

# The `revtex4-1` document class of the American Physical Society \*

Arthur Ogawa †

Version 4.1s, dated 2020/09/30

This file embodies the implementation of the APS REVTeX 4.1 document class for electronic submissions to journals.

The distribution point for this work is <http://publish.aps.org/revtex4/>, which contains fully unpacked, prebuilt runtime files and documentation.

## Contents

<b>1</b>	<b>Using REVTeX</b>	<b>2</b>
1.1	Bill of Materials . . . . .	2
1.1.1	Primary Source . . . . .	2
1.1.2	Generated by <code>latex ltxutil.dtx</code> . . . . .	3
1.1.3	Generated by <code>tex ltxutil.dtx</code> . . . . .	3
1.1.4	Auxiliary . . . . .	3
<b>2</b>	<b>Code common to all modules</b>	<b>3</b>
<b>3</b>	<b>The driver module <code>doc</code></b>	<b>4</b>
3.1	The Preamble . . . . .	4
3.1.1	Docstrip and info directives . . . . .	4
3.2	The “Read Me” File . . . . .	5
3.3	The Document Body . . . . .	8
<b>4</b>	<b>Using this package</b>	<b>8</b>
4.1	Invoking the package . . . . .	8
<b>5</b>	<b>Compatibility with L<sup>A</sup>T<sub>E</sub>X’s Required Packages</b>	<b>9</b>
5.1	<code>array</code> . . . . .	9
5.2	<code>longtable</code> . . . . .	9

---

\*Work under hire to The American Physical Society.

†First revision of REVTeX4.0 (unreleased) by David Carlisle

<b>6</b>	<b>Implementation of package</b>	<b>9</b>
6.1	Beginning of the <code>package</code> DOCSTRIP module	9
6.2	Banner and beginning of the <code>kernel</code> DOCSTRIP module	10
6.3	Errors and warnings	10
6.4	New Tools	11
6.5	Boolean Control	12
6.6	Begin Document Structure	14
6.7	Class Extensions	18
6.8	Type Tools	20
6.9	Display Math	21
6.10	Footnotes	23
6.11	Floats	28
	6.11.1 Usage notes	28
	6.11.2 Robustifying fragile commands	29
	6.11.3 Preparing for the <code>hyperref</code> package	30
	6.11.4 Footnotes within floats, unfloating floats, float font	30
	6.11.5 Writing floats out to a file	33
6.12	Counters	37
6.13	Customization of Sections	37
6.14	Patch the <code>tabular</code> and <code>array</code> Environments	41
6.15	Repair other broken parts of L <sup>A</sup> T <sub>E</sub> X	63
6.16	Syntax	63
6.17	Auto-indented Contents	63
6.18	Lists	67
6.19	Hypertext capabilities	68
6.20	End of the <code>kernel</code> DOCSTRIP module	71
	<b>Index</b>	<b>72</b>

## 1 Using REV<sub>T</sub>E<sub>X</sub>

The file `README` has retrieval and installation information.

User documentation is presented separately in `auguide.tex`.

The file `template.aps` is a boilerplate file.

### 1.1 Bill of Materials

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

#### 1.1.1 Primary Source

One single file generates all.

```
%ltxutil.dtx
%
```

### 1.1.2 Generated by latex ltxutil.dtx

Typesetting the source file under pdf<sub>l</sub>atex generates the readme and the documentation.

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

### 1.1.3 Generated by tex ltxutil.dtx

Typesetting this file with T<sub>E</sub>X generates the package file.

```
%ltxutil.sty
%
```

### 1.1.4 Auxiliary

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

```
%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<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. An appropriate message is displayed if a different T<sub>E</sub>X 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 [2020/09/30 4.1s utilities package (portions licensed from W. E. Baxter web at superscript.com
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 \pdfstringdefDisableCommands{%
29   \let\file\relax
30   \let\sc\relax
31 }
32 \makeatletter
33 \@ifundefined{package@font}{}%
34   {\expandafter\RequirePackage\expandafter{\csname package@font\endcsname}}
35 \makeatother
36 \CodelineIndex\EnableCrossrefs % makeindex -s gind.ist ltxutil
37 \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.

```
38 \setcounter{StandardModuleDepth}{1}
```

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

```
39 \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.

```
40 \begin{filecontents*}{00readme}
41 \title{%
42 A \LaTeX\ Package of utility macros%
43 \thanks{%
44 This file has version number \fileversion,
45 last revised \filedate.%
46 }%
47 \thanks{%
48 Version \fileversion\ \copyright\ 2009--2020 The American Physical Society
49 }%
50 }%
51 \author{%
52 Arthur Ogawa%
53 \thanks{\texttt{mailto:arthur\_ogawa at sbcglobal.net}}}%
54 }%
55 %\iffalse
56 % For version number and date,
57 % search on "\fileversion" in the .dtx file,
58 % or see the end of the 00readme file.
59 %\fi
60 \maketitle
61
62 This file embodies the \classname{ltxutil} package,
63 the implementation and its user documentation.
64
65 The distribution point for this work is
66 \url{publish.aps.org/revtex},
67 which contains prebuilt runtime files, documentation, and full source,
68 ready to add to a TDS-compliant \TeX\ installation.
69
70 The \classname{ltxutil} package was commissioned by the American Physical Society
71 and is distributed under the terms of the \LaTeX\ Project Public License,
72 the same license under which all the portions of \LaTeX\ itself are distributed.
```

