sw
1076 }%
1077 }{%
1078 \false@sw
1079 }%
1080 }{%
1081 \false@sw
1082 }%
1083 }{%
1084 \false@sw
1085 }%
1086 }{%
1087 \false@sw
1088 }%
1089 }{%
1090 \false@sw
1091 }%
1092 }{%
1093 \false@sw
1094 }%
1095 }{%
1096 \false@sw
1097 }%
1098 }%
1099 \class@info{Patching LaTeX tabular.}%
1100 }{%
1101 \class@info{Unrecognized LaTeX tabular. Please update this document class! (Proceeding with f
1102 }%
1103 \let\@array\@array@ltx
1104 \let\multicolumn\multicolumn@ltx
1105 \let\@tabular\@tabular@ltx

```

```

1106 \let\@tabarray\@tabarray@ltx
1107 \let\array\array@ltx
1108 \let\endarray\endarray@ltx
1109 \let\endtabular\endtabular@ltx
1110 \let\@mkpream\@mkpream@ltx
1111 \let\@addamp\@addamp@ltx
1112 \let\@arrayacol\@arrayacol@ltx
1113 \let\@tabacol\@tabacol@ltx
1114 \let\@arrayclassz\@arrayclassz@ltx
1115 \let\@tabclassiv\@tabclassiv@ltx
1116 \let\@arrayclassiv\@arrayclassiv@ltx
1117 \let\@tabclassz\@tabclassz@ltx
1118 \let\@classv\@classv@ltx
1119 \let\hline\hline@ltx
1120 \let\@tabularcr\@tabularcr@ltx
1121 \let\@xtabularcr\@xtabularcr@ltx
1122 \let\@xargarraycr\@xargarraycr@ltx
1123 \let\@yargarraycr\@yargarraycr@ltx
1124 }%

1125 \def\switch@array{%
1126 \ifpackageloaded{colortbl}{\let\switch@array@info\colortbl@message}{\let\switch@array@info\ar
1127 \let\@array@sw\@array@sw@LaTeX
1128 \@ifx{\@array\@array@array}{%
1129 \@ifx{\@tabular\@tabular@array}{%
1130 \@ifx{\@tabarray\@tabarray@array}{%
1131 \@ifx{\array\array@array}{%
1132 \@ifx{\endarray\endarray@array}{%
1133 \@ifx{\endtabular\endtabular@array}{%
1134 \@ifx{\@mkpream\@mkpream@array}{%
1135 \@ifx{\@classx\@classx@array}{%
1136 \@ifx{\insert@column\insert@column@array}{%
1137 \@ifx{\@arraycr\@arraycr@array}{%
1138 \@ifx{\@xarraycr\@xarraycr@array}{%
1139 \@ifx{\@xargarraycr\@xargarraycr@array}{%
1140 \@ifx{\@yargarraycr\@yargarraycr@array}{%
1141 \true@sw
1142 }{%
1143 \false@sw
1144 }%
1145 }{%
1146 \false@sw
1147 }%
1148 }{%
1149 \false@sw
1150 }%
1151 }{%
1152 \false@sw
1153 }%
1154 }{%

```

```

1155         \false@sw
1156     }%
1157 }{%
1158     \false@sw
1159     }%
1160 }{%
1161     \false@sw
1162     }%
1163 }{%
1164     \false@sw
1165     }%
1166 }{%
1167     \false@sw
1168     }%
1169 }{%
1170     \false@sw
1171     }%
1172 }{%
1173     \false@sw
1174     }%
1175 }{%
1176     \false@sw
1177     }%
1178 }{%
1179     \false@sw
1180 }{%
1181 \class@info{Patching array package.}%
1182 }{%
1183 \switch@array@info
1184 }%
1185 \let\@array    \@array@array@new
1186 \let\@@array   \@array % Così fan tutti
1187 \let\@tabular  \@tabular@array@new
1188 \let\@tabarray \@tabarray@array@new
1189 \let\@array    \@array@array@new
1190 \let\@endarray \@endarray@array@new
1191 \let\@endtabular\@endtabular@array@new
1192 \let\@mkpream  \@mkpream@array@new
1193 \let\@classx   \@classx@array@new
1194 \let\@arrayacol\@arrayacol@ltx
1195 \let\@tabacol  \@tabacol@ltx
1196 \let\@insert@column\@insert@column@array@new
1197 \expandafter\let\csname endtabular*\endcsname\endtabular % Così fan tutti
1198 \let\@arraycr  \@arraycr@new
1199 \let\@xarraycr \@xarraycr@new
1200 \let\@xargarraycr\@xargarraycr@new
1201 \let\@yargarraycr\@yargarraycr@new
1202 }%
1203 \def\array@message{%
1204 \class@info{Unrecognized array package. Please update this document class! (Proceeding with fi

```

```

1205 }%
1206 \def\colortbl@message{%
1207 \class@info{colortbl package is loaded. (Proceeding with fingers crossed.)}%
1208 }%

```

`\@array@sw` The Boolean `\@array@sw` must be different depending on whether the `array` package is loaded.

```

1209 \def\@array@sw@LaTeX{\@ifx{\@tabularcr}}%
1210 \def\@array@sw@array{\@ifx{\d@llarbegin\beginingroup}}%

```

`\@tabular` We provide the old versions of `\@tabular` along with the respective new versions. The change here is to avoid committing to LR mode. That will be done later (as late as possible, naturally).

Compatibility note: I had done `\let \col@sep \@undefined` here, but this was not compatible with `colortbl`. I have removed that statement.

```

1211 \def\@tabular@LaTeX{%
1212 \leavevmode
1213 \hbox\bgroup$%
1214 \let\@acol\@tabacol
1215 \let\@classz\@tabclassz
1216 \let\@classiv\@tabclassiv
1217 \let\@tabularcr
1218 \@tabarray
1219 }%
1220 \def\@tabular@ltx{%
1221 \let\@acoll\@tabacoll
1222 \let\@acolr\@tabacolr
1223 \let\@acol\@tabacol
1224 \let\@classz\@tabclassz
1225 \let\@classiv\@tabclassiv
1226 \let\@tabularcr
1227 \@tabarray
1228 }%
1229 \def\@tabular@array{%
1230 \leavevmode
1231 \hbox\bgroup$%
1232 \col@sep\tabcolsep
1233 \let\d@llarbegin\beginingroup
1234 \let\d@llarend\endgroup
1235 \@tabarray
1236 }%
1237 \def\@tabular@array@new{%
1238 \let\@acoll\@tabacoll
1239 \let\@acolr\@tabacolr
1240 \let\@acol\@tabacol
1241 \let\d@llarbegin\beginingroup
1242 \let\d@llarend\endgroup
1243 \let\col@sep\@undefined
1244 \@tabarray

```

1244 }%

`\@tabarray` Here we provide old and new versions of the `\@tabarray` procedure. The change here is to parametrize the default vertical alignment, which is 'c' in standard L^AT_EX. Under some circumstances, we want to change this to, say, 'v'.

FIXME: must decouple `array` and `tabular`. Done (it seems).

Note on `colortbl`: this package head-patches `\@tabarray` with its own command `\CT@start`, and tails onto `\endarray` with `\CT@end`. It fortuitously does the former at `\AtBeginDocument` time, and, fortuitously, we do not patch `\endarray`, which it overwrites.

```
1245 \def\@tabarray@LaTeX{%
1246 \m@th@ifnextchar[\@array{\@array[c]}%
1247 }%
1248 \def\@tabarray@ltx{%
1249 \m@th@ifnextchar[\@array{\expandafter\@array\expandafter[ $\array@default$ ]}%
1250 }%
1251 \def\@tabarray@array{%
1252 \@ifnextchar[{\@@array}{\@@array[c]}%
1253 }%
1254 \def\@tabarray@array@new{%
1255 \@ifnextchar[{\@@array}{\expandafter\@@array\expandafter[ $\array@default$ ]}%
1256 }%
```

`\@tabularcr` We provide for the `\@` command within `tabular` to provide control over page breaking, just the same as that of `eqnarray`.

`\@tabularcr` The count register `\intertabularlinepenalty` is similar to `\interdisplaylinepenalty`: it is the penalty associated with each row of a `tabular`. When it is set to `\@M`, the `tabular` will cleave together.

`\@xargarraycr` The count register `\@tbpen` is similar to `\@eqpen`: it memorizes the penalty to use after the current `tabular` row. If the `\@` command is in its star form, then `\@xarraycr` `\@eqpen` is set to `\@M`.

We append code to `\samepage` so that a `tabular` within its scope will cleave together.

We keep the standard definition of `\@tabularcr` in `\@tabularcr@LaTeX` for reference, and provide a new definition that works like `\@eqnocr`: it sets `\@tbpen` to `\@M` if the star was given.

We also provide new versions of `\@xtabularcr`, `\@xargarraycr`, and `\@yargarraycr`, all of which invoke `\@tbpen`.

The `\switch@tabular` procedure switches in the new definitions.

```
1257 \newcount\intertabularlinepenalty
1258 \intertabularlinepenalty=100
1259 \newcount\@tbpen
1260 \appdef\samepage{\intertabularlinepenalty\@M}%
1261 \def\@tabularcr@LaTeX{\ifnum 0='}\fi \ifstar \@xtabularcr \@xarraycr}%
1262 \def\@tabularcr@ltx{\ifnum 0='}\fi \ifstar {\global \@tbpen \@M \@xtabularcr }{\global \@tbpen \@M \@xarraycr }%
1263 \def\@xtabularcr@LaTeX{\ifnextchar [\@argtabularcr {\ifnum 0='{ $\fi$  } $\cr$  }%
1264 \def\@xtabularcr@ltx{\ifnextchar [\@argtabularcr {\ifnum 0='{ $\fi$  } $\cr$  \noalign {\penalty \@tbpen }%
1265 \def\@xargarraycr@LaTeX#1{\@tempdima #1\advance \@tempdima \dp \@arstrutbox \vrule \@height \z@
```

```

1266 \def\@xargarraycr<ltx#1{\@tempdima #1\advance \@tempdima \dp \@arstrutbox \vrule \@height \z@ \
1267 \def\@yargarraycr@LaTeX#1{\cr \noalign {\vskip #1}}%
1268 \def\@yargarraycr<ltx#1{\cr \noalign {\penalty \@tbpen \vskip #1}}%
    If the array package has been loaded, we must alter the meanings of
    \@arraycr, \@xarraycr, \@xargarraycr, and \@yargarraycr. In this case, it
    is \switch@array that switches in the new definitions.
1269 \def\@arraycr@array{%
1270 \relax
1271 \iffalse{\fi\ifnum 0='}\fi
1272 \@ifstar \@xarraycr \@xarraycr
1273 }%
1274 \def\@arraycr@new{%
1275 \relax
1276 \iffalse{\fi\ifnum 0='}\fi
1277 \@ifstar {\global \@tbpen \@M \@xarraycr }{\global \@tbpen \intertabularlinepenalty \@xarraycr}
1278 }%
1279 \def\@xarraycr@array{%
1280 \@ifnextchar [%]
1281 \@argarraycr {\ifnum 0='}\fi\cr}%
1282 }%
1283 \def\@xarraycr@new{%
1284 \@ifnextchar [%]
1285 \@argarraycr {\ifnum 0='}\fi\cr \noalign {\penalty \@tbpen }}%
1286 }%
1287 \def\@xargarraycr@array#1{%
1288 \unskip
1289 \@tempdima #1\advance\@tempdima \dp\@arstrutbox
1290 \vrule \@depth\@tempdima \@width\z@
1291 \cr
1292 }%
1293 \def\@xargarraycr@new#1{%
1294 \unskip
1295 \@tempdima #1\advance\@tempdima \dp\@arstrutbox
1296 \vrule \@depth\@tempdima \@width\z@
1297 \cr
1298 \noalign {\penalty \@tbpen }%
1299 }%
1300 \def\@yargarraycr@array#1{%
1301 \cr
1302 \noalign{\vskip #1}%
1303 }%
1304 \def\@yargarraycr@new#1{%
1305 \cr
1306 \noalign{\penalty \@tbpen \vskip #1}%
1307 }%

```

`\array` We provide old and new versions of the `\array` procedure for both L^AT_EX and the `array` package. The change here is to accomodate the new procedures that will be called for the array boundaries, even though at present they are not special. A

thought: here is where matrices can be readily accomodated.

```

1308 \def\array@LaTeX{%
1309 \let\@acol\@arrayacol
1310 \let\@classz\@arrayclassz
1311 \let\@classiv\@arrayclassiv
1312 \let\@arraycr
1313 \let\@halignto\@empty
1314 \@tabarray
1315 }%
1316 \def\array@ltx{%
1317 \@ifmode{}{\@badmath$}%
1318 \let\@acoll\@arrayacol
1319 \let\@acolr\@arrayacol
1320 \let\@acol\@arrayacol
1321 \let\@classz\@arrayclassz
1322 \let\@classiv\@arrayclassiv
1323 \let\@arraycr
1324 \let\@halignto\@empty
1325 \@tabarray
1326 }%
1327 \def\array@array{%
1328 \col@sep\arraycolsep
1329 \def\d@llarbegin{$}\let\d@llarend\d@llarbegin\gdef\@halignto{}%
1330 \@tabarray
1331 }
1332 \def\array@array@new{%
1333 \@ifmode{}{\@badmath$}%
1334 \let\@acoll\@arrayacol
1335 \let\@acolr\@arrayacol
1336 \let\@acol\@arrayacol
    Removed: \let\col@sep\@undefined
1337 \def\d@llarbegin{$}%
1338 \let\d@llarend\d@llarbegin
1339 \gdef\@halignto{}%
1340 \@tabarray
1341 }%

```

`\@array` Here we provide old and new versions of `\@array`. The change here is to provide a convenient, flexible, and extensible mechanism for new vertical alignment options.

Instead of testing the optional argument with `\if`, we use a dispatcher based on `\csname`.

We also refrain from using `\ialign`, which would set the `\tabskip` to the wrong value.

Finally, the procedure to set the `\@arstrutbox` is broken out so that it can be patched.