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

```

123 \item
124 \file{$TEXMF/}\file{source/}\file{latex/}\file{revtex/}\classname{ltxutil.dtx}
125 \item
126 \file{$TEXMF/}\file{doc/}\file{latex/}\file{revtex/}\classname{ltxutil.pdf}
127 \end{itemize}
128 where \file{$TEXMF/} stands for \file{texmf-local/}, or some other \texttt{texmf} tree
129 in your installation.
130 \item
131 Run \texttt{mktexlsr} on \file{$TEXMF/} (you may need root access).
132 \item
133 Build and installation are now complete;
134 now put a \cmd\usepackage\texttt{\ltxutil\} in your document preamble!
135 \end{enumerate}
136
137 \subsection{Change Log}
138 \changes{4.0b}{1999/06/20}{A0: Fixed spurious \texttt{CR} and (return) characters in output fil
139 \changes{4.0b}{1999/06/20}{A0: Removed superfluous \cs{def}s, changed to using \cs{floats@sw} a
140 \changes{4.0b}{1999/06/20}{only execute if there really were floats of the given type}
141 \changes{4.0b}{1999/06/20}{Support the hack with \cs{prepdef}, and delay until \cs{AtBeginDocum
142 \changes{4.0c}{1999/11/13}{(A0, 110) Install hooks for endfloats processing}
143 \changes{4.0c}{1999/11/13}{(A0, 116) Hyperref compatibility}
144 \changes{4.0c}{1999/11/13}{(A0, 130) Interference from array package}
145 \changes{4.0c}{1999/11/13}{*-form mandates pagebreak at each float; only print section head if
146 \changes{4.0d}{2000/04/10}{(A0, 127) Floats placed [h] to allow page breaks}
147 \changes{4.0d}{2000/04/10}{(A0, 174) kernel fix}
148 \changes{4.0d}{2000/05/19}{(A0, 224) Hyperref compatibility.}
149 \changes{4.0d}{2000/05/23}{Allow things to break over pages by setting array@default.}
150 \changes{4.0e}{2000/11/16}{(A0, 221) Remove samepage command from @xfloat@prep: If the float ca
151 \changes{4.0f}{2001/07/13}{(A0, 404) Hyperref compatibility}
152 \changes{4.1a}{2008/01/19}{(A0, 459) do not assume \cs{class@name} is defined}%
153 \changes{4.1a}{2008/01/19}{(A0, 461) Change the csname from \cs{@dotsep} to \cs{ltxu@dotsep}. T
154 \changes{4.1a}{2008/01/19}{(A0, 475) I had not properly reproduced the LaTeX macro \cs{eqnarray
155 \changes{4.1a}{2008/01/19}{(A0, 479) Per: Dylan Thurston<dpt at math.harvard.edu>}%
156 \changes{4.1a}{2008/06/30}{(A0) Make \cs{addtocontents} a \cs{long} \cs{def}; gobble up \cs{foo
157 \changes{4.1a}{2008/06/30}{(A0) Remove code that avoided changes to \cs{xfootnotemark}}%
158 \changes{4.1a}{2008/06/30}{(A0, 438) Complete rewrite of footnote macros.}
159 \changes{4.1a}{2008/07/07}{\cs{xfloat@prep} calls \cs{ltx@footnote@pop} to restore the origina
160 \changes{4.1a}{2008/08/12}{\cs{class@documenthook} is the last \cs{AtBeginDocument} token now}
161 \changes{4.1a}{2008/08/12}{Class extension mechanism \cs{@pushfilename@ltx} and \cs{@ppfilename
162 \changes{4.1a}{2008/08/12}{Class extension mechanism \cs{class@extension}, \cs{class@extensionf
163 \changes{4.1a}{2008/08/12}{Get rid of \cs{set@typesize@hook} \cs{set@pica@hook} and the \cs{nor
164 \changes{4.1b}{2008/08/12}{(A0, 487) Support for video figures and the \cs{setfloatlink} comman
165 \changes{4.1b}{2008/08/12}{(A0, 505) try to accommodate \classname{colortbl}.}
166 \changes{4.1b}{2008/08/12}{Acquire \classname{hyperref} saveoire}
167 \changes{4.1b}{2008/08/12}{Default assignment of \cs{float@sw} now, not at \cs{AtBeginDocument}
168 \changes{4.1b}{2008/08/12}{If class option \classoption{lengthcheck} is in effect, log the heig
169 \changes{4.1b}{2008/08/12}{No need to protect against undefined \cs{float@sw}}
170 \changes{4.1b}{2008/08/12}{Patch the array package even later: after all package patches go in.
171 \changes{4.1b}{2008/08/12}{Refine toc processing: provide default.}%
172 \changes{4.1b}{2008/08/12}{Tally and log the height of a float class}

```

```

173 \changes{4.1d}{2009/03/27}{(AO, 511) Compatability with lineno.sty's erroneous way of detecting
174 \changes{4.1f}{2009/07/07}{(AO, 515) Hook for setting the font of a footnote}
175 \changes{4.1f}{2009/07/10}{(AO, 518) Tally register overflow when locument is long}
176 \changes{4.1g}{2009/10/06}{(AO, 532) Both arguments of \cs{href} get sanitized}%
177 \changes{4.1g}{2009/10/07}{(AO, 525) Remove phantom paragraph above display math that is given
178 \changes{4.1g}{2009/10/07}{(AO, 539) Use of double-backslash in argument of \cs{section} gives
179 \changes{4.1n}{2009/12/05}{(AO, 569) Use of \classname{hyperref} interferes with column balanci
180 \changes{4.1n}{2009/12/06}{(AO) Incorporate change to ltmiscen.dtx v1.1i 2000/05/19}%
181 \changes{4.1n}{2009/12/09}{(AO, 569) execute \classname{atveryend}'s \cs{Call@AfterLastShipout}
182 \changes{4.1n}{2009/12/13}{(AO, 574) protect against \classname{lineno.sty}, which forces a vis
183 \changes{4.1n}{2010/01/02}{(AO, 571) Interface \cs{set@footnotewidth} for determining the set w
184 \changes{4.1n}{2010/01/02}{(AO, 571) allow split after last line of footnote}%
185 \changes{4.1n}{2010/01/06}{(AO, 572) title block footnotes numbered independently from body foo
186 \changes{4.1p}{2010/02/24}{(AO, 582) A patch of \classname{hyperref.sty} to provide backward co
187 \changes{4.1s}{2020/09/19}{(PHO) Adapt \cs{document} and \cs{enddocument} hooks to the 2020-10-
188
189 \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.

```

190 \begin{document}%
191 \expandafter\DocInput\expandafter{\jobname.dtx}%
192 \end{document}
193 %</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 $\LaTeX$ 's Required Packages

Certain packages, usually ones written by members of the  $\LaTeX$  Project itself, have been designated “required” and are distributed as part of standard  $\LaTeX$ . These packages have been placed in a privileged position vis á vis the  $\LaTeX$  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  $\LaTeX$ 's kernel procedures `\@outputdblcol`, `\@startcolumn`, and `\@makecol`.

Hereinafter follows some notes on specific  $\LaTeX$  packages.