```

1342 \def\@array@LaTeX[#1]#2{%
1343 \if #1t\vtop \else \if#1b\ vbox \else \vcenter \fi\fi
1344 \bgroup

```



```

1345 \setbox\@arstrutbox\hbox{%
1346   \vrule \@height\arraystretch\ht\strutbox
1347         \@depth\arraystretch \dp\strutbox
1348         \@width\z@}%
1349 \@mkpream{#2}%
1350 \edef\@preamble{%
1351   \ialign \noexpand\@halignto
1352     \bgroup \@arstrut \@preamble \tabskip\z@skip \cr}%
1353 \let\@startpbox\@startpbox \let\@endpbox\@endpbox
1354 \let\tabularnewline\%
1355   \let\par\@empty
1356   \let\@sharp##%
1357   \set@typeset@protect
1358   \lineskip\z@skip\baselineskip\z@skip
1359   \ifhmode \@preamerr\z@ \@@par\fi
1360   \@preamble
1361 }%
1362 \def\@array@ltx[#1]#2{%
1363 \@nameuse{\@array@align@#1}%
1364 \set@arstrutbox
1365 \mkpream{#2}%
1366 \prepdef\@preamble{%
1367   \tabskip\tabmid@skip
1368   \@arstrut
1369 }%
1370 \appdef\@preamble{%
1371   \tabskip\tabright@skip
1372   \cr
1373   \array@row@pre
1374 }%
1375 % \let\@startpbox\@startpbox
1376 % \let\@endpbox\@endpbox
1377 \let\tabularnewline\%
1378 \let\par\@empty
1379 \let\@sharp##%
1380 \set@typeset@protect
1381 \lineskip\z@skip\baselineskip\z@skip
1382 \tabskip\tableft@skip\relax
1383 \ifhmode \@preamerr\z@ \@@par\fi
1384 \everycr{}}%
1385 \expandafter\halign\expandafter\@halignto\expandafter\bgroup\@preamble
1386 }%
1387 %
1388 \def\set@arstrutbox{%
1389 \setbox\@arstrutbox\hbox{%
1390   \vrule \@height\arraystretch\ht\strutbox
1391         \@depth\arraystretch \dp\strutbox
1392         \@width\z@
1393 }%
1394 }%

```

\@array@array

```
1395 \def\@array@array[#1]#2{%
1396   \@tempdima \ht \strutbox
1397   \advance \@tempdima by\extrarowheight
1398   \setbox \@arstrutbox \hbox{\vrule
1399     \@height \arraystretch \@tempdima
1400     \@depth \arraystretch \dp \strutbox
1401     \@width \z@}%
1402   \begingroup
1403   \@mkpream{#2}%
1404   \xdef\@preamble{\noexpand \ialign \@halignto
1405     \bgroup \@arstrut \@preamble
1406     \tabskip \z@ \cr}%
1407   \endgroup
1408   \@arrayleft
1409   \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi \fi
1410   \bgroup
1411   \let \@sharp ##\let \protect \relax
1412   \lineskip \z@
1413   \baselineskip \z@
1414   \m@th
1415   \let\\\@arraycr \let\tabularnewline\\\let\par\@empty \@preamble
1416 }%
1417 \def\@array@array@new[#1]#2{%
1418   \@tempdima\ht\strutbox
1419   \advance\@tempdima by\extrarowheight
1420   \setbox\@arstrutbox\hbox{%
1421     \vrule \@height\arraystretch\@tempdima
1422     \@depth \arraystretch\dp\strutbox
1423     \@width \z@
1424   }%
1425   \begingroup
1426     \@mkpream{#2}%
1427     \xdef\@preamble{\@preamble}%
1428   \endgroup
1429   \prepdef\@preamble{%
1430     \tabskip\tabmid@skip
1431     \@arstrut
1432   }%
1433   \appdef\@preamble{%
1434     \tabskip\tabright@skip
1435     \cr
1436     \array@row@pre
1437   }%
1438   \@arrayleft
1439   \@nameuse{\@array@align@#1}%
1440   \m@th
1441   \let\\\@arraycr
1442   \let\tabularnewline\%
```

```

1443 \let\par\@empty
1444 \let\@sharp##%
1445 \set@typeset@protect
1446 \lineskip\z@\baselineskip\z@
1447 \tabskip\tableft@skip
1448 \everycr{}%
1449 \expandafter\halign\expandafter\@halignto\expandafter\bgroup\@preamble
1450 }%

```

`\endarray` Here we provide old and new versions of `\endarray`. The change here is to use a single procedure to close out any array-like structure, namely `\endarray@ltx`. It merely closes out the `\halign`.

```

1451 \def\endarray@LaTeX{%
1452 \crr\egroup\egroup
1453 }%
1454 \def\endarray@ltx{%
1455 \crr\array@row@pst\egroup\egroup
1456 }%
1457 \def\endarray@array{%
1458 \crr \egroup \egroup \@arrayright \gdef\@preamble{}%
1459 }%
1460 \def\endarray@array@new{%
1461 \crr\array@row@pst\egroup\egroup % Same as \endarray@ltx
1462 \@arrayright
1463 \global\let\@preamble\@empty
1464 }%

```

`\endtabular`

```

1465 \def\endtabular@LaTeX{%
1466 \crr\egroup\egroup $\egroup
1467 }%
1468 \def\endtabular@ltx{%
1469 \endarray
1470 }%
1471 \def\endtabular@array{%
1472 \endarray $\egroup
1473 }%
1474 \def\endtabular@array@new{%
1475 \endarray
1476 }%

```

`endtabular*` Here we provide a proper definition for the star-form of `\end{endtabular}`. It is one of the enduring curiosities that the \LaTeX kernel continues to use dangerously and inappropriately “optimized” definitions for such commands.

```

1477 \@namedef{endtabular*}{\endtabular}%

```

`\multicolumn`

```

1478 \long\def\multicolumn@LaTeX#1#2#3{%
1479 \multispan{#1}\begingroup

```

```

1480 \mkpream{#2}%
1481 \def\@sharp{#3}\set@typeset@protect
1482 \let\@startpbox\@startpbox\let\@endpbox\@endpbox
1483 \@arstrut \@preamble\hbox{}\endgroup\ignorespaces
1484 }%
1485 \long\def\multicolumn@ltx#1#2#3{%
1486 \multispan{#1}%
1487 \begingroup
1488 \mkpream{#2}%
1489 \def\@sharp{#3}%
1490 \set@typeset@protect
1491 %\let\@startpbox\@startpbox\let\@endpbox\@endpbox
1492 \@arstrut
1493 \@preamble
1494 \hbox{}%
1495 \endgroup
1496 \ignorespaces
1497 }%

```

`\array@align@` Here are the various procedures for the vertical alignment options. The change
`\array@default` from standard L^AT_EX is that we do not go into math mode in every case: only when required by `\vcenter`. Also, we use `\aftergroup` to close out the boxes and modes we have started. It requires only that each procedure issue exactly one unmatched `\bgroup`.

We establish here the default vertical alignment.

```

1498 \def\@array@align@t{\leavevmode\vtop\bgroup}%
1499 \def\@array@align@b{\leavevmode\vbox\bgroup}%
1500 \def\@array@align@c{\leavevmode\@ifmmode{\vcenter\bgroup}{$\vcenter\bgroup\aftergroup$\aftergroup}
1501 \def\@array@align@v{%
1502 \@ifmmode{%
1503 \badmath
1504 \vcenter\bgroup
1505 }{%
1506 \@ifinner{%
1507 $\vcenter\bgroup\aftergroup$
1508 }{%
1509 \@par\bgroup
1510 }%
1511 }%
1512 }%
1513 \def\array@default{c}%

```

`\array@row@pre` The procedure `\array@row@rst` reestablishes a default context for an alignment,
`\array@row@pst` so that they can be nested. Any environment or procedure that alters the way
`\array@row@rst` alignments are formatted must patch this procedure to restore from that alteration.
To start things off, we equate `\@array@align@v` to `\@array@align@c`, because it does not make sense to do the former in any context other than the MVL or in a list that will be unboxed onto the MVL.

```

1514 \def\array@row@rst{%

```

```

1515 \let\array@align@v\array@align@
1516 }%
1517 \def\array@row@pre{%
1518 \def\array@row@pst{%

\toprule Default definitions for \toprule, \colrule, \botrule
\colrule 1519 \newcommand\toprule{\tab@rule{\column@font}{\column@fil}{\frstrut}}%
\botrule 1520 \newcommand\colrule{\unskip\lrstrut\\\tab@rule{\body@font}{\frstrut}}%
1521 \newcommand\botrule{\unskip\lrstrut\\\noalign{\hline@rule}{}}%

\hline
1522 \def\hline@LaTeX{%
1523 \noalign{\ifnum0=}\fi\hrule \@height \arrayrulewidth \futurelet
1524 \reserved@a\@xhline
1525 }%
1526 \def\hline@ltx{%
1527 \noalign{%
1528 \ifnum0=}\fi
1529 \hline@rule
1530 \futurelet\reserved@a\@xhline
1531 % \noalign ended in \@xhline
1532 }%
1533 \def\@xhline@unneeded{%
1534 \say\reserved@a
1535 \ifx\reserved@a\hline
1536 \vskip\doublerulesep
1537 \vskip-\arrayrulewidth
1538 \fi
1539 \ifnum0=}\fi}%
1540 }%
1541 \def\tab@rule#1#2#3{%
1542 \crr
1543 \noalign{%
1544 \hline@rule
1545 \gdef\@arstrut@hook{%
1546 \global\let\@arstrut@hook\@empty
1547 #3%
1548 }%
1549 \gdef\cell@font{#1}%
1550 \gdef\cell@fil{#2}%
1551 }%
1552 }%
1553 \def\column@font{%
1554 \def\column@fil{%
1555 \def\body@font{%
1556 \def\cell@font{%
1557 \def\frstrut{%
1558 \def\lrstrut{%

```

```

\@arstrut@hline The procedure \@arstrut@hline is substantially the same as \@arstrut, except
\@arstrut@org the strut copied in is \@arstrutbox@hline instead of \@arstrutbox.
\@arstrut@hook The procedure \@arstrut@hook is redefined in \tab@rule!
\@arstrutbox@hline The register \@arstrutbox@hline.
\set@arstrutbox We append to \set@arstrutbox the code necessary to set a strut following an
\hline@rule \hline.
                The procedure \hline@rule lays down a rule, and changes the meaning of
                \@arstrut so that the next line will be correctly strutted.
                The \@arstrut@hline@clnc is a klotch, a magic number.
1559 \def\@arstrut@hline{%
1560 \relax
1561 \ifmmode{\copy}{\unhcopy}\@arstrutbox@hline
1562 \@arstrut@hook
1563 }%
1564 %
1565 \let\@arstrut@org\@arstrut
1566 \def\@arstrut@hook{%
1567 \global\let\@arstrut\@arstrut@org
1568 }%
1569 %
1570 \newbox\@arstrutbox@hline
1571 \appdef\set@arstrutbox{%
1572 \setbox\@arstrutbox@hline\hbox{%
1573 \setbox\z@\hbox{\$0^{0}_f}\$}%
1574 \dimen@ht\z@\advance\dimen@\@arstrut@hline@clnc
1575 \ifdim\dimen@<\arraystretch\ht\strutbox}{\dimen@=\arraystretch\ht\strutbox}{}%
1576 \vrule \@height\dimen@
1577 \depth\arraystretch \dp\strutbox
1578 \width\z@
1579 }%
1580 }%
1581 %
1582 \def\hline@rule{%
1583 \hrule \@height \arrayrulewidth
1584 \global\let\@arstrut\@arstrut@hline
1585 }%
1586 \def\@arstrut@hline@clnc{2\p@}% % Klotch: magic number

\tableleft@skip
1587 \def\tableleft@skip{\z@skip}%
1588 \def\tabmid@skip{\z@skip}\@flushglue
1589 \def\tabright@skip{\z@skip}%
1590 \def\tableleftsep{\tabcolsep}%
1591 \def\tabmidsep{\tabcolsep}%
1592 \def\tabrightsep{\tabcolsep}%
1593 \def\cell@fil{}%
1594 \def\pbox@hook{}%

\@arstrut

```

```

1595 \appdef\@arstrut{\@arstrut@hook}%
1596 \let\@arstrut@hook\@empty
1597 \def\@addtopreamble{\appdef\@preamble}%

```

\@mkpream

```

1598 \def\@mkpream@LaTeX#1{%
1599   \@firststamptrue\@lastchclass6
1600   \let\@preamble\@empty
1601   \let\protect\@unexpandable@protect
1602   \let\@sharp\relax
1603   \let\@startpbox\relax\let\@endpbox\relax
1604   \@expast{#1}%
1605   \expandafter\@tfor \expandafter
1606     \@nextchar \expandafter:\expandafter=\reserved@a\do
1607     {\@testpach\@nextchar
1608     \ifcase \@chclass \@classz \or \@classi \or \@classii \or \@classiii
1609     \or \@classiv \or \@classv \fi\@lastchclass\@chclass}%
1610   \ifcase \@lastchclass \@acol
1611     \or \or \@preamerr \@ne\or \@preamerr \tw@\or \or \@acol \fi
1612 }%
1613 \def\@mkpream@ltx#1{%
1614   \@firststamptrue
1615   \@lastchclass6
1616   \let\@preamble\@empty
1617   \let\protect\@unexpandable@protect
1618   \let\@sharp\relax
1619   \let\@startpbox\relax\let\@endpbox\relax
1620   \@expast{#1}%
1621   \expandafter\@tfor\expandafter\@nextchar\expandafter:\expandafter=\reserved@a
1622   \do{%
1623     \expandafter\@testpach\expandafter{\@nextchar}%
1624     \ifcase\@chclass
1625       \@classz
1626       \or
1627       \@classi
1628       \or
1629       \@classii
1630       \or
1631       \@classiii
1632       \or
1633       \@classiv
1634       \or
1635       \@classv
1636       \fi
1637     \@lastchclass\@chclass
1638   }%
1639   \ifcase\@lastchclass
1640     \@acolr % right-hand column
1641   \or
1642   \or

```

```

1643 \@preamerr\@ne
1644 \or
1645 \@preamerr\tw@
1646 \or
1647 \or
1648 \@acolr % right-hand column
1649 \fi
1650 }%

```

`\insert@column`

```

1651 \def\insert@column@array{%
1652   \the@toks \the \@tempcnta
1653   \ignorespaces \@sharp \unskip
1654   \the@toks \the \count@ \relax
1655 }%
1656 \def\insert@column@array@new{%
1657   \the@toks\the\@tempcnta
1658   \array@row@rst\cell@font
1659   \ignorespaces\@sharp\unskip
1660   \the@toks\the\count@
1661   \relax
1662 }%

```

`\@mkpream@relax` The procedure `\@mkpream@relax` participates in a strange and wonderful method of binding the alignment procedure—but only certain parts thereof.

Here is how it works: in \LaTeX , the `array` package, and in the `longtable` package alike, there is a need to create an alignment preamble (using `\@mkpream`) for use by the upcoming `\halign`. Then, in both `array` and `longtable`, \TeX 's `\edef` is used to ‘compile in place’ that alignment preamble.

In the case of `array`, the operation is done in order to pre-expand the use of `*`; in `longtable`, it is to set the widths of the columns.

Now, during this `\edef`, certain control sequence names must *not* be expanded, and those are robustified by `\@mkpream@relax`.

```

1663 \def\@mkpream@relax{%
1664   \let\tableleftsep \relax
1665   \let\abmidsep \relax
1666   \let\abrightsep \relax
1667   \let\array@row@rst\relax
1668   \let\cell@font \relax
1669   \let\@startpbox \relax
1670 }%

```

`\@mkpream` We insert `\@mkpream@relax` at the head of the procedure. The robustifying of `\@startpbox` and `\@endpbox` is taken over by this mechanism. We also invoke `\@acolr` instead of `\@acol` when a right-hand column is at hand.

Note on `colortbl`: this package head-patches `\@mkpream` to robustify a number of its commands during the construction of the alignment preamble. The best we can do is to supplement the `\@mkpream@relax` procedure to perform this action.


```

1671 \def\@mkpream@array#1{%
1672   \gdef\@preamble{ }\@lastchclass 4 \@firstamptrue
1673   \let\@sharp\relax \let\@startpbox\relax \let\@endpbox\relax
1674   \@temptokena{#1}\@tempswatru
1675   \@whiles\if@tempswa\fi{\@tempswafalse\the\NC@list}%
1676   \count@m@ne
1677   \let\the@toks\relax
1678   \prepnext@tok
1679   \expandafter \@tfor \expandafter \@nextchar
1680   \expandafter :\expandafter =\the\@temptokena \do
1681   {\@testpach
1682   \ifcase \@chclass \@classz \or \@classi \or \@classii
1683   \or \save@decl \or \or \@classv \or \@classvi
1684   \or \@classvii \or \@classviii
1685   \or \@classx
1686   \or \@classx \fi
1687   \@lastchclass\@chclass}%
1688   \ifcase\@lastchclass
1689   \@acol \or
1690   \or
1691   \@acol \or
1692   \@preamerr \thr@@ \or
1693   \@preamerr \tw@ \@addtopreamble\@sharp \or
1694   \or
1695   \else \@preamerr \@ne \fi
1696   \def\the@toks{\the@toks}%
1697 }%
1698 \def\@mkpream@array@new#1{%
1699 \gdef\@preamble{ }%
1700 \@lastchclass\@f@ur
1701 \@firstamptrue
1702 \let\@sharp\relax
1703 \@mkpream@relax
1704 %\let\@startpbox\relax\let\@endpbox\relax
1705 \@temptokena{#1}\@tempswatru
1706 \@whiles\if@tempswa\fi{\@tempswafalse\the\NC@list}%
1707 \count@m@ne
1708 \let\the@toks\relax
1709 \prepnext@tok
1710 \expandafter \@tfor \expandafter \@nextchar \expandafter :\expandafter =\the\@temptokena
1711 \do{%
1712   \@testpach
1713   \ifcase\@chclass
1714   \@classz
1715   \or
1716   \@classi
1717   \or
1718   \@classii
1719   \or
1720   \save@decl

```

```

1721 \or
1722 \or
1723 \@classv
1724 \or
1725 \@classvi
1726 \or
1727 \@classvii
1728 \or
1729 \@classviii
1730 \or
1731 \@classx
1732 \or
1733 \@classx
1734 \fi
1735 \@lastchclass\@chclass
1736 }%
1737 \ifcase\@lastchclass
1738 \@acolr % right-hand column
1739 \or
1740 \or
1741 \@acolr % right-hand column
1742 \or
1743 \@preamerr\thr@@
1744 \or
1745 \@preamerr\tw@\@addtopreamble\@sharp
1746 \or
1747 \or
1748 \else
1749 \@preamerr\@ne
1750 \fi
1751 \def\the@toks{\the\toks}%
1752 }%

```

`\@mkpream@relax` David P. Carlisle's `colortbl` package headpatches `\@mkpream` in place during package loading, so it does not know whom it is working on. Let us try to accommodate this package by doing what it would have liked to have done.

Note: it would be far better to break out this mechanism in the `array` package.

```

1753 \appdef\@mkpream@relax{%
1754 \let\CT@setup \relax
1755 \let\CT@color \relax
1756 \let\CT@do@color \relax
1757 \let\color \relax
1758 \let\CT@column@color\relax
1759 \let\CT@row@color \relax
1760 \let\CT@cell@color \relax
1761 }%

```

`\@addamp`

```

1762 \def\@addamp@LaTeX{%

```

```

1763 \if@firstamp\@firstampfalse\else\edef\@preamble{\@preamble &}\fi
1764 }%
1765 \def\@addamp@ltx{%
1766 \if@firstamp\@firstampfalse\else\@addtopreamble{&}\fi
1767 }%

\@arrayacol
1768 \def\@arrayacol@LaTeX{%
1769 \edef\@preamble{\@preamble \hskip \arraycolsep}%
1770 }%
1771 \def\@arrayacol@ltx{%
1772 \@addtopreamble{\hskip\arraycolsep}%
1773 }%

\@tabacol
1774 \def\@tabacolI{%
1775 \@addtopreamble{\hskip\tableleftsep\relax}%
1776 }%
1777 \def\@tabacol@LaTeX{%
1778 \edef\@preamble{\@preamble \hskip \tabcolsep}%
1779 }%
1780 \def\@tabacol@ltx{%
1781 \@addtopreamble{\hskip\abmidsep\relax}%
1782 }%
1783 \def\@tabacolr{%
1784 \@addtopreamble{\hskip\abrightsep\relax}%
1785 }%

\@arrayclassz
1786 \def\@arrayclassz@LaTeX{%
1787 \ifcase \@lastchclass \@acolampacol \or \@ampacol \or
1788 \or \or \@addamp \or
1789 \@acolampacol \or \@firstampfalse \@acol \fi
1790 \edef\@preamble{\@preamble
1791 \ifcase \@chnum
1792 \hfil$\relax\@sharp$\hfil \or $\relax\@sharp$\hfil
1793 \or \hfil$\relax\@sharp$\fi}%
1794 }%
1795 \def\@arrayclassz@ltx{%
1796 \ifcase\@lastchclass
1797 \@acolampacol
1798 \or
1799 \@ampacol
1800 \or
1801 \or
1802 \or
1803 \@addamp
1804 \or
1805 \@acolampacol

```

```

1806 \or
1807 \@firstampfalse\@acol
1808 \fi
1809 \ifcase\@chnum
1810 \@addtopreamble{%
1811   \hfil\array@row@rst$\relax\@sharp$\hfil
1812 }%
1813 \or
1814 \@addtopreamble{%
1815   \array@row@rst$\relax\@sharp$\hfil
1816 }%
1817 \or
1818 \@addtopreamble{%
1819   \hfil\array@row@rst$\relax\@sharp$%
1820 }%
1821 \fi
1822 }%

```

\@tabclassz

```

1823 \def\@tabclassz@LaTeX{%
1824   \ifcase\@lastchclass
1825     \@acolampacol
1826   \or
1827     \@ampacol
1828   \or
1829   \or
1830   \or
1831     \@addamp
1832   \or
1833     \@acolampacol
1834   \or
1835     \@firstampfalse\@acol
1836   \fi
1837   \edef\@preamble{%
1838     \@preamble{%
1839       \ifcase\@chnum
1840         \hfil\ignorespaces\@sharp\unskip\hfil
1841       \or
1842         \hskip1sp\ignorespaces\@sharp\unskip\hfil
1843       \or
1844         \hfil\hskip1sp\ignorespaces\@sharp\unskip
1845       \fi}}%
1846 }%
1847 \def\@tabclassz@ltx{%
1848   \ifcase\@lastchclass
1849     \@acolampacol
1850   \or
1851     \@ampacol
1852   \or
1853   \or

```

```

1854 \or
1855 \@addamp
1856 \or
1857 \@acolampacol
1858 \or
1859 \@firstampfalse\@acoll
1860 \fi
1861 \ifcase\@chnum
1862 \@addtopreamble{%
1863   {\hfil\array@row@rst\cell@font\ignorespaces\@sharp\unskip\hfil}%
1864 }%
1865 \or
1866 \@addtopreamble{%
1867   {\cell@fil\hskip1sp\array@row@rst\cell@font\ignorespaces\@sharp\unskip\hfil}%
1868 }%
1869 \or
1870 \@addtopreamble{%
1871   {\hfil\hskip1sp\array@row@rst\cell@font\ignorespaces\@sharp\unskip\cell@fil}%
1872 }%
1873 \fi
1874 }%

```

\@tabclassiv

```

1875 \def\@tabclassiv@LaTeX{%
1876 \@addtopreamble\@nextchar
1877 }%
1878 \def\@tabclassiv@ltx{%
1879 \expandafter\@addtopreamble\expandafter{\@nextchar}%
1880 }%

```

\@arrayclassiv

```

1881 \def\@arrayclassiv@LaTeX{%
1882 \@addtopreamble{\$\@nextchar$}%
1883 }%
1884 \def\@arrayclassiv@ltx{%
1885 \expandafter\@addtopreamble\expandafter{\expandafter$\@nextchar$}%
1886 }%

```

\@classv

```

1887 \def\@classv@LaTeX{%
1888 \@addtopreamble{\@startpbox{\@nextchar}\ignorespaces
1889 \@sharp\@endpbox}%
1890 }%
1891 \def\@classv@ltx{%
1892 \expandafter\@addtopreamble
1893 \expandafter{%
1894 \expandafter \@startpbox
1895 \expandafter {\@nextchar}%
1896 \pbox@hook\array@row@rst\cell@font\ignorespaces\@sharp\@endpbox

```

```

1897 }%
1898 }%

```

```
\@classx
```

```

1899 \def\@classx@array{%
1900   \ifcase \@lastchclass
1901   \@acolampacol \or
1902   \@addamp \@acol \or
1903   \@acolampacol \or
1904   \or
1905   \@acol \@firstampfalse \or
1906   \@addamp
1907   \fi
1908 }%
1909 \def\@classx@array@new{%
1910   \ifcase \@lastchclass
1911   \@acolampacol
1912   \or
1913   \@addamp \@acol
1914   \or
1915   \@acolampacol
1916   \or
1917   \or
1918   \@firstampfalse\@acoll
1919   \or
1920   \@addamp
1921   \fi
1922 }%

```

6.15 Repair other broken parts of L^AT_EX

```
\@xbitor Expansion part has extraneous space token. Removed.
```

```

1923 \def\@xbitor@LaTeX #1{\@tempcntb \count#1
1924   \ifnum \@tempcnta =\z@
1925   \else
1926     \divide\@tempcntb\@tempcnta
1927     \ifodd\@tempcntb \@testtrue\fi
1928   \fi}%
1929 \def\@xbitor@ltx#1{%
1930   \@tempcntb\count#1\relax
1931   \@ifnum{\@tempcnta=\z@}{-}{%
1932     \divide\@tempcntb\@tempcnta
1933     \@ifodd\@tempcntb{\@testtrue}{-}}%
1934 }%
1935 }%
1936 \@ifx{\@xbitor\@xbitor@LaTeX}{%
1937   \class@info{Repairing broken LaTeX \string\@xbitor}%
1938 }{%
1939   \class@info{Unrecognized LaTeX \string\@xbitor. Please update this document class! (Proceedin

```

```
1940 }%
1941 \let\@xbitor\@xbitor@ltx
```

6.16 Syntax

`\@gobble@opt@one` The `\@gobble@opt@one` command eats up an optional argument and one required argument.

```
1942 \newcommand*\@gobble@opt@one [2] [] {}%
```

6.17 Auto-indented Contents

Facility to automatically determine the proper indentation of the TOC entries.

Note on `hyperref` compatibility: We must respect that `\contentsline` now has a fourth argument. So, instead of trying to override the meaning of `\contentsline`, we use the aux file to remember max values from one run to the next.

In this respect, this package retains compatibility with `hyperref`.

`\starttoc` Install hooks at beginning and end of the TOC processing.

```
1943 \def\starttoc#1{%
1944   \begingroup
1945     \toc@pre
1946     \makeatletter
1947     \@input{\jobname.#1}%
1948     \if@filesw
1949       \expandafter\newwrite\csname tf@#1\endcsname
1950       \immediate\openout \csname tf@#1\endcsname \jobname.#1\relax
1951     \fi
1952     \nobreakfalse
1953     \toc@post
1954   \endgroup
1955 }%
1956 \def\toc@pre{}%
1957 \def\toc@post{}%
```

`\toc@@font` Interface for setting the formatting characteristics of this part of the TOC.

Note: `\toc@@font` is the common font for all auto-sizing toc commands, although this, too, could become a dispatcher.

```
1958 \def\toc@@font{}%
1959 \def\ltxu@dotsep{z}%
```

`\l@section` Interface for determining which TOC elements are automatically indented.

All of the `\l@...` commands simply go through the utility procedure `\l@sections`. The calling convention is to pass the name of self and the name of parent. If you want to exclude any of these from the indentation scheme, simply leave the `\l@...` command undefined.

Note that the parent of “section” is nil, so we have to define a stub.

```

\def\l@section{\l@sections{}{section}}% Implicit #3#4

\def\tocleft@{\z@}%

\def\l@subsection{\l@sections{section}{subsection}}% Implicit #3#4

\def\l@subsubsection{\l@sections{subsection}{subsubsection}}% Implicit #3#4

\def\l@paragraph{\l@sections{subsubsection}{paragraph}}% Implicit #3#4

\def\l@subparagraph#1#2{\l@sections{paragraph}{subparagraph}}% Implicit #3#4

```

Glom some \dimen registers.

```

1960 \let\tocdim@section      \leftmargini
1961 \let\tocdim@subsection  \leftmarginii
1962 \let\tocdim@subsubsection \leftmarginiii
1963 \let\tocdim@paragraph   \leftmarginiv
1964 \let\tocdim@appendix    \leftmarginv
1965 \let\tocdim@pagenum     \leftmarginvi

```

`\toc@pre@auto` We patch `\@starttoc` to: 1) before TOC processing, initialize the max registers
`\toc@post@auto` and set the needed dimensions from the values stored in the auxiliary file, and 2)
after TOC processing, store out those max register values into the auxiliary file.

Note that the font is set here: all other TOC entries must override these font settings.

To activate this override of the standard L^AT_EX processing, the substyle does:

`\let\toc@pre\toc@pre@auto` and `\let\toc@post\toc@post@auto`.

```

1966 \def\toc@pre@auto{%
1967   \toc@@font
1968   \@tempdima\z@
1969   \toc@setindent\@tempdima{section}%
1970   \toc@setindent\@tempdima{subsection}%
1971   \toc@setindent\@tempdima{subsubsection}%
1972   \toc@setindent\@tempdima{paragraph}%
1973   \toc@letdimen{appendix}%
1974   \toc@letdimen{pagenum}%
1975 }%
1976 \def\toc@post@auto{%
1977   \if@filesw
1978     \begingroup
1979       \toc@writedimen{section}%
1980       \toc@writedimen{subsection}%
1981       \toc@writedimen{subsubsection}%
1982       \toc@writedimen{paragraph}%
1983       \toc@writedimen{appendix}%
1984       \toc@writedimen{pagenum}%
1985     \endgroup
1986   \fi
1987 }%

```


`\toc@setindent`

```
1988 \def\toc@setindent#1#2{%
1989 \csname tocdim@#2\endcsname\tocdim@min\relax
1990 \@ifundefined{tocmax@#2}{\@namedef{tocmax@#2}{\z@}}{}%
1991 \advance#1\@nameuse{tocmax@#2}\relax
1992 \expandafter\edef\csname tocleft@#2\endcsname{\the#1}%
1993 }%
```

`\toc@letdimen`

```
1994 \def\toc@letdimen#1{%
1995 \csname tocdim@#1\endcsname\tocdim@min\relax
1996 \@ifundefined{tocmax@#1}{\@namedef{tocmax@#1}{\z@}}{}%
1997 \expandafter\let\csname tocleft@#1\expandafter\endcsname\csname tocmax@#1\endcsname
1998 }%
```

`\toc@writedimen`

```
1999 \def\toc@writedimen#1{%
2000 \immediate\write\@auxout{%
2001 \gdef\expandafter\string\csname tocmax@#1\endcsname{%
2002 \expandafter\the\csname tocdim@#1\endcsname
2003 }%
2004 }%
2005 }%
```

`\l@sections` The procedure for formatting the indented TOC entries. We use control sequence names such as `\tocmax@section` and `\tocleft@section`, the former being written to the auxiliary file and the latter only defined for the duration of the TOC processing.

Note that the assignment of `\box\@tempboxa` by `\set@tocdim@pagenum` must endure over the invocation of `#3`: it contains the page number which will be set just before the `\par`.

The arguments:

`#1` superior section

`#2` this section

`#3` content, including possible `\numberline`

`#4` page number

```
2006 \def\l@sections#1#2#3#4{%
2007 \begingroup
2008 \everypar{}%
2009 \set@tocdim@pagenum\@tempboxa{#4}%
2010 \global\@tempdima\csname tocdim@#2\endcsname
2011 \leftskip\csname tocleft@#2\endcsname\relax
2012 \dimen@\csname tocleft@#1\endcsname\relax
2013 \parindent-\leftskip\advance\parindent\dimen@
2014 \rightskip\tocleft@pagenum plus 1fil\relax
```

```

2015 \skip@\parfillskip\parfillskip\z@
2016 \let\numberline\numberline@@sections
2017 \@nameuse{1of#2}%
2018 \ignorespaces#3\unskip\nobreak\hskip\skip@
2019 \hb@xt@\rightskip{\hfil\unhbox\@tempboxa}\hskip-\rightskip\hskip\z@skip

```

By side effect, set the value of, e.g., `\tocdim@section`.

Note that the `\par` must not be executed before the value of `\@tempdima` is expanded (outside the current group). Otherwise, the `lineno.sty` package may interfere (it unfortunately does a global assignment of `\@tempdima`).