### 5.1 array

This package alters the way tabular environments are done, therefore it could run afoul of the  $\LaTeX$  “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  $\LaTeX$  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

```

194 %⟨*package⟩
195 \def\package@name{ltxutil}%
196 \expandafter\PackageInfo\expandafter{\package@name}{%
197 Utility macros for \protect\LaTeXe,

```

```

198 by A. Ogawa (arthur_ogawa at sbcglobal.net)%
199 }%
200 %</package>

```

## 6.2 Banner and beginning of the kernel DOCSTRIP module

```
201 %<*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 202 \def\class@err#1{\ClassError{\class@name}{#1}\@eha}%
203 \def\class@warn#1{\ClassWarningNoLine{\class@name}{#1}}%
204 \def\class@info#1{\ClassInfo{\class@name}{#1}}%
205 \def\obsolete@command#1{%
206 \class@warn@end{Command \string#1\space is obsolete.^^JPlease remove from your document}%
207 \global\let#1\@empty
208 #1%
209 }%
210 \def\replace@command#1#2{%
211 \class@warn@end{Command \string#1\space is obsolete;^^JUse \string#2\space instead}%
212 \global\let#1#2%
213 #1%
214 }%
215 \def\replace@environment#1#2{%
216 \class@warn@end{Environment #1 is obsolete;^^JUse #2 instead}%
217 \glet@environment{#1}{#2}%
218 \@nameuse{#1}%
219 }%
220 \def\incompatible@package#1{%
221 \@ifpackageloaded{#1}{%
222 \def\@tempa{I cannot continue. You must remove the \string\usepackage\ statement that caused
223 \ClassError{\class@name}{The #1 package cannot be used with \class@name}%
224 \@tempa\stop
225 }{%
226 \class@info{#1 was not loaded (OK!)}%
227 }%
228 }%
229 \def\class@warn@end#1{%
230 \gappdef\class@enddocumenthook{\class@warn{#1}}%
231 }%

Give \class@name a meaning if it does not already have one.
232 \ifx\undefined\class@name
233 \def\class@name{ltxutil}%
234 \class@warn{You should define the class name before reading in this package. Using default}%
235 \fi

```

## 6.4 New Tools

```

\t@
236 \def\t@{to}%

\dimen@iii
237 \dimendef\dimen@iii\thr@@

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

\fur Analogous to \@ne, \tw@, and \thr@@.
239 \chardef\fur=4\relax
240 \chardef\cat@letter=11\relax
241 \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.
242 \def\let@environment#1#2{%
243 \expandafter\let
244 \csname#1\expandafter\endcsname\csname#2\endcsname
245 \expandafter\let
246 \csname end#1\expandafter\endcsname\csname end#2\endcsname
247 }%
248 \def\glet@environment#1#2{%
249 \global\expandafter\let
250 \csname#1\expandafter\endcsname\csname#2\endcsname
251 \global\expandafter\let
252 \csname end#1\expandafter\endcsname\csname end#2\endcsname
253 }%

\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.
254 \newcommand\tracingplain{%
255 \tracingonline\z@\tracingcommands\z@\tracingstats\z@
256 \tracingpages\z@\tracingoutput\z@\tracinglostchars\@ne
257 \tracingmacros\z@\tracingparagraphs\z@\tracingrestores\z@
258 \showboxbreadth5\showboxdepth3\relax %\errorstopmode
259 }%
260 \newcommand\traceoutput{%
261 \appdef\@resetactivechars{\showoutput}%
262 }%

\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.
263 \newcommand\say[1]{\typeout{<\noexpand#1=\meaning#1>}}%
264 \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.

```
265 \def\fullinterlineskip{\prevdepth\z@}%
```

```
\count@i
\count@ii 266 \countdef\count@i\@ne
267 \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.

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

```

294 \expandafter{#2}%
295 }%
296 \toksdef\toks@ii=\tw@

```

`\@ifxundefined` Certain utility procedures use `\@ifxundefined`, which is defined here in terms of `\@ifx`. Others use `\@ifnotrelax`, namely when the control sequence name is manufactured by the use of `\csname`.

`\@argswap` The procedures `\@argswap` and `\@argswap@val` are used to facilitate control of expansion.

```

297 \long\def\@ifxundefined#1{\@ifx{\undefined#1}}%
298 \long\def\@ifnotrelax#1#2#3{\@ifx{\relax#1}{#3}{#2}}%
299 \long\def\@argswap#1#2{#2#1}%
300 \long\def\@argswap@val#1#2{#2{#1}}%
301 \def\@ifxundefined@cs#1{\expandafter\@ifx\expandafter{\csname#1\endcsname\relax}}%

```

`\rvtx@ifformat@geq` Some changes in the L<sup>A</sup>T<sub>E</sub>X kernel requires us to conditionally define some macros depending on the version of the kernel. `\rvtx@ifformat@geq` will check if the release date of the currently-running L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> kernel is greater or equal to the argument (the argument should be in the format yyyy-mm-dd).

```

302 \ifx\IfFormatAtLeastTF\undefined
303 \def\rvtx@ifformat@geq{\ifl@t@r@fmtversion}%
304 \else
305 \let\rvtx@ifformat@geq\IfFormatAtLeastTF
306 \fi

```

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

```

307 \def\@boolean#1#2{%
308 \long\def#1{%
309 #2% \if<something>
310 \expandafter\true@sw
311 \else
312 \expandafter\false@sw
313 \fi
314 }%
315 }%
316 \def\@boole@def#1#\@boolean{#1}}% Implicit #2

```

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

```

317 \def\@booleantrue#1{\let#1\true@sw}%
318 \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 `\@ifempty`. However, for compatibility with AMSL<sup>A</sup>T<sub>E</sub>X, we must avoid the following three names: `\@ifempty`, `\@xifempty`, and `\@ifnotempty`.

```

\@ifcat
\@ifdim
\@ifeof
319 \@boole@def\@ifx#1{\ifx#1}%

```

```

\@ifhbox
\@ifhmode
\@ifinner
\@ifmmode
\@ifnum
\@ifodd
\@ifvbox
\@ifvmode
\@ifvoid

```

```

320 \@boole@def\@ifx@empty#1{\ifx\@empty#1}%
321 \@boole@def\@if@empty#1{\if!#1!}%
322 %\@boole@def\@if@sw#1{\csname if#1\endcsname}%
323 \def\@if@sw#1#2{#1\expandafter\true@sw\else\expandafter\false@sw#2}%
324 \@boole@def\@ifdim#1{\ifdim#1}%
325 \@boole@def\@ifeof#1{\ifeof#1}%
326 \@boole@def\@ifhbox#1{\ifhbox#1}%
327 \@boole@def\@ifhmode{\ifhmode}%
328 \@boole@def\@ifinner{\ifinner}%
329 \@boole@def\@ifmmode{\ifmmode}%
330 \@boole@def\@ifnum#1{\ifnum#1}%
331 \@boole@def\@ifodd#1{\ifodd#1}%
332 \@boole@def\@ifvbox#1{\ifvbox#1}%
333 \@boole@def\@ifvmode{\ifvmode}%
334 \@boole@def\@ifvoid#1{\ifvoid#1}%

```

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

```

\false@sw
335 \long\def\true@sw#1#2{#1}%
336 \long\def\false@sw#1#2{#2}%

```

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