```

2020 \expandafter\par
2021 \expandafter\aftergroup\csgname tocdim#2%
2022 \expandafter\endcsgname
2023 \expandafter\endgroup
2024 \the\@tempdima\relax
2025 }%

```

In the call to `\set@tocdim@pagenum`, I am now exposing the use of the particular box register.

```

2026 \def\set@tocdim@pagenum#1#2{%
2027 \setbox#1\hbox{\ignorespaces#2}%
2028 \@ifdim{\tocdim@pagenum<\wd#1}{\global\tocdim@pagenum\wd#1}{}%
2029 }%

```

`\numberline@@sections` The utility procedure for all `\numberline` processing in indented TOC entries. The first argument is self.

We use `\@tempdima` to pass a value around (via global assignment) because `\numberline` executes inside a group if the `hyperref` package is loaded. Would that it were not so!

```

2030 \def\numberline@@sections#1{%
2031 \leavevmode\hb@xt@-\parindent{%
2032 \hfil
2033 \@ifempty{#1}{}{%
2034 \setbox\z@\hbox{#1.\kern\ltxu@dotsep}%
2035 \@ifdim{\@tempdima<\wd\z@}{\global\@tempdima\wd\z@}{}%
2036 \unhbox\z@
2037 }%
2038 }%
2039 \ignorespaces
2040 }%
2041 \def\tocdim@min{\z@}%

```

6.18 Lists

`\list` Using `\parshape` to implement lists was always suspect (can you get behind `\parshape@ne?`) and we now see that it was a mistake all along. Why? Because `\parshape`, like `\hangindent`, achieves its effect via “shifting” the `\hboxes` in a paragraph instead of using `\leftskip` and `\parindent`, which is robust during column balancing.

We introduce the alternative method with a hook into the L^AT_EX kernel procedure `\list`, which is the implementation of all lists.

```

2042 \def\list#1#2{%
2043   \ifnum \@listdepth >5\relax
2044     \@toodeep
2045   \else
2046     \global\advance\@listdepth\@ne
2047   \fi
2048   \rightmargin\z@
2049   \listparindent\z@
2050   \itemindent\z@
2051   \csname @list\romannumeral\the\@listdepth\endcsname
2052   \def\@itemlabel{#1}%
2053   \let\makelabel\@mklab
2054   \@nmblistfalse
2055   #2\relax
2056   \@trivlist
2057   \parskip\parsep
2058   \set@listindent
2059   \ignorespaces
2060 }%
2061 \def\set@listindent@parshape{%
2062   \parindent\listparindent
2063   \advance\@totalleftmargin\leftmargin
2064   \advance\linewidth-\rightmargin
2065   \advance\linewidth-\leftmargin
2066   \parshape\@ne\@totalleftmargin\linewidth
2067 }%
2068 \def\set@listindent@{%
2069   \parindent\listparindent
2070   \advance\@totalleftmargin\leftmargin
2071   \advance\rightskip\rightmargin
2072   \advance\leftskip\@totalleftmargin
2073 }%
2074 \let\set@listindent\set@listindent@parshape

```

6.19 Hypertext capabilities

`\href` We provide support for the `\href`, `\url`, and `\doi` commands. Packages, like `\url` `hyperref`, may override these definitions and provide better semantics.

```

\URL@prefix 2075 \providecommand\href[0]{\begingroup\@sanitize@url\@href}%
\doi 2076 \def\@href#1{\@startlink{#1}\endgroup\@@href}%
\doibase 2077 \def\@@href#1{#1\@endlink}%
2078 \providecommand \url [0]{\begingroup\@sanitize@url \@url }%
2079 \def \@url #1{\endgroup\@href {#1}{\URL@prefix#1}}%
2080 \providecommand \URL@prefix [0]{URL }%
2081 \providecommand\doi[0]{\begingroup\@sanitize@url\@doi}%
2082 \def\@doi#1{\endgroup\@startlink{\doibase#1}doi:\discretionary {}{}{}#1\@endlink }%
2083 \providecommand \doibase [0]{http://dx.doi.org/}%

```

```

2084 \providecommand \@sanitize@url[0]{\chardef\cat@space\the\catcode'\ \@sanitize\catcode'\ \cat@sp

\@@startlink How we define \@@startlink and \@@endlink will depend on whether we are
\@@endlink running under PDFLATEX. If so, and if PDF output is requested, then we
\pdfstartlink@attr use its primitives to implement hypertext, breaking out the link attributes in
\hypertext@enable@ltx \pdfstartlink@attr and using the hyperref defaults; \pdfstartlink@attr can
be redefined by a client package. Otherwise we fall back the HyperTeX standard
and leave things to the DVI translator.
A class or package that wishes to employ hypertext capabilities should execute
the \hypertext@enable@ltx procedure.
2085 \def\@@startlink#1{%
2086 \def\@@endlink{%
2087 \ifxundefined \pdfoutput {\true@sw}{\ifnum\z@=\pdfoutput}{\true@sw}{\false@sw}}%
2088 {%
2089 \def\@@startlink@hypertext#1{\leavevmode\special{html:<a href="#1">}}%
2090 \def\@@endlink@hypertext{\special{html:</a>}}%
2091 }{%
2092 \def\@@startlink@hypertext#1{%
2093 \leavevmode
2094 \pdfstartlink\pdfstartlink@attr
2095 user{/Subtype/Link/A<</Type/Action/S/URI/URI(#1)>>}}%
2096 \relax
2097 }%
2098 \def\@@endlink@hypertext{\pdfendlink}%
2099 \def\pdfstartlink@attr{attr{/Border[0 0 1 ]/H/I/C[0 1 1]}}%
2100 }%
2101 \def\hypertext@enable@ltx{%
2102 \let\@@startlink\@@startlink@hypertext
2103 \let\@@endlink\@@endlink@hypertext
2104 }%

```

`\href` The `\href` command of `hyperref` was extend somewhere between versions 6.75r and 6.80e. We apply a repair to the earlier version (if present) so that it works like the later version.

The issue is the presence of whitespace, either following the `\href` token or following the first argument's closing brace character.

```

2105 \def\href@Hy{\hyper@normalise \href@ }%
2106 \def\href@Hy@ltx{\ifnextchar\bgroup\Hy@href{\hyper@normalise\href@}}%
2107 \def\Hy@href#1{\hyper@normalise\href@}%
2108 \begingroup
2109 \endlinechar=-1 %
2110 \catcode'\^^A=14 %
2111 \catcode'\^^M\active
2112 \catcode'\%\active
2113 \catcode'\#\active
2114 \catcode'\_\active
2115 \catcode'\$\active
2116 \catcode'\&\active

```

```

2117 \gdef\hyper@normalise@ltx{^^A
2118   \begingroup
2119   \catcode'\^^M\active
2120   \def^^M{ }^^A
2121   \catcode'\%\active
2122   \let%\@percentchar
2123   \let%\@percentchar
2124   \catcode'\#\active
2125   \def#\{hyper@hash}^^A
2126   \def\#{hyper@hash}^^A
2127   \@makeother\&^^A
2128   \edef&{\string&}^^A
2129   \edef&{\string&}^^A
2130   \edef\textunderscore{\string_}^^A
2131   \let\_ \textunderscore
2132   \catcode'\_ \active
2133   \let\_ \textunderscore
2134   \let~\hyper@tilde
2135   \let\~\hyper@tilde
2136   \let\textasciitilde\hyper@tilde
2137   \let\\\@backslashchar
2138   \edef$\{string$}^^A
2139   \Hy@safe@activestrue
2140   \hyper@normalise
2141 }^^A
2142 \catcode'\#=6 ^^A
2143 \gdef\Hy@ActiveCarriageReturn@ltx{^^M}^^A
2144 \gdef\hyper@normalise@ltx#1#2{^^A
2145   \def\Hy@tempa{#2}^^A
2146   \ifx\Hy@tempa\Hy@ActiveCarriageReturn
2147     \Hy@ReturnAfterElseFi{^^A
2148       \hyper@@normalise{#1}^^A
2149     }^^A
2150   \else
2151     \Hy@ReturnAfterFi{^^A
2152       \hyper@@normalise{#1}{#2}^^A
2153     }^^A
2154   \fi
2155 }^^A
2156 \gdef\hyper@@normalise@ltx#1#2{^^A
2157   \edef\Hy@tempa{^^A
2158     \endgroup
2159     \noexpand#1{\Hy@RemovePercentCr#2%^^M\@nil}^^A
2160   }^^A
2161   \Hy@tempa
2162 }^^A
2163 \gdef\Hy@RemovePercentCr@ltx#1%^^M#2\@nil{^^A
2164   #1^^A
2165   \ifx\limits#2\limits
2166   \else

```

```

2167     \Hy@ReturnAfterFi{^^A
2168     \Hy@RemovePercentCr #2\@nil
2169     }^^A
2170   \fi
2171 }^^A
2172 \endgroup
2173 \def\switch@hyperref@href{%
2174 \expandafter\@ifx\expandafter{\csname href \endcsname\href@Hy}{
2175 \class@info{Repairing hyperref 6.75r \string\href}%
2176 \let\hyper@normalise\hyper@normalise@ltx
2177 \let\hyper@@normalise\hyper@@normalise@ltx
2178 \let\hyper@n@rmalise\hyper@n@rmalise@ltx
2179 \let\Hy@ActiveCarriageReturn\Hy@ActiveCarriageReturn@ltx
2180 \let\Hy@RemovePercentCr\Hy@RemovePercentCr@ltx
2181 \let\href\href@Hy@ltx
2182 }{}%
2183 }%
2184 \appdef\document@inithook{\switch@hyperref@href}%

```

\typeout We make the \typeout procedure of L^AT_EX be \long, because sometimes we are talking about \par.

```

2185 \def\typeout@org#1{%
2186 \begingroup
2187 \set@display@protect
2188 \immediate\write\@unused{#1}%
2189 \endgroup
2190 }%
2191 \long\def\typeout@ltx#1{%
2192 \begingroup
2193 \set@display@protect
2194 \immediate\write\@unused{#1}%
2195 \endgroup
2196 }%
2197 \@ifx{\typeout\typeout@org}{%
2198 \class@info{Making \string\typeout\space \string\long}%
2199 \let\typeout\typeout@ltx
2200 }{}%

```

6.20 End of the kernel DOCSTRIP module

Here ends the module.

```
2201 %</kernel>
```

Index

Symbols	
\#	2113, 2124, 2126, 2142
\\$	2115
\$TEXMF/	3
\%	857, 2112, 2121, 2123
\&	2116, 2127, 2129
\(622
\)	623
.aux	39
.cls	19
.dtx	5
.rtx	18
\@	435, 443
\@array	1186, 1252, 1255
\@end	354, 423
\@endlink	68
\@endlink	2077, 2082, 2085
\@endlink@hypertext	2090, 2098, 2103
\@endpbox	1353, 1376, 1482, 1491
\@chref	2076, 2077
\@par	752, 928, 968, 1359, 1383, 1509
\@startlink	68
\@startlink	2076, 2082, 2085
\@startlink@hypertext	2089, 2092, 2102
\@startpbox	1353, 1375, 1482, 1491
\@Esphack	845, 847
\@M	46
\@MM	27
\@acol	56
\@acol	1214, 1223, 1240, 1309, 1320, 1336, 1610, 1611, 1689, 1691, 1789, 1835, 1902, 1905, 1913
\@acolampacol	1787, 1789, 1797, 1805, 1825, 1833, 1849, 1857, 1901, 1903, 1911, 1915
\@acoll	1221, 1238, 1318, 1334, 1807, 1859, 1918
\@acolr	56
\@acolr	1222, 1239, 1319, 1335, 1640, 1648, 1738, 1741
\@addamp	1021, 1111, 1762 , 1788, 1803, 1831, 1855, 1902, 1906, 1913, 1920
\@addamp@LaTeX	1021, 1762
\@addamp@ltx	1111, 1765
\@addtopreamble	1597, 1693, 1745, 1766, 1772, 1775, 1781, 1784, 1810, 1814, 1818, 1862, 1866, 1870, 1876, 1879, 1882, 1885, 1888, 1892
\@afterindentfalse	889
\@afterindenttrue	887
\@ampacol	1787, 1799, 1827, 1851
\@argarraycr	1281, 1285
\@argswap	13
\@argswap	292
\@argswap@val	13
\@argswap@val	292 , 531
\@argtabularcr	1263, 1264
\@array	48
\@array	1013, 1103, 1128, 1185, 1186, 1246, 1249, 1342
\@array@LaTeX	1013, 1342
\@array@align@	1498
\@array@align@b	1499
\@array@align@c	52
\@array@align@c	1500, 1515
\@array@align@t	1498
\@array@align@v	52
\@array@align@v	1501, 1515
\@array@array	1128, 1395
\@array@array@new	1185, 1417
\@array@ltx	1103, 1362
\@array@sw	45
\@array@sw	1012, 1127, 1209
\@array@sw@LaTeX	1127, 1209
\@array@sw@array	1012, 1210
\@arrayacol	1022, 1112, 1194, 1309, 1318–1320, 1334–1336, 1768

<code>\@arrayacol@LaTeX</code> . . .	1022, 1768	<code>\@bsphack</code>	635, 850
<code>\@arrayacol@ltx</code>	1112, 1194, 1771	<code>\@caption</code>	30
<code>\@arrayclassiv</code>	1026, 1116, 1311, 1322, 1881	<code>\@capttype</code>	34
<code>\@arrayclassiv@LaTeX</code>	1026, 1881	<code>\@capttype</code>	706
<code>\@arrayclassiv@ltx</code> .	1116, 1884	<code>\@centercr</code>	19, 20
<code>\@arrayclassz</code> .	1024, 1114, 1310, 1321, 1786	<code>\@centercr</code>	454
<code>\@arrayclassz@LaTeX</code>	1024, 1786	<code>\@centering</code> .	461, 464, 467, 486, 492, 505
<code>\@arrayclassz@ltx</code> . .	1114, 1795	<code>\@chclass</code>	1608, 1609, 1624, 1637, 1682, 1687, 1713, 1735
<code>\@arrayacr</code>	47	<code>\@checkend</code>	350
<code>\@arrayacr</code>	1137, 1198, 1257 , 1312, 1323, 1415, 1441	<code>\@chnum</code> . .	1791, 1809, 1839, 1861
<code>\@arrayacr@array</code>	1137, 1269	<code>\@classi</code> .	1608, 1627, 1682, 1716
<code>\@arrayacr@new</code>	1198, 1274	<code>\@classii</code>	1608, 1629, 1682, 1718
<code>\@arrayleft</code>	1408, 1438	<code>\@classiii</code>	1608, 1631
<code>\@arrayright</code>	1458, 1462	<code>\@classiv</code>	1216, 1225, 1311, 1322, 1609, 1633
<code>\@arstrut</code>	54	<code>\@classv</code> .	1028, 1118, 1609, 1635, 1683, 1723, 1887
<code>\@arstrut</code>	1352, 1368, 1405, 1431, 1483, 1492, 1565, 1567, 1584, 1595	<code>\@classv@LaTeX</code>	1028, 1887
<code>\@arstrut@hline</code>	54	<code>\@classv@ltx</code>	1118, 1891
<code>\@arstrut@hline</code>	1559	<code>\@classvi</code>	1683, 1725
<code>\@arstrut@hline@clnc</code>	54	<code>\@classvii</code>	1684, 1727
<code>\@arstrut@hline@clnc</code>	1574, 1586	<code>\@classviii</code>	1684, 1729
<code>\@arstrut@hook</code>	54	<code>\@classx</code> .	1135, 1193, 1685, 1686, 1731, 1733, 1899
<code>\@arstrut@hook</code>	1545, 1546, 1559 , 1595, 1596	<code>\@classx@array</code>	1135, 1899
<code>\@arstrut@org</code>	1559	<code>\@classx@array@new</code> .	1193, 1909
<code>\@arstrutbox</code>	48 , 54	<code>\@classz</code> .	1215, 1224, 1310, 1321, 1608, 1625, 1682, 1714
<code>\@arstrutbox</code>	<code>\@clsextension</code>	18 , 19
	<code>\@clsextension</code> . . .	417, 434, 442
	1265, 1266, 1289, 1295, 1345, 1389, 1398, 1420	<code>\@ctrerr</code>	881
<code>\@arstrutbox@hline</code>	54	<code>\@currbox</code>	34
<code>\@arstrutbox@hline</code>	1559	<code>\@currbox</code>	797, 800, 802, 803, 809, 810, 819
<code>\@auxout</code>	627, 636, 2000	<code>\@currentHref</code>	572, 994
<code>\@backslashchar</code>	2137	<code>\@currentlabel</code>	457, 473, 599, 637
<code>\@boole@def</code>	14	<code>\@currentlabelname</code>	38
<code>\@boole@def</code>	297 , 309–312, 314–324	<code>\@currentxt</code>	18
<code>\@boolean</code>	14	<code>\@currentxt</code>	399, 416, 417, 433, 441
<code>\@boolean</code>	297	<code>\@currname</code>	18
<code>\@booleanfalse</code>	14	<code>\@currname</code> . . .	399, 415, 432, 440
<code>\@booleanfalse</code> . . .	307 , 760, 805	<code>\@currnamestack@ltx</code> .	427, 431, 436, 444, 446
<code>\@booleantrue</code>	14	<code>\@dblarg</code>	897, 898
<code>\@booleantrue</code>	307 , 690, 803		

<code>\@dblfloat</code>	741, 743, 765	<code>\@hangfrom@section</code>	37
<code>\@dblfloat@LaTeX</code>	741, 765	<code>\@height</code>	1265, 1266, 1346, 1390, 1399, 1421, 1523, 1576, 1583
<code>\@defaultsubs</code>	373	<code>\@href</code>	2075, 2076, 2079
<code>\@depth</code>	1265, 1266, 1290, 1296, 1347, 1391, 1400, 1422, 1577	<code>\@ialph</code>	876, 877
<code>\@dofilelist</code>	366	<code>\@if@empty</code>	309, 2033
<code>\@doi</code>	2081, 2082	<code>\@if@sw</code>	312, 313, 358, 375, 377
<code>\@eha</code>	197	<code>\@ifcat</code>	309
<code>\@enddocumenthook</code>	349	<code>\@ifdim</code>	309, 367, 822, 921, 961, 1575, 2028, 2035
<code>\@endpbox</code>	56	<code>\@ifempty</code>	14
<code>\@endpbox</code>	1353, 1376, 1482, 1491, 1603, 1619, 1673, 1704, 1889, 1896	<code>\@ifeof</code>	309
<code>\@eqcnt</code>	460, 465, 466, 468, 474, 487, 490, 493	<code>\@ifhbox</code>	309, 818, 837
<code>\@eqncr</code>	46	<code>\@ifhmode</code>	309
<code>\@eqncr</code>	462, 476	<code>\@ifinner</code>	309, 669, 1506
<code>\@eqnset</code>	464, 486	<code>\@ifl@ter</code>	419
<code>\@eqnswtrue</code>	458, 474	<code>\@ifmmode</code>	309, 1317, 1333, 1500, 1502, 1561
<code>\@eqpen</code>	46	<code>\@ifnextchar</code>	514, 523, 538, 1246, 1249, 1252, 1255, 1263, 1264, 1280, 1284, 2106
<code>\@esphack</code>	639	<code>\@ifnotempty</code>	14
<code>\@expast</code>	1604, 1620	<code>\@ifnotrelax</code>	13
<code>\@extension</code>	18	<code>\@ifnotrelax</code>	292, 642, 817, 836
<code>\@finalstrut</code>	605	<code>\@ifnum</code>	309, 802, 903, 932, 946, 1931, 2087
<code>\@firstampfalse</code>	1763, 1766, 1789, 1807, 1835, 1859, 1905, 1918	<code>\@ifodd</code>	309, 1933
<code>\@firstamptrue</code>	1599, 1614, 1672, 1701	<code>\@ifpackageloaded</code>	216, 563, 1002, 1126
<code>\@firstofone</code>	348	<code>\@ifstar</code>	896, 1261, 1262, 1272, 1277
<code>\@float</code>	28, 32, 34	<code>\@ifundefined</code>	912, 925, 930, 941, 944, 965, 970, 975, 978, 1990, 1996
<code>\@float@LaTeX</code>	739, 740, 764	<code>\@ifvbox</code>	309
<code>\@floatboxreset</code>	712	<code>\@ifvmode</code>	309, 508
<code>\@flushglue</code>	452, 1588	<code>\@ifvoid</code>	309, 666, 677
<code>\@font@warning</code>	368	<code>\@ifx</code>	13, 14
<code>\@footnotemark</code>	25	<code>\@ifx</code>	292, 293, 296, 309, 330, 376, 497, 576, 864, 1013–1033, 1128–1140, 1209, 1210, 1936, 2174, 2197
<code>\@footnotetext</code>	22, 25	<code>\@ifx@empty</code>	309
<code>\@footnotetext</code>	544, 564, 565, 580	<code>\@ifxundefined</code>	13
<code>\@gobble@opt@one</code>	63	<code>\@ifxundefined</code>	264, 269, 274, 292, 739, 2087
<code>\@gobble@opt@one</code>	1942		
<code>\@halignto</code>	1313, 1324, 1329, 1339, 1351, 1385, 1404, 1449		
<code>\@hang@from</code>	925, 956		
<code>\@hang@froms</code>	965, 984		
<code>\@hangfrom</code>	36–38		
<code>\@hangfrom</code>	956		

<code>\@ifxundefined@cs</code>	296, 747, 758, 778	<code>\@newl@bel</code>	361
<code>\@iiiparbox</code>	688	<code>\@nextchar</code>	1606, 1607, 1621, 1623, 1679, 1710, 1876, 1879, 1882, 1885, 1888, 1895
<code>\@itemlabel</code>	2052	<code>\@nmbulistfalse</code>	2054
<code>\@lastchclass</code>	1599, 1609, 1610, 1615, 1637, 1639, 1672, 1687, 1688, 1700, 1735, 1737, 1787, 1796, 1824, 1848, 1900, 1910	<code>\@nbreakfalse</code>	1952
<code>\@latex@warning@no@line</code>	378	<code>\@normalcr</code>	19
<code>\@listdepth</code>	2043, 2046, 2051	<code>\@normalcr</code>	448
<code>\@mainaux</code>	16	<code>\@nul</code>	862, 873
<code>\@mainaux</code>	359	<code>\@onefilewithoptions</code>	17, 18
<code>\@makecol</code>	10	<code>\@outputdblcol</code>	10
<code>\@makefnmark</code>	22, 26	<code>\@p@pfilename</code>	430
<code>\@makefnmark</code>	532, 580, 726	<code>\@p@pfilename@ltx</code>	427, 439
<code>\@makefntext</code>	26	<code>\@parboxrestore</code>	598, 711
<code>\@makefntext</code>	603	<code>\@pass@ptions</code>	405
<code>\@makeother</code>	857, 861, 2127	<code>\@percentchar</code>	2122, 2123
<code>\@minipagefalse</code>	685	<code>\@preamble</code>	1350, 1352, 1360, 1366, 1370, 1385, 1404, 1405, 1415, 1427, 1429, 1433, 1449, 1458, 1463, 1483, 1493, 1597, 1600, 1616, 1672, 1699, 1763, 1769, 1778, 1790, 1837, 1838
<code>\@mklab</code>	2053	<code>\@preamerr</code>	1359, 1383, 1611, 1643, 1645, 1692, 1693, 1695, 1743, 1745, 1749
<code>\@mkpream</code>	56, 58	<code>\@provide</code>	329, 1004, 1006, 1009
<code>\@mkpream</code>	1020, 1110, 1134, 1192, 1349, 1365, 1403, 1426, 1480, 1488, 1598, 1671	<code>\@pushfilename</code>	430
<code>\@mkpream@LaTeX</code>	1020, 1598	<code>\@pushfilename@ltx</code>	410, 430
<code>\@mkpream@array</code>	1134, 1671	<code>\@refundefined</code>	374
<code>\@mkpream@array@new</code>	1192, 1698	<code>\@reset@ptions</code>	19
<code>\@mkpream@ltx</code>	1110, 1613	<code>\@reset@ptions</code>	413, 428
<code>\@mkpream@relax</code>	56	<code>\@resetactivechars</code>	256
<code>\@mkpream@relax</code>	1663, 1703, 1753	<code>\@rightskip</code>	451
<code>\@mpargs</code>	688	<code>\@runin@to</code>	37
<code>\@mpfn</code>	23, 27	<code>\@runin@to</code>	941, 957
<code>\@mpfn</code>	517, 521, 530, 543, 549, 553, 556, 559, 600, 699	<code>\@runin@tos</code>	975, 985
<code>\@mpfootins</code>	30	<code>\@sanitize</code>	2084
<code>\@mpfootins</code>	655, 658, 659, 666, 667, 670, 673, 677, 678	<code>\@sanitize@url</code>	2075, 2078, 2081, 2084
<code>\@mpfootnotetext</code>	25	<code>\@secCNTformat</code>	36, 37
<code>\@mpfootnotetext</code>	566, 567, 580	<code>\@secCNTformat</code>	913
<code>\@mpmakefntext</code>	691	<code>\@secpenalty</code>	894
<code>\@multiplelabels</code>	376, 384	<code>\@sect</code>	36, 37
<code>\@namedef</code>	771, 1477, 1990, 1996	<code>\@sect</code>	901
<code>\@nameuse</code>	213, 694, 1363, 1439, 1991, 2017	<code>\@sect@ltx</code>	898, 902
<code>\@ne</code>	12, 66	<code>\@setckpt</code>	360

<code>\@sharp</code> .. 1356, 1379, 1411, 1444, 1481, 1489, 1602, 1618, 1653, 1659, 1673, 1693, 1702, 1745, 1792, 1793, 1811, 1815, 1819, 1840, 1842, 1844, 1863, 1867, 1871, 1889, 1896	<code>\@tabular</code> 45
<code>\@ssect</code> 36, 37, 39	<code>\@tabular</code> 1008, 1015, 1105, 1129, 1187, <u>1211</u>
<code>\@sssect</code> 958	<code>\@tabular@LaTeX</code> 1015, 1211
<code>\@sssect@ltx</code> 897, 958	<code>\@tabular@array</code> 1129, 1229
<code>\@startcolumn</code> 10	<code>\@tabular@array@new</code> 1187, 1237
<code>\@startpbox</code> 56	<code>\@tabular@ltx</code> 1105, 1220
<code>\@startpbox</code> 1353, 1375, 1482, 1491, 1603, 1619, 1669, 1673, 1704, 1888, 1894	<code>\@tabular@ltx</code> 1105, 1220
<code>\@startsection</code> 36, 39	<code>\@tabular@ltx</code> 1105, 1220
<code>\@startsection</code> 882	<code>\@tabular@ltx</code> 1105, 1220
<code>\@startsection@hook</code> .. 883, 900	<code>\@tabular@ltx</code> 1105, 1220
<code>\@starttoc</code> 64	<code>\@tabular@ltx</code> 1105, 1220
<code>\@starttoc</code> <u>1943</u>	<code>\@tabular@ltx</code> 1105, 1220
<code>\@svsec</code> 37	<code>\@tabular@ltx</code> 1105, 1220
<code>\@svsec</code> .. 905, 911, 913, 916, 926, 942	<code>\@tabular@ltx</code> 1105, 1220
<code>\@svsechd</code> 939, 973	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabacol</code> 1023, 1113, 1195, 1214, 1223, 1240, <u>1774</u>	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabacol@LaTeX</code> 1023, 1777	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabacol@ltx</code> .. 1113, 1195, 1780	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabacoll</code> 1221, 1238, 1774	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabacolr</code> 1222, 1239, 1783	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabarray</code> 46	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabarray</code> 1016, 1106, 1130, 1188, 1218, 1227, 1235, 1243, <u>1245</u> , 1314, 1325, 1330, 1340	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabarray@LaTeX</code> ... 1016, 1245	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabarray@array</code> ... 1130, 1251	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabarray@array@new</code> 1188, 1254	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabarray@ltx</code> 1106, 1248	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabclassiv</code> . 1025, 1115, 1216, 1225, <u>1875</u>	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabclassiv@LaTeX</code> . 1025, 1875	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabclassiv@ltx</code> ... 1115, 1878	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabclassz</code> .. 1027, 1117, 1215, 1224, <u>1823</u>	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabclassz@LaTeX</code> .. 1027, 1823	<code>\@tabular@ltx</code> 1105, 1220