```

\loopwhile
337 \long\def\loopuntil#1{#1}{\loopuntil{#1}}}%
338 \long\def\loopwhile#1{#1{\loopwhile{#1}}{}}}%

```

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

```

339 \def\@provide#1{%
340 \@ifx{\undefined#1}{\true@sw}{\@ifx{\relax#1}{\true@sw}{\false@sw}}}%
341 {\def#1}{\def\j@nk}%
342 }%

```

## 6.6 Begin Document Structure

The standard L<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X 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.

`\document` We begin by installing hooks into `\document` that we will manage ourselves.

The 2020-10-01 L<sup>A</sup>T<sub>E</sub>X release got a new hook management system and several new hooks (several previously provided by `etoolbox`). The one we want here is `begindocument/before`, the first thing executed by `\document`, right after ending the group started by `\begin`.

Thus, if the L<sup>A</sup>T<sub>E</sub>X kernel date is 2020-10-01 we just add to that hook, otherwise resort to the old method, patching `\document`: end the group started by `\begin`, apply our hook, and conclude our shenanigans by absorbing the first token of the expansion of `\document`, which we assume to be `\endgroup` (true until the aforementioned release).

```

343 \rvtx@ifformat@geq{2020-10-01}%
344   {%
345     \AddToHook{begindocument/before}{\document@inithook}%
346   }{%
347     \prepdef\document{%
348       \endgroup
349       \document@inithook
350       \true@sw{%
351     }%
352   }

```

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

```

353 \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`.

```

354 \appdef\document@inithook{%
355   \AtBeginDocument{\class@documenthook}%
356 }%
357 \AtEndDocument{%
358   \class@enddocumenthook
359 }%
360 \let\class@documenthook\@empty
361 \let\class@enddocumenthook\@empty

```

`\enddocument` The standard L<sup>A</sup>T<sub>E</sub>X `\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.

This approach, however, will only be done for older versions of the L<sup>A</sup>T<sub>E</sub>X kernel:

```

362 \rvtx@ifformat@geq{2020-10-01}{%

```

```

363 % <definitions for newer LaTeX later>
364 }{%
365 % <definitions for older LaTeX>
366 \def\enddocument{%

The following line from ltxutil.dtxltmiscen.dtx ‘resets \AtEndDocumentfor la-
tex/3060’.
367 \let\AtEndDocument\@firstofone
368 \@enddocumenthook
369 \@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.
370 \clear@document

We are very close to ending the TEX run, now.
371 \check@aux
372 \deadcycles\z@
373 \@@end
374 }{%
375 \def\check@aux{\do@check@aux}%
376 \def\do@check@aux{%
377 \if@sw\if@filesw\fi{%
378 \immediate\closeout\@mainaux
379 \let\@setckpt\@gobbletwo
380 \let\@newl@bel\@testdef
381 \@tempwafalse
382 \makeatletter
383 \input\jobname.aux\relax
384 }{}%
385 \@dofilelist
386 \@ifdim{\font@submax >\fontsubfuzz\relax}{%
387 \@font@warning{%
388 Size substitutions with differences\MessageBreak
389 up to \font@submax\space have occurred.\@gobbletwo
390 }%
391 }{}%
392 \@defaultsubs
393 \@refundefined
394 \if@sw\if@filesw\fi{%
395 \ifx{\@multiplelabels\relax}{%
396 \if@sw\if@tempswa\fi{%
397 \latex@warning@no@line{%
398 Label(s) may have changed.
399 Rerun to get cross-references right%
400 }%
401 }{}%
402 }{%

```



```

403   \@multiplelabels
404 }%
405 }{}%
406 }%
407 }

```

`\rvtx@enddocument@patch` For newer L<sup>A</sup>T<sub>E</sub>X we'll try to be a bit more future-proof (no miracle though). The code for `\enddocument` (in pre-2020-10-01 L<sup>A</sup>T<sub>E</sub>X) is roughly:

```

% \def\enddocument{%
%   <hooks and bookkeeping>
%   \clearpage
%   <read main .aux and final checks>
%   \@end
% }
%

```

and the patches above replace the `\clearpage` by its own `\clear@document`, and `<read main .aux and final checks>` by `\do@check@aux`, which it can later control the timing.

Now we will apply the same changes, but this time without redefining `\enddocument`: we will instead replace tokens on-the-fly, when `\enddocument` is expanded. This will grant us a slightly safer approach that won't depend so much on the internals of `\enddocument`.

This entire patch should work with the previous definition of `\enddocument` as well (except it cannot be used in the hook), but for now leave previous versions untouched.

The entire patching will reside in the `enddocument` hook:

```

408 \rvtx@ifformat>=2020-10-01{%
409   \AddToHook{enddocument}{\rvtx@enddocument@patch{}}%
410 }{}

```

This macro will be executed after `\enddocument` has expanded, so all its tokens are now exposed. Here we will assume that `\enddocument` contains the tokens `\@checkend{document}` and `\endgroup`, and use them as delimiters:

```

411 \protected\long\def\rvtx@enddocument@patch#1#2\@checkend#3{%
412   \begingroup
413   \edef\x{\detokenize{#3}}%
414   \edef\y{\detokenize{document}}%
415   \expandafter\endgroup
416   \ifx\x\y
417     \expandafter\rvtx@enddocument@patch@end
418   \else
419     \expandafter\rvtx@enddocument@patch@more
420   \fi
421   {#1#2}{#3}}
422 \def\rvtx@enddocument@patch@more#1#2{%
423   \rvtx@enddocument@patch{#1\@checkend{#2}}

```

When the `\@checkend{document}` is reached, use `\clearpage` and `\enddocument` as delimiters for the `<read main .aux and final checks>` part, and save it in `\do@check@aux`:

```
424 \long\def\rvtx@enddocument@patch@end#1#2\clearpage#3\endgroup{%
425 \def\do@check@aux{#3\endgroup}%
```

Then execute the code consumed in the previous step:

```
426 #1%
427 \@checkend{#2}%
```

Do `\clear@document` instead of `\clearpage` and `\check@aux` instead of the code grabbed.

```
428 \clear@document
429 \check@aux}
430 \def\check@aux{\do@check@aux}%
```

`\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.

```
431 \def\clear@document{%
432 \clearpage
433 \do@output@cclv{%
434 \Call@AfterLastShipout
435 }%
436 }%
437 \appdef\class@documenthook{%
438 \providecommand\Call@AfterLastShipout{}}%
439 }%
```