<code>\@tabclassz@ltx</code> 1117, 1847	<code>\@tabular@ltx</code> 1105, 1220
	<code>\@tempb</code> 854, 864
	<code>\@tempboxa</code> 65
	<code>\@tempc</code> 852, 870
	<code>\@tempcnta</code> 34
	<code>\@tempcntb</code> 839, 1923, 1926, 1927, 1930, 1932, 1933
	<code>\@tempdima</code> 66
	<code>\@tempskipa</code> . 886, 888, 889, 894, 920, 921, 960, 961
	<code>\@temptokena</code> . 1674, 1680, 1705, 1710
	<code>\@testdef</code> 361
	<code>\@testpach</code> 1607, 1623, 1681, 1712
	<code>\@testtrue</code> 1927, 1933
	<code>\@textsuperscript</code> 582
	<code>\@tfor</code> ... 1605, 1621, 1679, 1710
	<code>\@thefnmark</code> .. 550, 557, 583, 600
	<code>\@toodeep</code> 2044
	<code>\@totalleftmargin</code> . 2063, 2066, 2070, 2072
	<code>\@tpfootnotetext</code> 580
	<code>\@trivlist</code> 2056
	<code>\@twopowerfourteen</code> 34
	<code>\@twopowerfourteen</code> ... 793, 821
	<code>\@twopowertwo</code> 34
	<code>\@twopowertwo</code> 794, 838, 839
	<code>\@undefined</code> 45

<code>\@undefined</code>	28, 426	<code>\@yargarraycr@LaTeX</code>	1033, 1267
<code>\@unexpandable@protect</code>	1601, 1617	<code>\@yargarraycr@array</code>	1140, 1300
<code>\@unused</code>	2188, 2194	<code>\@yargarraycr@ltx</code>	1123, 1268
<code>\@unusedoptionlist</code>	405, 406	<code>\@yargarraycr@new</code>	1201, 1304
<code>\@url</code>	2078, 2079	<code>\[</code>	21, 22
<code>\@whilesw</code>	1675, 1706	<code>\[</code>	861
<code>\@width</code>	1265, 1266, 1290, 1296, 1348, 1392, 1401, 1423, 1578	<code>\]</code>	19, 20, 46
<code>\@write@floatline</code>	35	<code>\{</code>	129, 857, 861
<code>\@write@floatline</code>	860	<code>\}</code>	129, 857, 861
<code>\@writefile</code>	627	<code>\]</code>	21
<code>\@x@sf</code>	528, 535	<code>\]</code>	861
<code>\@xargarraycr</code>	46, 47	<code>\^</code>	856, 869, 2110, 2111, 2119
<code>\@xargarraycr</code>	1032, 1122, 1139, 1200, 1257	<code>_</code>	48, 2114, 2131, 2132
<code>\@xargarraycr@LaTeX</code>	1032, 1265	<code>\~</code>	2135
<code>\@xargarraycr@array</code>	1139, 1287	<code>00readme</code>	3, 5
<code>\@xargarraycr@ltx</code>	1122, 1266		
<code>\@xargarraycr@new</code>	1200, 1293	<code>_</code>	37, 43, 63, 66, 67, 71, 88, 90, 96, 101, 108, 217, 2084
<code>\@xarraycr</code>	47		
<code>\@xarraycr</code>	1138, 1199, 1257		
<code>\@xarraycr@array</code>	1138, 1279		
<code>\@xarraycr@new</code>	1199, 1283		
<code>\@xbitor</code>	1923		
<code>\@xbitor@LaTeX</code>	1923, 1936		
<code>\@xbitor@ltx</code>	1929, 1941		
<code>\@xfloat</code>	691		
<code>\@xfloat@LaTeX</code>	691, 695		
<code>\@xfloat@anchored</code>	695, 705		
<code>\@xfloat@prep</code>	693, 697		
<code>\@xfootnotemark</code>	22		
<code>\@xfootnotenext</code>	25		
<code>\@xhline</code>	1524, 1530, 1531		
<code>\@xhline@unneeded</code>	1533		
<code>\@xifempty</code>	14		
<code>\@xssect</code>	954, 982		
<code>\@xtabularcr</code>	46		
<code>\@xtabularcr</code>	1031, 1121, 1257		
<code>\@xtabularcr@LaTeX</code>	1031, 1263		
<code>\@xtabularcr@ltx</code>	1121, 1264		
<code>\@xvi</code>	792, 802, 803		
<code>\@xxxii</code>	798, 799, 801		
<code>\@yargarraycr</code>	46, 47		
<code>\@yargarraycr</code>	1033, 1123, 1140, 1201, 1257		

argument, optional
 number 23
\array 47
\array ... 1017, 1107, 1131, 1189,
 1308
array document class . 10, 40, 41,
 45, 47, 56, 58
array environment 2, 40, 46
\array@array 1131, 1327
\array@array@new ... 1189, 1332
\array@default . 763, 1249, 1255,
 1498
\array@hook 1007
\array@LaTeX 1017, 1308
\array@ltx 1107, 1316
\array@message 1126, 1203
\array@row@pre . 1373, 1436, 1514
\array@row@pst . 1455, 1461, 1514
\array@row@rst 52
\array@row@rst
 .. 1514, 1658, 1667, 1811,
 1815, 1819, 1863, 1867, 1871,
 1896
\arraycolsep 465, 466, 1328, 1769,
 1772
\arrayrulewidth 1523, 1537, 1583
\arraystretch
 .. 1346, 1347, 1390, 1391,
 1399, 1400, 1421, 1422, 1575,
 1577
\AtBeginDocument . 15, 16, 28, 46
\AtBeginDocument 340
\AtEndDocument 15, 16
\AtEndDocument 342, 348
\AtEndOfClass 18
atveryend document class 17, 90,
 91
\author 46

B

\baselineskip 12, 21
\baselineskip . 1358, 1381, 1413,
 1446
\baselineskip 12
\begin 15

\begin@float@pagebreak ... 707,
 721, 753, 779
\belowdisplayshortskip ... 481
\belowdisplayskip 480
\bgroup 25, 52
\body@font 1520, 1555
\botrule 53
\botrule 1519
\box 65

C

\c@mpfootnote 574, 575, 701
\c@secnumdepth ... 903, 932, 946
\Call@AfterLastShipout 17
\Call@AfterLastShipout 391, 395
\caption 30
\caption 651
\cat@letter 235
\cat@space 2084
\catcode 18
\catcode 435, 443, 856,
 861, 2084, 2110–2116, 2119,
 2121, 2124, 2132, 2142
\cell@fil 1550, 1593, 1867, 1871
\cell@font 1549, 1556, 1658, 1668,
 1863, 1867, 1871, 1896
\centering 19
\changes 133–181
\chardef 234–236, 792, 852, 2084
\check@aux 347
⟨class⟩ placeholder 18
class, argument 18
\class@documenthook . 15, 16, 32
\class@documenthook . 339, 394,
 561, 651
\class@enddocumenthook . 15, 16,
 28
\class@enddocumenthook 225, 339
\class@err 197
\class@ext@hook 397
\class@extension 397
\class@extensionfile 18
\class@extensionfile 397
\class@info 197,
 420, 422, 498, 577, 808, 840,

901, 1099, 1101, 1181, 1204, 1207, 1937, 1939, 2175, 2198	<code>\count@</code> . . . 1654, 1660, 1676, 1707
<code>\class@name</code> 11	<code>\count@i</code> 261
<code>\class@name</code> . . . 197–199, 218, 227, 228	<code>\count@ii</code> 261
<code>\class@warn</code> 197, 229	<code>\countdef</code> 261, 262
<code>\class@warn@end</code> . . . 201, 206, 211, 224	<code>\crrc</code> 1452, 1455, 1458, 1461, 1466, 1542
<i>(class customization commands)</i> place- holder 9	<code>\cs</code> 133, 134, 136, 147–149, 151, 152, 154–159, 162, 164, 171, 173, 176–178
<code>\ClassError</code> 197, 218	<code>\csname</code> 13, 48
<code>classes.dtx</code> 28	<code>\csname</code> 28, 29, 239, 241, 245, 247, 296, 312, 399, 401, 407, 517, 521, 549, 553, 559, 571, 572, 600, 642, 643, 732, 734, 737, 738, 769, 770, 774, 783, 787, 789, 816, 835, 852, 855, 917, 925, 930, 935, 941, 944, 949, 965, 970, 975, 978, 1197, 1949, 1950, 1989, 1992, 1995, 1997, 2001, 2002, 2010–2012, 2021, 2051, 2174
<code>classes.dtx</code> document class . . 32	<code>\CT@cell@color</code> 1760
<code>\ClassInfo</code> 199	<code>\CT@color</code> 1755
<code>\classname</code> 57, 65, 117, 119, 121, 136, 160, 161, 173, 174, 176, 177, 181	<code>\CT@column@color</code> 1758
<code>\classoption</code> 163	<code>\CT@do@color</code> 1756
<code>\ClassWarningNoLine</code> 198	<code>\CT@end</code> 46
<code>\clear@document</code> 16, 17	<code>\CT@row@color</code> 1759
<code>\clear@document</code> 351, 388	<code>\CT@setup</code> 1754
<code>\clearpage</code> 389	<code>\CT@start</code> 46
<code>\closeout</code> 359, 773	<code>\CurrentOption</code> 412
<code>\cmd</code> 129	
<code>\col@sep</code> 45	D
<code>\col@sep</code> 1232, 1328	<code>\dollarbegin</code> . . . 1210, 1233, 1241, 1329, 1337, 1338
<code>\color</code> 1757	<code>\dollarend</code> 1234, 1242, 1329, 1338
<code>\color@begingroup</code> 602	<code>\deadcycles</code> 353
<code>\color@endgroup</code> 607, 686	<code>\DeclareRobustCommand</code> 19
<code>colortbl</code> document class . . 45, 46, 56, 58, 89, 90	<code>\DeclareRobustCommand</code> 454
<code>\colortbl@message</code> . . . 1126, 1206	<code>\dimen</code> 64
<code>\colrule</code> 53	<code>\dimen@</code> . . . 820, 821, 823, 824, 826, 827, 1574–1576, 2012, 2013
<code>\colrule</code> 1519	<code>\dimen@iii</code> 232
<code>\column@fil</code> 1519, 1554	<code>\dimendef</code> 232
<code>\column@font</code> 1519, 1553	<code>\discretionary</code> 2082
<code>\columnwidth</code> 25	<code>displaymath</code> environment 21
<code>\columnwidth</code> 710	<code>\displaystyle</code> . . . 464, 467, 486, 492
<code>\contentsline</code> 29, 63	
<code>\contentsline</code> 631, 648, 649	
<code>\contentsline@latex</code> . . . 645, 648	
<code>\copy</code> 1561	
<code>\copyright</code> 43	
<code>\count</code> 34	
<code>\count</code> 797, 800, 802, 803, 809, 1923, 1930	

<code>\displaywidth</code>	463		
<code>\do</code>	1606, 1622, 1680, 1711		
<code>\do@check@aux</code>	347		
<code>\do@if@floats</code>	28 , 32		
<code>\do@if@floats</code>	729		
<code>\do@output@cclv</code>	390		
<code>doc</code>	5		
<code>doc/</code>	3		
<code>\DocInput</code>	9		
<code>\document</code>	15		
<code>\document</code>	333		
document class			
<code>array</code>	10 , 40 , 41 , 45 , 47 , 56 , 58		
<code>atveryend</code>	17 , 90 , 91		
<code>classes.dtx</code>	32		
<code>colortbl</code>	45 , 46 , 56 , 58 , 89 , 90		
<code>ftnright</code>	10		
<code>hyperref</code> ..	20 , 22 , 24 , 25 , 29 , 30 , 38 , 39 , 63 , 66–68 , 88–91		
<code>hyperref_lpackage</code>	2 , 29		
<code>hyperref.sty</code>	91		
<code>lineno.sty</code>	66 , 90 , 91		
<code>longtable</code>	10 , 56		
<code>ltxdoc</code>	5 , 9		
<code>ltxfront</code>	26		
<code>ltxgrid</code>	10 , 16 , 17 , 25		
<code>ltxutil</code>	1 , 10 , 17		
<code>ltxutil.dtx</code>	3		
<code>ltxutil.pdf</code>	3		
<code>ltxutil.sty</code>	3		
<code>revtex4</code>	17		
<code>textcase</code>	19 , 90		
document class option			
<code>hypertext</code>	90		
<code>lengthcheck</code>	90		
document environment	5 , 16		
<code>\document@inithook</code>	15		
<code>\document@inithook</code> ..	335 , 338 , 339 , 647 , 997 , 1001 , 2184		
<code>\doi</code>	67		
<code>\doi</code>	2075		
<code>\doibase</code>	2075		
<code>\doublerulesep</code>	1536		
<code>\dp</code>	614 , 810 , 826 , 827 , 839 , 1265 , 1266 , 1289 , 1295 , 1347 , 1391 , 1400 , 1422 , 1577		
		E	
		<code>\edef</code>	56
		<code>\edef</code>	266 , 271 , 276 , 528 , 1350 , 1763 , 1769 , 1778 , 1790 , 1837 , 1992 , 2128–2130 , 2138 , 2157
		<code>\emph</code>	88
		<code>\end</code>	35
		<code>\end@float@anchored</code> ..	708 , 709 , 715
		<code>\end@float@pagebreak</code>	719 , 722 , 753
		<code>\endarray</code>	46 , 51
		<code>\endarray</code> 1001 , 1018, 1108, 1132,	
		1190, 1451 , 1469, 1472, 1475	
		<code>\endarray@array</code>	1132 , 1457
		<code>\endarray@array@new</code>	1190 , 1460
		<code>\endarray@hook</code>	1005 , 1006
		<code>\endarray@LaTeX</code>	1018 , 1451
		<code>\endarray@ltx</code>	51
		<code>\endarray@ltx</code> ..	1108 , 1454 , 1461
		<code>\endcsname</code>	28 , 29 , 239 , 241 , 245 , 247 , 296 , 312 , 399 , 401 , 407 , 517 , 521 , 549 , 553 , 559 , 571 , 572 , 600 , 642 , 643 , 732 , 734 , 737 , 738 , 769 , 770 , 774 , 783 , 787 , 789 , 816 , 835 , 852 , 855 , 917 , 925 , 930 , 935 , 941 , 944 , 949 , 965 , 970 , 975 , 978 , 1197 , 1949 , 1950 , 1989 , 1992 , 1995 , 1997 , 2001 , 2002 , 2010–2012 , 2022 , 2051 , 2174
		<code>\enddocument</code>	347
		<code>\endgroup</code>	15
		<code>\endlinechar</code>	2109
		<code>\endminipage</code>	681
		<code>\endtabular</code>	
		..	1001 , 1019 , 1109 , 1133 , 1191 , 1197 , 1465 , 1477
		endtabular environment	51
		<code>\endtabular*</code>	1477
		<code>\endtabular@array</code> ..	1133 , 1471
		<code>\endtabular@array@new</code> ...	1191 , 1474
		<code>\endtabular@hook</code> ...	1003 , 1004
		<code>\endtabular@LaTeX</code> ..	1019 , 1465

<code>\endtabular@ltx</code>	1109, 1468	<code>.rtx</code>	18
<code>\endwrite@float</code>	845, 855	<code>00readme</code>	3, 5
<code>\endwrite@floats</code>	847	<code>aps.rtx</code>	18
environment		<code>classes.dtx</code>	28
<code>array</code>	2, 40, 46	<code>doc</code>	5
<code>displaymath</code>	21	<code>doc/</code>	3
<code>document</code>	5, 16	<code>kernel</code>	2, 10, 70
<code>endtabular</code>	51	<code>latex/</code>	3
<code>eqnarray</code>	20, 21, 46	<code>ltxfloat.dtx</code>	22
<code>equation</code>	21	<code>ltxutil.dtx</code>	2, 3
<code>figure</code>	28	<code>ltxutil.pdf</code>	1
<code>figure@write</code>	28, 32	<code>ltxutil.sty</code>	1–3
<code>longtable</code>	28, 35	<code>makeindex</code>	3
<code>minipage</code>	22, 27	<code>nameref.sty</code>	38
<code>tabular</code>	2, 40, 46	<code>package</code>	2, 10
<code>\eqnarray</code>	20	<code>pdfmark.def</code>	38
<code>\eqnarray</code>	497, 499, 512	<code>revtex/</code>	3
<code>eqnarray</code> environment	20, 21, 46	<code>source/</code>	3
<code>\eqnarray@fleqn@fixed</code>	455	<code>tex/</code>	3
<code>\eqnarray@LaTeX</code>	455	<code>texmf-local/</code>	3
<code>\eqncolsep</code>	488, 491, 500, 501	<code>\file</code>	76, 79, 85, 86, 95, 101, 103, 106, 107, 110, 111, 117, 119, 121, 123, 126
<code>\equation</code>	22	<code>float, argument</code>	35
<code>\equation</code>	511	<code>\float@end@tag</code>	35
<code>equation</code> environment	21	<code>\float@end@tag</code>	860
<code>\errorstopmode</code>	253	<code>\floatingpenalty</code>	25, 27
<code>\extrarowheight</code>	1397, 1419	<code>\floats@sw</code>	31–33
F		<code>\floats@sw</code>	690, 695, 730, 760
<code>\f@ur</code>	234, 1700	<code>\flushing</code>	447
<code>\false@sw</code>	302, 308, 313, 325, 330, 1036, 1039, 1042, 1045, 1048, 1051, 1054, 1057, 1060, 1063, 1066, 1069, 1072, 1075, 1078, 1081, 1084, 1087, 1090, 1093, 1096, 1143, 1146, 1149, 1152, 1155, 1158, 1161, 1164, 1167, 1170, 1173, 1176, 1179, 2087	<code>\font@submax</code>	367, 370
<code>figure</code> environment	28	<code>\fontsubfuzz</code>	367
<code>figure@write</code> environment	28, 32	<code>\footins</code>	26
<code>\figuresname</code>	28	<code>\footins</code>	588, 678
file		<code>\footmark</code>	23
<code>\$TEXMF/</code>	3	<code>\footnote</code>	22, 23, 27
<code>.aux</code>	39	<code>\footnote</code>	514, 621
<code>.cls</code>	19	<code>\footnote@latex</code>	562
<code>.dtx</code>	5	<code>\footnotemark</code>	22, 23, 25
		<code>\footnotemark</code>	514
		<code>\footnotesep</code>	604, 613
		<code>\footnotesize</code>	611
		<code>\footnotetest</code>	514
		<code>\footnotetext</code>	22, 23
		<code>\footnotetext</code>	538
		<code>\frontmatter@footnotetext</code>	26

<code>\frontmatter@makefntext</code>	...	26	<code>\Hy@ActiveCarriageReturn@ltx</code>	2143, 2179	
<code>\frustrut</code>	1519, 1520, 1557	<code>\Hy@footnote@currentHref</code>	..	531, 572	
<code>ftnright</code>	document class	10	<code>\Hy@href</code>	2106, 2107
<code>\ftype@</code>	34	<code>\Hy@raisedlink</code>	989	
<code>\fullinterlineskip</code>	...	260, 668	<code>\Hy@RemovePercentCr</code>	2159, 2168,	2180	
<code>\futurelet</code>	1523, 1530	<code>\Hy@RemovePercentCr@ltx</code>	..	2163, 2180	
G						
<code>\gapptdef</code>	225, 263	<code>\Hy@ReturnAfterElseFi</code>	...	2147	
<code>\GetFileInfo</code>	34	<code>\Hy@ReturnAfterFi</code>	..	2151, 2167	
<code>\glet@environment</code>	212, 237	<code>\Hy@safe@activestru</code>	...	2139	
<code>\glossary</code>	620	<code>\Hy@tempa</code>	2145, 2146,	2157, 2161	
H						
<code>\H@@footnotetext</code>	22	<code>\hyper@normalise</code>	..	2148, 2152,	
<code>\H@@footnotetext</code>	..	564, 565, 702	2177	<code>\hyper@normalise@ltx</code>	...	
<code>\H@mpfootnotetext</code>	566, 567, 702		2156,	2177		
<code>\H@refstepcounter</code>	37	<code>\hyper@anchorend</code>	39	
<code>\H@refstepcounter</code>	907, 996	<code>\hyper@anchorend</code>	991	
<code>\H@svsec</code>	..	904, 908, 926, 942, 959,	<code>\hyper@anchorstart</code>	39	
		966, 976	<code>\hyper@anchorstart</code>	990	
<code>\halign</code>	51, 56	<code>\hyper@hash</code>	2125, 2126	
<code>\halign@</code>	233, 485	<code>\hyper@last</code>	39	
<code>\hangindent</code>	66	<code>\hyper@linkend</code>	39	
<code>\hb@xt@</code>	468, 493, 726, 2019, 2031		<code>\hyper@linkend</code>	533, 993	
<code>\hbox</code>	21, 66	<code>\hyper@linkstart</code>	39	
<code>\hline</code>	54	<code>\hyper@linkstart</code>	531, 992	
<code>\hline</code>	1029, 1119, 1522	<code>\hyper@makecurrent</code>	988	
<code>\hline@LaTeX</code>	1029, 1522	<code>\hyper@normalise</code>	...	2140, 2178	
<code>\hline@ltx</code>	1119, 1526	<code>\hyper@normalise@ltx</code>	2144, 2178		
<code>\hline@rule</code>	54	<code>\hyper@normalise</code>	...	2105–2107,	
<code>\hline@rule</code>	..	1521, 1529, 1544,	2176	<code>\hyper@normalise@ltx</code>	2117, 2176	
		1559	<code>\hyper@tilde</code>	2134–2136	
<code>\href</code>	67, 68	<code>\hyperanchor</code>	39	
<code>\href</code>	2075, 2105	<code>hyperref</code>	document class	..	
<code>\href@</code>	2105–2107	24, 25, 29, 30, 38, 39, 63,	66–68, 88–91		
<code>\href@Hy</code>	2105, 2174	<code>hyperref_Lpackage</code>	document class	
<code>\href@Hy@ltx</code>	2106, 2181	2, 29	<code>hyperref.sty</code>	document class	
<code>\hrule</code>	12	91	<code>hypertext</code>	document class option	
<code>\hsize</code>	15	90	<code>\hypertext@enable@ltx</code>	
<code>\ht</code>	810, 820, 823,	68	<code>\hypertext@enable@ltx</code>	...	
		824, 838, 1346, 1390, 1396,	2085			
		1418, 1574, 1575				
<code>\Hy@ActiveCarriageReturn</code>	2146,	2179				

I		J	
<code>\ialign</code>	48	<code>\j@nk</code>	331
<code>\ialign</code>	1351, 1404	K	
<code>\if</code>	48	<code>kernel</code>	2, 10, 70
<code>\if</code>	299, 311, 1343, 1409	L	
<code>\if@filesw</code> .	358, 375, 1948, 1977	<code>\l@</code>	63
<code>\if@firstamp</code>	1763, 1766	<code>\l@@sections</code>	63
<code>\if@nobreak</code>	891	<code>\l@@sections</code>	2006
<code>\if@noskipsec</code>	884	<code>\l@section</code>	1960
<code>\ifeof</code>	315	<code>\label</code>	618, 629
<code>\iffalse</code>	50, 1271, 1276	<code>\lastbox</code>	508, 671, 777
<code>\IfFileExists</code>	398	<code>\LaTeX</code> .	37, 66, 67, 77, 87, 88, 90, 101, 105, 108, 110
<code>\ifhbox</code>	316	<code>latex/</code>	3
<code>\ifinner</code>	318	<code>\LaTeXe</code>	71, 96, 192
<code>\ifmmode</code>	319	<code>\leftmargin</code> . . .	2063, 2065, 2070
<code>\ifodd</code>	321, 1927	<code>\leftmargini</code>	1960
<code>\ifvbox</code>	322	<code>\leftmarginii</code>	1961
<code>\ifvoid</code>	324	<code>\leftmarginiii</code>	1962
<code>\ignorespaces</code>		<code>\leftmarginiv</code>	1963
. . .	604, 1483, 1496, 1653, 1659, 1840, 1842, 1844, 1863, 1867, 1871, 1888, 1896, 2018, 2027, 2039, 2059	<code>\leftmarginv</code>	1964
<code>\immediate</code> . .	359, 735, 772, 870, 1950, 2000, 2188, 2194	<code>\leftmarginvi</code>	1965
<code>\incompatible@package</code>	215	<code>\leftskip</code>	66
<code>\index</code>	619	<code>\leftskip</code> .	449, 2011, 2013, 2072
<code>\init@hyperref</code>	986	<code>lengthcheck</code> document class option	90
<code>\insert</code>	26	<code>\lengthcheck@sw</code>	755
<code>\insert</code>	588, 678	<code>\let</code>	45, 64
<code>\insert@column</code> .	1136, 1196, 1651	<code>\let@environment</code>	12
<code>\insert@column@array</code>	1136, 1651	<code>\let@environment</code> .	237, 745, 746, 748, 767, 768
<code>\insert@column@array@new</code>	1196, 1656	<code>\limits</code>	2165
<code>\interdisplaylinepenalty</code> . .	46	<code>lineno.sty</code> document class	66, 90, 91
<code>\interfootnotelinepenalty</code>	612	<code>\lineskip</code>	1358, 1381, 1412, 1446
<code>\interlinepenalty</code>	612, 923, 963	<code>\linewidth</code>	485, 2064–2066
<code>\interlineskip</code>	26	<code>\list</code>	67
<code>\intertabularlinepenalty</code> . .	46	<code>\list</code>	2042
<code>\intertabularlinepenalty</code>	1257, 1258, 1260, 1262, 1277	<code>\listparindent</code> .	2049, 2062, 2069
<code>\intertextsep</code>	721, 722	<code>\LoadClass</code>	18
<code>\item</code> .	99, 109, 112, 116, 118, 120, 125, 127	<code>\long</code>	70
<code>\itemindent</code>	2050	<code>longtable</code> document class	10, 56
<code>\itshape</code>	574, 583	<code>longtable</code> environment . .	28, 35
		<code>\loop</code>	15

<code>\loopuntil</code>	327	<code>\makelabel</code>	2053
<code>\loopwhile</code>	327	<code>\maketitle</code>	25
<code>\lrstrut</code>	1520, 1521, 1558	<code>\maketitle</code>	55
<code>\LT@array</code>	10	<code>\mark</code>	28
<code>ltxfloat.dtx</code>	22	<code>\mathchardef</code>	793, 794
<code>\ltx@contentsline</code>	629	<code>\mathindent</code>	21
<code>\ltx@def@footproc</code>	23, 24	<code>\meaning</code>	258
<code>\ltx@def@footproc</code>	516, 524, 539, 547	<code>\MessageBreak</code>	369
<code>\ltx@footmark</code>	23	minipage environment	22, 27
<code>\ltx@footmark</code>	514	<code>\minipagefootnote@drop</code>	25
<code>\ltx@footnote@pop</code>	698, 704	<code>\minipagefootnote@drop</code>	608, 654
<code>\ltx@foottext</code>	23	<code>\minipagefootnote@foot</code>	654
<code>\ltx@foottext</code>	514	<code>\minipagefootnote@here</code>	30
<code>\ltx@make@current@footnote</code>	514	<code>\minipagefootnote@here</code>	652, 654, 716
<code>\ltx@mathindent</code>	21	<code>\minipagefootnote@init</code>	654, 713
<code>\ltx@mathindent</code>	475, 505	<code>\minipagefootnote@pick</code>	592, 654
<code>\ltx@stp@footproc</code>	23, 24	<code>\mpfootnote</code>	577
<code>\ltx@stp@footproc</code>	520, 525, 540, 555	<code>\multicolumn</code>	1014, 1104, 1478
<code>\ltx@thempfootnote</code>	575, 578	<code>\multicolumn@LaTeX</code>	1014, 1478
<code>\ltx@xfootmark</code>	523, 524	<code>\multicolumn@ltx</code>	1104, 1485
<code>\ltx@xfootnote</code>	23	<code>\multiply</code>	799
<code>\ltx@xfootnote</code>	514	<code>\multispan</code>	1479, 1486
<code>\ltx@xfoottext</code>	538, 539		
<code>\ltx@yfootmark</code>	523, 525	N	
<code>\ltx@yfootnote</code>	514, 519	<code>nameref.sty</code>	38
<code>\ltx@yfoottext</code>	538, 540	<code>\NC@list</code>	1675, 1706
<code>ltxdoc</code> document class	5, 9	<code>\newbox</code>	1570
<code>ltxgrid</code> document class	26	<code>\newcount</code>	1257, 1259
<code>ltxgrid</code> document class	10, 16, 17, 25	<code>\newenvironment</code>	35
<code>\ltxu@dotsep</code>	1959, 2034	<code>\newlabel</code>	29
<code>ltxutil</code> document class	1, 10, 17	<code>\newlabel</code>	637
<code>ltxutil.dtx</code>	2, 3	<code>\newlinechar</code>	869
<code>ltxutil.dtx</code> document class	3	<code>\newwrite</code>	731, 1949
<code>ltxutil.pdf</code>	1	<code>\noalign</code>	1264, 1266–1268, 1285, 1298, 1302, 1306, 1521, 1523, 1527, 1531, 1543
<code>ltxutil.pdf</code> document class	3	<code>\noexpand</code>	258, 259, 1351, 1404, 2159
<code>ltxutil.sty</code>	1–3	<code>\normalfont</code>	583
<code>ltxutil.sty</code> document class	3	number, optional argument	23
		<code>\numberline</code>	65, 66
		<code>\numberline@@sections</code>	2016, 2030
M			
<code>\make@footnotetext</code>	25, 26		
<code>\make@footnotetext</code>	580		
<code>\makeatletter</code>	363, 411, 1946		
<code>makeindex</code>	3		

O	
<code>\obeylines</code>	866
<code>\obsolete@command</code>	200
<code>\oneapage</code>	753
<code>\openout</code>	736, 1950
<code>\OptionNotUsed</code>	19
<code>\other</code>	236
P	
<code>\p@equation</code>	457, 473
<code>package</code>	2, 10
<code>\package@name</code>	190, 191
<code>\PackageInfo</code>	191
<code>\par</code>	65, 66, 70
<code>\parindent</code>	21, 66
<code>\parindent</code> 724, 2013, 2031, 2062, 2069	
<code>\parsep</code>	2057
<code>\parshape</code>	66
<code>\parshape</code>	2066
<code>\parskip</code>	21
<code>\parskip</code>	479, 2057
<code>\parskipfillskip</code>	21
<code>\partopsep</code>	478
<code>\pbox@hook</code>	1594, 1896
<code>\pdfendlink</code>	2098
<code>pdfmark.def</code>	38
<code>\pdfoutput</code>	2087
<code>\pdfstartlink</code>	2094
<code>\pdfstartlink@attr</code>	68
<code>\pdfstartlink@attr</code>	2085
<code>\phantomsection</code>	39
<code>\phantomsection</code> .. 904, 909, 959, 987	
placeholder	
<code>\langle class \rangle</code>	18
<code>\langle class customization commands \rangle</code>	9
<code>\langle your document here \rangle</code>	9
<code>\prep@math</code>	21
<code>\prep@math</code>	507
<code>\prep@math@patch</code>	21, 22
<code>\prep@math@patch</code>	507
<code>\prepdef</code>	13
<code>\prepdef</code>	263, 333, 511, 512, 652, 769, 770,
	913, 915, 1003, 1005, 1008, 1366, 1429
<code>\prepnex@tok</code>	1678, 1709
<code>\prevdepth</code>	12
<code>\prevdepth</code>	260
<code>\print@float</code>	33
<code>\print@float</code>	752
<code>\printfigures</code>	28
<code>\ProcessOptions</code>	19
<code>\protected@edef</code>	24
<code>\protected@edef</code> .. 572, 599, 911	
<code>\protected@write</code>	627, 636
<code>\protected@xdef</code>	557
<code>\providecommand</code>	395, 987–994, 1007, 2075, 2078, 2080, 2081, 2083, 2084
<code>\ProvidesFile</code>	6
<code>\ProvidesPackage</code>	19
R	
<code>\raggedleft</code>	19
<code>\raggedright</code>	19
<code>\realfootnote</code>	25
<code>\RecordChanges</code>	32
<code>\refstepcounter</code>	37
<code>\relax</code>	37, 40
<code>\repeat</code>	15
<code>\replace@command</code>	205
<code>\replace@environment</code>	210
<code>\RequirePackage</code>	25, 27, 29
<code>\reserved@a</code> .. 1524, 1530, 1534, 1535, 1606, 1621	
<code>\reset@font</code>	611
<code>revtex/</code>	3
<code>revtex4</code> document class	17
<code>\robustify@contents</code>	617
<code>\romannumeral</code>	2051
<code>\rule</code>	604
S	
<code>\samepage</code>	46
<code>\samepage</code>	1260
<code>\save@decl</code>	1683, 1720
<code>\say</code>	12
<code>\say</code>	258, 1534
<code>\saythe</code>	12

<code>\saythe</code>	258	<code>\substyle@ext</code>	18
<code>\sc</code>	87	<code>\switch@array</code>	41, 47
<code>\sec@upcase</code>	40	<code>\switch@array</code>	1002, 1011
<code>\sec@upcase</code>	1000	<code>\switch@array@info</code>	1126, 1183
<code>\section</code>	83, 780	<code>\switch@hyperref@href</code>	2173, 2184
<code>\Sectionformat</code>	38	<code>\switch@tabular</code>	41, 46
<code>\set@arstrutbox</code>	54	<code>\switch@tabular</code>	1002, 1011
<code>\set@arstrutbox</code>	1364, 1388, 1559		
<code>\set@display@protect</code>	2187, 2193		
<code>\set@eqnarray@skips</code>	506		
<code>\set@footnotefont</code>	27		
<code>\set@footnotefont</code>	596, 610		
<code>\set@footnotewidth</code>	25, 27		
<code>\set@footnotewidth</code>	580		
<code>\set@footnotewidth@one</code>	616		
<code>\set@listindent</code>	2058, 2074		
<code>\set@listindent@</code>	2068		
<code>\set@listindent@parshape</code>	2061, 2074		
<code>\set@tocdim@pagenum</code>	65, 66		
<code>\set@tocdim@pagenum</code>	2009, 2026		
<code>\set@typeset@protect</code>	1357, 1380, 1445, 1481, 1490		
<code>\shipout</code>	12		
<code>\show@box@size@sw</code>	807		
<code>\showboxbreadth</code>	253		
<code>\showboxdepth</code>	253		