## 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  $\LaTeX$ '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  $\langle class \rangle$ . 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.

```

440 \def\class@extension#1#2{%
441 \IfFileExists{#1.#2}{%
442 \expandafter\class@extensionfile\csname ver@\@currname.\@currentx\endcsname{#1}#2%
443 }{%
444 \csname rtx@#1\endcsname
445 }%
446 }%
447 \def\class@extensionfile#1#2#3{%
448 \@pass@options#3\@unusedoptionlist{#2}%
449 \global\let\@unusedoptionlist\@empty
450 \expandafter\class@ext@hook\csname#2.#3-h@k\endcsname#1{#2}#3%
451 }%
452 \def\class@ext@hook#1#2#3#4{%
453 \@pushfilename@ltx
454 \makeatletter
455 \let\CurrentOption\@empty
456 \@reset@options
457 \let#1\@empty
458 \xdef\@currname{#3}%
459 \global\let\@currentx#4%
460 \global\let\@clsextension\@currentx
461 \input{#3.#4}%
462 \@ifl@ter#4{#3}#2{%
463 \class@info{Class extension later than: #2}%
464 }{%
465 \class@info{Class extension earlier: #2}%
466 \@@end
467 }%
468 #1%
469 \let#1\@undefined
470 \expandafter\@p@pfilename@ltx\@currnamestack@ltx\@nil
471 \@reset@options
472 }%

```

`\@pushfilename` But!  $\text{\LaTeX}$  does not provide for a class extension other than `.cls`, therefore we must extend  $\text{\LaTeX}$ '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.

```

473 \def\@pushfilename@ltx{%

```

```

474 \xdef\@currnamestack@ltx{%
475   {\@currname}%
476   {\@currentx}%
477   {\@clsextension}%
478   {\the\catcode'\@}%
479   \@currnamestack@ltx
480 }%
481 }%
482 \def\@p@pfilename@ltx#1#2#3#4#5\@nil{%
483   \gdef\@currname{#1}%
484   \gdef\@currentx{#2}%
485   \gdef\@clsextension{#3}%
486   \catcode'\@#4\relax
487   \gdef\@currnamestack@ltx{#5}%
488 }%
489 \global\let\@currnamestack@ltx\@empty

```

We carefully patch L<sup>A</sup>T<sub>E</sub>X 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`.

```

490 \def\flushing{%
491   \let\\\@normalcr
492   \leftskip\z@skip
493   \rightskip\z@skip
494   \@rightskip\z@skip
495   \parfillskip\@flushglue
496 }%

```

`\@centercr` The `\@centercr` command is the replacement for `\@normalcr` when setting type centered or ragged. Normally, the meaning of `\\` is `\@normalcr`, which L<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X's early failures that will never get fixed.

The following exemplar fails under L<sup>A</sup>T<sub>E</sub>X 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.

```
497 \expandafter\DeclareRobustCommand\expandafter\@centercr\expandafter{\@centercr}%
```

## 6.9 Display Math

`\eqnarray@LaTeX` Team L<sup>A</sup>T<sub>E</sub>X 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.

```
498 \def\eqnarray@LaTeX{%
499   \stepcounter{equation}%
500   \def\@currentlabel{\p@equation\theequation}%
501   \global\@eqnswtrue
502   \m@th
503   \global\@eqcnt\z@
504   \tabskip\@centering
505   \let\@@@eqnocr
506   $$\everycr{\}\halign to\displaywidth\bgroup
507     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
508     &\global\@eqcnt\@ne\hskip \tw@\arraycolsep \hfil${##}$\hfil
509     &\global\@eqcnt\tw@ \hskip \tw@\arraycolsep
510     $\displaystyle{##}$\hfil\tabskip\@centering
511     &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
512     \tabskip\z@skip
513   \cr
514 }
515 \long\def\eqnarray@fleqn@fixed{%
516   \stepcounter{equation}\def\@currentlabel{\p@equation\theequation}%
517   \global\@eqnswtrue\m@th\global\@eqcnt\z@
518   \tabskip\ltx@mathindent
519   \let\@@@eqnocr
520   \setlength\abovedisplayskip{\topsep}%
521   \ifvmode\addtolength\abovedisplayskip{\partopsep}\fi
522   \addtolength\abovedisplayskip{\parskip}%
523   \setlength\belowdisplayskip{\abovedisplayskip}%
524   \setlength\belowdisplayshortskip{\abovedisplayskip}%
525   \setlength\abovedisplayshortskip{\abovedisplayskip}%
526   $$%
527   \everycr{\}%
528   \halignt@\linewidth\bgroup
529   \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
530   &\global\@eqcnt\@ne
531   \hskip\tw@\eqncolsep
532   \hfil${}\##{}}$\hfil
533   &\global\@eqcnt\tw@
534   \hskip\tw@\eqncolsep
```

```

535   $\displaystyle{##}$\hfil\tabskip\@centering
536   &\global\@eqcnt\thr@@\hb@xt@\z@\bgroup\hss##\egroup
537   \tabskip\z@skip
538   \cr
539 }%
540 \@ifx{\eqnarray\eqnarray@LaTeX}{%
541   \class@info{Repairing broken LaTeX eqnarray}%
542   \let\eqnarray\eqnarray@fleqn@fixed
543   \newlength\eqncolsep
544   \setlength\eqncolsep\z@
545   \let\eqnarray@LaTeX\relax
546   \let\eqnarray@fleqn@fixed\relax
547 }{}%

```

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<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> means. Even though our `eqnarray` environment is modelled after `ltxutil.dtxfleqn.clo`, we must program defensively here.

```

548 \def\ltx@mathindent{\@centering}%
549 \def\set@eqnarray@skips{}%

```

`\prep@math` If we are in vertical mode when display math mode is entered (via `$$`), T<sub>E</sub>X 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 `\parskipfillskip` 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.

```

550 \def\prep@math{%
551   \@ifvmode{\everypar{\setbox\z@\lastbox}}{}%

```

```

552 }%
553 \def\prep@math@patch{%
554 \prepdef\equation{\prep@math}%
555 \prepdef\eqnarray{\prep@math}%
556 }%

```

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

```

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

```
557 \def\footnote{\@ifnextchar[\ltx@xfootnote\ltx@yfootnote}%
558 \def\ltx@xfootnote[#1]{%
559 \ltx@def@footproc\ltx@footmark[#1]%
560 \expandafter\ltx@foottext\expandafter{\the\csname c@\mpfn\endcsname}%
561 }%
562 \def\ltx@yfootnote{%
563 \ltx@stp@footproc\ltx@footmark
564 \expandafter\ltx@foottext\expandafter{\the\csname c@\mpfn\endcsname}%
565 }%
```

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.

```
566 \def\footnotemark{\@ifnextchar[\ltx@xfootmark\ltx@yfootmark}%
567 \def\ltx@xfootmark{\ltx@def@footproc\ltx@footmark}%
568 \def\ltx@yfootmark{\ltx@stp@footproc\ltx@footmark}%
569 \def\ltx@footmark#1{%
570 \leavevmode
571 \ifhmode\edef\@xsf{\the\spacefactor}\nobreak\fi
572 \begingroup
573 \expandafter\ltx@make@current@footnote\expandafter{\mpfn}{#1}%
574 \expandafter\@argswap@val\expandafter{\Hy@footnote@currentHref}{\hyper@linkstart {link}}%
575 \@makefnmark
576 \hyper@linkend
577 \endgroup
578 \ifhmode\spacefactor\@xsf\fi
579 \relax
580 }%
```

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.

```
581 \def\footnotetext{\@ifnextchar[\ltx@xfoottext\ltx@yfoottext}%
582 \def\ltx@xfoottext{\ltx@def@footproc\ltx@foottext}%
583 \def\ltx@yfoottext{\ltx@stp@footproc\ltx@foottext}%
584 \long\def\ltx@foottext#1#2{%
585 \begingroup
586 \expandafter\ltx@make@current@footnote\expandafter{\mpfn}{#1}%
587 \@footnotetext{#2}%
588 \endgroup
589 }%
```



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.

```

590 \def\ltx@def@footproc#1[#2]{%
591 \begingroup
592 \csname c@\mpfn\endcsname #2\relax
593 \unrestored@protected@xdef\@thefnmark{\thempfn}%
594 \expandafter\endgroup
595 \expandafter#1%
596 \expandafter{\the\csname c@\mpfn\endcsname}%
597 }%
598 \def\ltx@stp@footproc#1{%
599 \expandafter\stepcounter\expandafter{\@mpfn}%
600 \protected@xdef\@thefnmark{\thempfn}%
601 \expandafter#1%
602 \expandafter{\the\csname c@\mpfn\endcsname}%
603 }%

```

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.

```

604 \appdef\class@documenthook{%
605 \let\footnote@latex\footnote
606 \@ifpackageloaded{hyperref}{}-%
607 \let\H@@footnotetext\@footnotetext
608 \def\@footnotetext{\H@@footnotetext}%
609 \let\H@mpfootnotetext\@mpfootnotetext
610 \def\@mpfootnotetext{\H@mpfootnotetext}%
611 }%
612 }%

```

In the following, we must use L<sup>A</sup>T<sub>E</sub>X'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?

```

613 \def\ltx@make@current@footnote#1#2{%
614 \csname c#1\endcsname#2\relax
615 \protected@edef\Hy@footnote@currentHref{\@currentHref-#1.\csname the#1\endcsname}%
616 }%
617 \def\thempfootnote@latex{\itshape \@alph@c@mpfootnote }%
618 \def\ltx@thempfootnote{\@alph@c@mpfootnote}%
619 \@ifx{\thempfootnote\thempfootnote@latex}{%
620 \class@info{Repairing hyperref-unfriendly LaTeX definition of \string\mpfootnote}%
621 \let\thempfootnote\ltx@thempfootnote
622 }{}%

```

Note on `hyperref` compatibility: In its “Automated L<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X'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<sup>A</sup>T<sub>E</sub>X 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!

<code>\@footnotetext</code> <code>\@mpfootnotetext</code> <code>\@tpfootnotetext</code> <code>\make@footnotetext</code> <code>\set@footnotewidth</code>	<p>The two procedures <code>\@footnotetext</code> and <code>\@mpfootnotetext</code> share code. We make that explicit here.</p> <p>Note that the procedure calling <code>\make@footnotetext</code> will open a group with <code>\bgroup</code> which is then closed by <code>\minipagefootnote@drop</code>.</p>
---	---

Difference from L<sup>A</sup>T<sub>E</sub>X: here we do not set `\floatingpenalty` to infinity. Doing this must date back to a time when L<sup>A</sup>T<sub>E</sub>X 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.

```

623 \def\@makefnmark{%
624   \hbox{%
625     \@textsuperscript{%
626       \normalfont\itshape\@thefnmark
627     }%
628   }%
629 }%

630 \long\def\@footnotetext{%
631   \insert\footins\bgroup
632   \make@footnotetext
633 }%

634 \long\def\@mpfootnotetext{%
635   \minipagefootnote@pick
636   \make@footnotetext
637 }%

```

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@makefntext` instead of `\@makefntext`.

Dependency note: the `\@makefntext` 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\@makefntext[1]{%
% \parindent 1em\noindent
% \hb@xt@1.8em{\hss\@makefnmark}%
% #1%
%}%
%\def\@makefnmark{%
% \hbox{\@textsuperscript{\normalfont\@thefnmark}}%
%}%
%

```

```

638 \long\def\make@footnotetext#1{%
639   \set@footnotefont

```

As noted above, we do *not* do `\floatingpenalty \@MM`, as in standard L<sup>A</sup>T<sub>E</sub>X.

```

640   \set@footnotewidth
641   \@parboxrestore
642   \protected@edef\@currentlabel{%

```

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

```
643 \csname p@\@mpfn\endcsname\@thefnmark
644 }%
645 \color@begingroup
646 \@makefnmark{%
647 \rule\z@\footnotesep\ignorespaces#1%
```

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

```
648 \@finalstrut\strutbox\vadjust{\vskip\z@skip}%
649 }%
650 \color@endgroup
651 \minipagefootnote@drop
652 }%
```

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

```
653 \def\set@footnotefont{%
654 \reset@font\footnotesize
655 \interlinepenalty\interfootnotelinepenalty
656 \splittopskip\footnotesep
657 \splitmaxdepth\dp\strutbox
658 }%
```

`\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.

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

## 6.11 Floats

### 6.11.1 Usage notes

We extend the  $\LaTeX$  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 unfloatable. 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<sup>A</sup>T<sub>E</sub>X'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 660 \def\robustify@contents{%
661 \let \label \@gobble
662 \let \index \@gobble
663 \let \glossary \@gobble
664 \let\footnote \@gobble
665 \def\({\string\}%
666 \def\)\{\string\})%
667 \def\{\{\string\}\}%
668 }%
669 \long\def\addtocontents#1#2{%
670 \protected@write\@auxout{\robustify@contents}{\string \@writefile {#1}{#2}}%
671 }%

```

### 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<sup>A</sup>T<sub>E</sub>X. 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<sup>A</sup>T<sub>E</sub>X'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.