<code>\showoutput</code>	256		
<code>\skip@</code>	2015, 2018		
<code>source/</code>	3		
<code>\spacefactor</code>	528, 535		
<code>\special</code>	2089, 2090		
<code>\splitmaxdepth</code>	614		
<code>\splittopskip</code>	613		
<code>\stepcounter</code>	456, 473, 556		
<code>\stop</code>	219		
<code>\StopEventually</code>	6		
<code>\string</code>	201, 206, 217, 577, 622–624, 627, 637, 901, 1937, 1939, 2001, 2128–2130, 2138, 2175, 2198		
<code>\strutbox</code>	605, 614, 1346, 1347, 1390, 1391, 1396, 1400, 1418, 1422, 1575, 1577		
<code>\subsection</code>	93, 132		
		T	
		<code>\t@</code>	231, 233
		<code>\tab@rule</code>	54
		<code>\tab@rule</code>	1519, 1520, 1541
		<code>\tabcolsep</code>	1232, 1590–1592, 1778
		<code>\tableleft@skip</code>	1382, 1447, 1587
		<code>\tableleftsep</code>	1590, 1664, 1775
		<code>\tableofcontents</code>	81
		<code>\tabmid@skip</code>	1367, 1430, 1588
		<code>\tabmidsep</code>	1591, 1665, 1781
		<code>\tabright@skip</code>	1371, 1434, 1589
		<code>\tabrightsep</code>	1592, 1666, 1784
		<code>\tabskip</code>	21, 48
		<code>\tabskip</code>	461, 464, 467, 469, 475, 486, 492, 494, 1352, 1367, 1371, 1382, 1406, 1430, 1434, 1447
		<code>\tabular</code>	40
		tabular environment	2, 40, 46
		<code>\tabular@hook</code>	1008, 1009
		<code>\tabularnewline</code>	1354, 1377, 1415, 1442
		<code>\tally@float</code>	34
		<code>\tally@float</code>	792
		<code>\TeX</code>	63, 71, 110, 181
		<code>tex/</code>	3
		<code>texmf-local/</code>	3
		text, argument	23, 40
		<code>\textasciitilde</code>	2136
		textcase document class	19, 90
		<code>\texttt</code>	48, 106, 107, 114, 123, 126, 129, 133
		<code>\textunderscore</code>	2130, 2131, 2133
		<code>\textwidth</code>	822
		<code>\thanks</code>	38, 42, 48
		<code>\the@toks</code>	1652, 1654, 1657, 1660, 1677, 1696, 1708, 1751

`\vcenter` . 1343, 1409, 1500, 1504,
1507
`\voidb@x` 655
`\vrule` 1265, 1266, 1290, 1296, 1346,
1390, 1398, 1421, 1576
`\vsize` 15
`\vsplit` 26
`\vtop` 670, 1343, 1409, 1498

W

`\write` 870, 2000, 2188, 2194
`\write@float` [844](#), [848](#)
`\write@float` [28](#), [32](#), [35](#)
`\write@float` 742, [844](#)
`\write@floatline` [35](#)
`\write@floatline` 858, [860](#)
`\write@floats` 743, [844](#)

X

`\xdef` 415, 431, 1404, 1427

Y

`your_toks_here`, argument . [15](#)
`<your document here>` placeholder [9](#)

Z

`\z@skip` .. 449–451, 464, 469, 486,
494, 605, 717, 1352, 1358,
1381, 1587–1589, 2019

Change History

4.0b		4.0c	
	<code>\@mpmakefn</code> text: AO: Removed superfluous <code>\defs</code> , changed to using <code>\floats@sw</code> as the flag. Also stopped using DPC's <code>\if@twocolumn</code> flag: using <code>\floats@sw</code> instead. Also added <code>\par\vskip\z@skip</code> after the <code>\minipagefootnotes</code> so that the float box would have zero depth like the kernel one. 31		<code>\@mpmakefn</code> text: (AO, 110) Install hooks for endfloats processing 31
	General: AO: Fixed spurious CR and (return) characters in output file. Also, if the document did not have the <code>\endfigure</code> on a line of its own, the macro wouldn't work. Fixed. 3		<code>\@sssect</code> : (AO, 116) Hyperref compatibility 39
	AO: Removed superfluous <code>\defs</code> , changed to using <code>\floats@sw</code> as the flag. Also stopped using DPC's <code>\if@twocolumn</code> flag: using <code>\floats@sw</code> instead. Also added <code>\par\vskip\z@skip</code> after the <code>\minipagefootnotes</code> so that the float box would have zero depth like the kernel one. 3		General: (AO, 110) Install hooks for endfloats processing 3
	only execute if there really were floats of the given type 3		(AO, 116) Hyperref compatibility 3
	Support the hack with <code>\prepdef</code> , and delay until <code>\AtBeginDocument</code> time, since hyperref clobbers <code>\caption</code> . . . 3		(AO, 130) Interference from array package 3
	<code>\caption</code> : Support the hack with <code>\prepdef</code> , and delay until <code>\AtBeginDocument</code> time, since hyperref clobbers <code>\caption</code> . . 30		*-form mandates pagebreak at each float; only print section head if there is something there. 3
	<code>\print@float</code> : only execute if there really were floats of the given type 33		<code>\endarray</code> : (AO, 130) Interference from array package 40
	<code>\write@float</code> : AO: Fixed spurious CR and (return) characters in output file. Also, if the document did not have the <code>\endfigure</code> on a line of its own, the macro wouldn't work. Fixed. 35		<code>\print@float</code> : *-form mandates pagebreak at each float; only print section head if there is something there. 33
		4.0d	
			<code>\@mpmakefn</code> text: (AO, 127) Floats placed [h] to allow page breaks 31
			(AO, 224) Hyperref compatibility. 31
			General: (AO, 127) Floats placed [h] to allow page breaks 3
			(AO, 174) kernel fix 3, 22
			(AO, 224) Hyperref compatibility. 3
			Allow things to break over pages by setting <code>array@default</code> 3
			<code>\print@float</code> : Allow things to break over pages by setting <code>array@default</code> 33
		4.0e	
			<code>\@mpmakefn</code> text: (AO, 221) Remove <code>samepage</code> command from <code>@xfloat@prep</code> : If the float can break over pages, we want better control. 31
			General: (AO, 221) Remove <code>samepage</code> command from <code>@xfloat@prep</code> : If the float can break over pages, we want better control. 3

4.0f			
	<code>\@ssect:</code> (AO, 404) Hyperref compatibility	39	
	General: (AO, 404) Hyperref compatibility	3	
4.1a			
	<code>\@mpmakefntext:</code> <code>\@xfloat@prep</code> calls <code>\ltx@footnote@pop</code> to restore the original <code>\ltx@footmark</code> and <code>\ltx@foottext</code> procedures, in case footnote processing has switched.	31	
	<code>\@p@pfilename:</code> Class extension mechanism <code>\@pushfilename@ltx</code> and <code>\@p@pfilename@ltx</code> .	19	
	General: (AO) Make <code>\addtocontents</code> a <code>\long \def;</code> gobble up <code>\footnote</code>	3	
	(AO) Remove code that avoided changes to <code>\xfootnotemark</code>	3	
	(AO, 438) Complete rewrite of footnote macros.	3	
	(AO, 459) do not assume <code>\class@name</code> is defined	3, 11	
	(AO, 461) Change the csname from <code>\@dotsep</code> to <code>\ltxu@dotsep</code> . The former is understood in mu. (What we wanted was a dimension.)	3	
	(AO, 475) I had not properly reproduced the LaTeX macro <code>\eqnarray</code> .	3	
	<code>\@xfloat@prep</code> calls <code>\ltx@footnote@pop</code> to restore the original <code>\ltx@footmark</code> and <code>\ltx@foottext</code> procedures, in case footnote processing has switched.	3	
	<code>\class@documenthook</code> is the last <code>\AtBeginDocument</code> token now	3	
	Class extension mechanism <code>\@pushfilename@ltx</code> and <code>\@p@pfilename@ltx</code> .	3	
	Class extension mechanism <code>\class@extension</code> , <code>\class@extensionfile</code> , and <code>\class@ext@hook</code> .	3, 17	
	Get rid of <code>\set@typesize@hook</code> <code>\set@pica@hook</code> and the		
	<code>\normalsize</code> directive	3	
	<code>\class@enddocumenthook:</code> <code>\class@documenthook</code> is the last <code>\AtBeginDocument</code> token now	15	
	<code>\document:</code> Get rid of <code>\set@typesize@hook</code> <code>\set@pica@hook</code> and the <code>\normalsize</code> directive	15	
	<code>\eqnarray@fleqn@fixed:</code> (AO, 475) I had not properly reproduced the LaTeX macro <code>\eqnarray</code> .	20	
	<code>\footnote:</code> (AO) Remove code that avoided changes to <code>\xfootnotemark</code>	25	
	<code>\ltx@make@current@footnote:</code> (AO, 438) Complete rewrite of footnote macros.	23	
	<code>\numberline@sections:</code> (AO, 461) Change the csname from <code>\@dotsep</code> to <code>\ltxu@dotsep</code> . The former is understood in mu. (What we wanted was a dimension.)	66	
	<code>\robustify@contents:</code> (AO) Make <code>\addtocontents</code> a <code>\long \def;</code> gobble up <code>\footnote</code>	29	
	<code>\toc@font:</code> (AO, 461) Change the csname from <code>\@dotsep</code> to <code>\ltxu@dotsep</code> . The former is understood in mu. (What we wanted was a dimension.)	63	
			4.1b
	<code>\@mkpream:</code> (AO, 505) try to accommodate <code>colortbl</code> .	56	
	<code>\@mkpream@relax:</code> (AO, 505) try to accommodate <code>colortbl</code> .	58	
	<code>\@mpmakefntext:</code> No need to protect against undefined <code>\float@sw</code>	31	
	<code>\@tabarray:</code> (AO, 505) try to accommodate <code>colortbl</code> .	46	
	<code>\@tabular:</code> (AO, 505) try to accommodate <code>colortbl</code> .	45	
	General: (AO, 487) Support for video figures and the <code>\setfloatlink</code> command	3	
	(AO, 505) try to accommodate <code>colortbl</code> .	3	
	Acquire <code>hyperref</code> <code>savoire</code>	3	

Default assignment of <code>\float@sw</code> now, not at <code>\AtBeginDocument</code> time.	3	(AO, 518) Tally register overflow when locument is long	3
If class option <code>lengthcheck</code> is in effect, log the height of this float class.	3	<code>\set@footnotewidth</code> : (AO, 515) Hook for setting the font of a footnote	27
No need to protect against undefined <code>\float@sw</code>	3	<code>\total@float</code> : (AO, 518) Tally register overflow when locument is long	34
Patch the array package even later: after all package patches go in.	3	4.1g	
Refine toc processing: provide default.	3	General: (AO, 525) Remove phantom paragraph above display math that is given in vertical mode	3, 21
Tally and log the height of a float class	3	(AO, 532) Both arguments of <code>\href</code> get sanitized	3
<code>\array</code> : (AO, 505) try to accommodate <code>colortbl</code>	48	(AO, 539) Use of double-backslash in argument of <code>\section</code> gives error. The <code>textcase</code> package is involved.	3, 19
<code>\doif@floats</code> : No need to protect against undefined <code>\float@sw</code> .	32	<code>\doibase</code> : (AO, 532) Both arguments of <code>\href</code> get sanitized .	67
<code>\doibase</code> : (AO, 487) Support for video figures and the <code>\setfloatlink</code> command . . .	67	4.1j	
<code>\endarray</code> : Patch the array package even later: after all package patches go in.	40	<code>\doibase</code> : (AO, 545) Provide definition for <code>\doi</code> that does hypertext	67
<code>\floats@sw</code> : Default assignment of <code>\float@sw</code> now, not at <code>\AtBeginDocument</code> time. . . .	31	<code>\hypertext@enable@ltx</code> : (AO, 545) hypertext capabilities off by default; enable with <code>hypertext</code>	68
<code>\init@hyperref</code> : Acquire <code>hyperref</code> <code>savoire</code>	39	4.1n	
<code>\ltx@contentsline</code> : Refine toc processing: provide default. . .	29	General: (AO) Incorporate change to <code>ltxmiscen.dtx</code> v1.1i 2000/05/19	3
<code>\print@float</code> : If class option <code>lengthcheck</code> is in effect, log the height of this float class. .	33	(AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page	3
<code>\switch@array</code> : (AO, 505) Try to accommodate <code>colortbl</code>	43	(AO, 569) execute <code>atveryend</code> 's <code>\Call@AfterLastShipout</code> at the proper time	3
<code>\total@float</code> : Tally and log the height of a float class	34	(AO, 571) Interface <code>\set@footnotewidth</code> for determining the set width of footnotes	3
4.1d		(AO, 571) allow split after last line of footnote	3
General: (AO, 511) Compatability with <code>lineno.sty</code> 's erroneous way of detecting <code>fleqn.clo</code>	3	(AO, 572) title block footnotes numbered independently from body footnotes	3
<code>\eqnarray@fleqn@fixed</code> : (AO, 511) Compatability with <code>lineno.sty</code> 's erroneous way of detecting <code>fleqn.clo</code>	21	(AO, 574) protect against <code>lineno.sty</code> , which forces a visit	
4.1f			
General: (AO, 515) Hook for setting the font of a footnote	3		

to the output routine, which appears to destroy the value of <code>\@tempdima</code>	3	the value of <code>\@tempdima</code>	66
<code>\clear@document</code> : (AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page . .	17	<code>\set@footnotewidth</code> : (AO, 571) Interface <code>\set@footnotewidth</code> for determining the set width of footnotes	27
(AO, 569) execute <code>atveryend</code> 's <code>\Call@AfterLastShipout</code> at the proper time	17	(AO, 571) allow split after last line of footnote	27
<code>\do@check@aux</code> : (AO) Incorporate change to <code>ltnmiscen.dtx v1.1i 2000/05/19</code>	16	(AO, 572) title block footnotes numbered independently from body footnotes	26
(AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page	16	4.1p	
<code>\l@sections</code> : (AO, 574) protect against <code>lineno.sty</code> , which forces a visit to the output routine, which appears to destroy		General: (AO, 582) A patch of <code>hyperref.sty</code> to provide backward compatibility to <code>T_EXLive 2007</code> 's version 6.75r	3
		<code>\href</code> : (AO, 582) A patch of <code>hyperref.sty</code> to provide backward compatibility to <code>T_EXLive 2007</code> 's version 6.75r	68