```
672 \def\addcontentsline#1#2#3{%
673 \addtocontents{#1}{%
674 \protect\contentsline{#2}{#3}{\thepage}{}%
675 }%
676 }%
677 \def\label#1{%
678 \@bsphack
679 \protected@write\@auxout{}{%
680 \string\newlabel{#1}{\@currentlabel}{\thepage}{}{}}%
681 }%
682 \@esphack
683 }%
684 \def\ltx@contentsline#1{%
685 \expandafter\@ifnotrelax\csname l@#1\endcsname{}{%
686 \expandafter\let\csname l@#1\endcsname\@gobbletwo
687 }%
688 \contentsline@latex{#1}%
689 }%
690 \appdef\document@inithook{%
691 \let\contentsline@latex\contentsline
692 \let\contentsline\ltx@contentsline
693 }%
```

### 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<sup>A</sup>T<sub>E</sub>X syntax at this point: If a caption is used below a table, then put out the footnotes before the caption.

```
694 \appdef\class@documenthook{%
695 \prepdef\caption{\minipagefootnote@here}%
696 }%
```

Note on `hyperref` compatibility: this change to the `\caption` command is compatible with the “Automated L<sup>A</sup>T<sub>E</sub>X 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 697 \def\minipagefootnote@init{%
\minipagefootnote@drop 698 \setbox\@mpfootins\box\voidb@x
699 }%
700 \def\minipagefootnote@pick{%
701 \global\setbox\@mpfootins\vbox\bgroup
702 \unvbox\@mpfootins
703 }%
704 \def\minipagefootnote@drop{%
705 \egroup
706 }%
707 \def\minipagefootnote@here{%
708 \par
709 \@ifvoid\@mpfootins{}{%
710 \vskip\skip\@mpfootins
711 \fullinterlineskip
712 \@ifinner{%
713 \vtop{\unvcopy\@mpfootins}%
714 {\setbox\z@\lastbox}%
715 }{}%
716 \unvbox\@mpfootins
717 }%
718 }%
719 \def\minipagefootnote@foot{%
720 \@ifvoid\@mpfootins{}{%
721 \insert\footins\bgroup\unvbox\@mpfootins\egroup
722 }%
723 }%
724 \def\endminipage{%
725 \par
726 \unskip
727 \minipagefootnote@here
728 \@minipagefalse %% added 24 May 89

```

```

729 \color@endgroup
730 \egroup
731 \expandafter\@iiparbox\@mpargs{\unvbox\@tempboxa}%
732 }%

```

`\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.

```
733 \@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.

```

734 \let\@xfloat@LaTeX\@xfloat
735 \def\@xfloat#1[#2]{%
736   \@xfloat@prep
737   \@nameuse{fp@proc#2}%
738   \floats@sw{\@xfloat@LaTeX{#1}[#2]}\@xfloat@anchored{#1}[]}%
739 }%
740 \def\@xfloat@prep{%
741   \ltx@footnote@pop
742   \def\@mpfn{mpfootnote}%
743   \def\thempfn{\thempfootnote}%
744   \c@mpfootnote\z@
745   \let\H@footnotetext\H@mpfootnotetext
746 }%
747 \let\ltx@footnote@pop\@empty
748 \def\@xfloat@anchored#1[#2]{%
749   \def\@capttype{#1}%
750   \begin@float@pagebreak
751   \let\end@float\end@float@anchored
752   \let\end@dblfloat\end@float@anchored
753     \hsize\columnwidth
754     \@parboxrestore
755     \floatboxreset
756   \minipagefootnote@init
757 }%
758 \def\end@float@anchored{%
759   \minipagefootnote@here
760   \par\vskip\z@skip
761   \par
762   \end@float@pagebreak
763 }%
764 \def\begin@float@pagebreak{\par\addvspace\intextsep}%
765 \def\end@float@pagebreak{\par\addvspace\intextsep}%
766 \def\@mpmakefntext#1{%
767   \parindent=1em

```



```

768 \noindent
769 \hb@xt@1em{\hss\@makefnmark}%
770 #1%
771 }%

```

### 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.

```

772 \def\do@if@floats#1#2{%
773 \floats@sw}{-%

```

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

```

774 \expandafter\newwrite
775           \csname#1write\endcsname
776 \expandafter\def
777           \csname#1@stream\endcsname{\jobname#2}%
778 \expandafter\immediate
779 \expandafter\openout
780           \csname#1write\endcsname
781           \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`.

```

782 \@ifxundefined\@float@LaTeX{%
783 \let\@float@LaTeX\@float
784 \let\@dblfloat@LaTeX\@dblfloat
785 \let\@float\write@float
786 \let\@dblfloat\write@floats
787 }{%
788 \let@environment{#1@float}{#1}%
789 \let@environment{#1@floats}{#1*}%
790 \@ifxundefined@cs{#1@write}{-%
791 \let@environment{#1}{#1@write}%
792 }%
793 }%
794 }%

```

`\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.

```

795 \def\triggerpar{\leavevmode\@par}%
796 \def\onepage{\def\begin@float@pagebreak{\newpage}\def\end@float@pagebreak{\newpage}}%
797 \def\print@float#1#2{%

```

```

798 \lengthcheck@sw{%
799 \total@float{#1}%
800 }{%
801 \ifundefined@cs{#1write}{}{%
802 \begingroup
803 \@booleanfalse\floats@sw
804 #2%
805 \raggedbottom
806 \def\array@default{v}% floats must
807 \let\@float\@float@LaTeX
808 \let\@dblfloat\@dblfloat@LaTeX
809 \let\trigger@float@par\triggerpar
810 \let@environment{#1}{#1@float}%
811 \let@environment{#1*}{#1@floats}%
812 \expandafter\prepdef\csname#1\endcsname{\trigger@float@par}%
813 \expandafter\prepdef\csname#1*\endcsname{\trigger@float@par}%
814 \@namedef{fps@#1}{h!}%
815 \expandafter\immediate
816 \expandafter\closeout
817 \csname#1write\endcsname
818 \everypar{%
819 \global\let\trigger@float@par\relax
820 \global\everypar{}\setbox\z@\lastbox
821 \ifundefined@cs{#1sname}{}{%
822 \begin@float@pagebreak
823 \expandafter\section
824 \expandafter*%
825 \expandafter{%
826 \csname#1sname\endcsname
827 }%
828 }%
829 }%
830 \input{\csname#1@stream\endcsname}%
831 \endgroup
832 \global\expandafter\let\csname#1write\endcsname\relax
833 }%
834 }%

```

`\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  $2^2$ .

```

835 \chardef\@xvi=16\relax
836 \mathchardef\@twopowerfourteen="4000
837 \mathchardef\@twopowertwo="4
838 \def\tally@float#1{%
839 \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.

```

840 \@tempcnta\count\@currbox
841 \divide\@tempcnta\@xxxii
842 \multiply\@tempcnta\@xxxii
843 \advance\count\@currbox-\@tempcnta
844 \divide\@tempcnta\@xxxii
845 \ifnum{\count\@currbox>\@xvi}{%
846 \advance\count\@currbox-\@xvi\@booleantrue\@temp@sw
847 }{%
848 \@booleanfalse\@temp@sw
849 }%

```

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

```

850 \show@box@size@sw{%
851 \class@info{Float #1
852 (\the\@tempcnta)[\@temp@sw{16+}{}\the\count\@currbox]^^J%
853 (\the\ht\@currbox+\the\dp\@currbox)X\the\wd\@currbox
854 }%
855 }-}%
856 \endgroup

```

Here we tally the height of this float object.

```

857 \expandafter\let
858 \expandafter\@tempa
859 \csname fbox@\csname ftype@#1\endcsname\endcsname
860 \ifnotrelax\@tempa{%
861 \ifhbox\@tempa{%
862 \setbox\@tempboxa\vbox{\unvcopy\@currbox\hrule}%
863 \dimen@ht\@tempboxa
864 \divide\dimen@\@twopowerfourteen
865 \ifdim{\wd\@tempboxa<\textwidth}{%
866 \advance\dimen@ht\@tempa
867 \global\ht\@tempa\dimen@
868 }{%
869 \advance\dimen@dp\@tempa
870 \global\dp\@tempa\dimen@
871 }%
872 }-}%
873 }-}%
874 }%
875 \def\total@float#1{%
876 \expandafter\let
877 \expandafter\@tempa
878 \csname fbox@\csname ftype@#1\endcsname\endcsname
879 \ifnotrelax\@tempa{%
880 \ifhbox\@tempa{%
881 \@tempdima\the\ht\@tempa\divide\@tempdima\@twopowertwo\@tempcnta\@tempdima
882 \@tempdimb\the\dp\@tempa\divide\@tempdimb\@twopowertwo\@tempcntb\@tempdimb

```

```

883   \class@info{Total #1: Column(\the\@tempcnta pt), Page(\the\@tempcnta pt)}%
884   }{}%
885 }{}%
886 }%

```

`\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 user-level environment, therefore it is properly not defined with `\newenvironment`.

```

887 \def\write@float#1{\write@@float{#1}{#1}}%
888 \def\endwrite@float{\@Esphack}%
889 \def\write@floats#1{\write@@float{#1*}{#1}}%
890 \def\endwrite@floats{\@Esphack}%

```

`\write@@float`

```

891 \def\write@@float#1#2{%
892   \ifhmode
893     \@bsphack
894   \fi
895   \chardef\@tempc\csname#2write\endcsname
896   \toks@{\begin{#1}}%
897   \def\@tempb{#1}%
898   \expandafter\let\csname end#1\endcsname\endwrite@float
899   \catcode'\^M\active
900   \@makeother\{\@makeother\}\@makeother\%
901   \write@floatline
902 }%

```

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

```

903 \begingroup
904 \catcode'\[the\catcode'\{\catcode'\}\the\catcode'\}\@makeother\{\@makeother\}%
905 \gdef\float@end@tag#1\end{#2}#3\nul[%
906   \def\@tempa[#2]%
907   \@ifx[\@tempa\@tempb][\end{#2}][\write@floatline]%
908 ]%
909 \obeylines%
910 \gdef\write@floatline#1^M[%
911   \begingroup%
912   \newlinechar'\^M%
913   \toks@\expandafter[the\toks@#1]\immediate\write\@tempc[the\toks@]%
914   \endgroup%
915   \toks@[]%
916   \float@end@tag#1\end{\}\@nul%
917 ]%
918 \endgroup

```

## 6.12 Counters

The following definitions override those of the L<sup>A</sup>T<sub>E</sub>X kernel, providing for a greater range of inputs.

```
919 \def\@alph#1{\ifcase#1\or a\or b\or c\or d\else\@ialph{#1}\fi}
920 \def\@ialph#1{\ifcase#1\or \or \or \or \or e\or f\or g\or h\or i\or j\or
921 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
922 y\or z\or aa\or bb\or cc\or dd\or ee\or ff\or gg\or hh\or ii\or jj\or
923 kk\or ll\or mm\or nn\or oo\or pp\or qq\or rr\or ss\or tt\or uu\or
924 vv\or ww\or xx\or yy\or zz\else\@ctrerr\fi}
```

## 6.13 Customization of Sections

Patch the standard L<sup>A</sup>T<sub>E</sub>X 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.

```
925 \def\@startsection#1#2#3#4#5#6{%
926 \@startsection@hook
927 \if@noskipsec \leavevmode \fi
928 \par
929 \@tempskipa #4\relax
930 \@afterindenttrue
931 \ifdim \@tempskipa <\z@
932 \@tempskipa -\@tempskipa \@afterindentfalse
933 \fi
934 \if@nobreak
935 \everypar{}}%
```

```

936 \else
937 \addpenalty\@secpenalty\addvspace\@tempskipa
938 \fi
939 \@ifstar
940 {\@dblarg{\@sect@ltx{#1}{#2}{#3}{#4}{#5}{#6}}}%
941 {\@dblarg{\@sect@ltx {#1}{#2}{#3}{#4}{#5}{#6}}}%
942 }%
943 \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 `\@hangfrom` 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 `\@hangfrom` and `\@runin@to` can be used to process the argument of the head. The head can define, e.g., `\@hangfrom@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<sup>A</sup>T<sub>E</sub>X procedure.

```

944 \class@info{Repairing broken LaTeX \string\@sect}%
945 \def\@sect@ltx#1#2#3#4#5#6[#7]#8{%
946 \ifnum{#2}>\c@secnumdepth}{%
947 \def\H@svsec{\phantomsection}%
948 \let\@svsec\@empty
949 }{%
950 \H@refstepcounter{#1}%
951 \def\H@svsec{%
952 \phantomsection
953 }%
954 \protected@edef\@svsec{{#1}}%
955 \@ifundefined{@#1cntformat}{%
956 \prepdef\@svsec\@secntformat
957 }{%
958 \expandafter\prepdef
959 \expandafter\@svsec
960 \csname @#1cntformat\endcsname
961 }%
962 }%
963 \@tempskipa #5\relax
964 \@ifdim{\@tempskipa>\z@}{%
965 \begingroup
966 \interlinepenalty \@M
967 #6{%
968 \@ifundefined{@hangfrom@#1}{\@hang@from}{\csname @hangfrom@#1\endcsname}}%
969 {\hskip#3\relax\H@svsec}{\@svsec}{#8}%
970 }%
971 \@@par
972 \endgroup

```

```

973 \ifundefined{#1mark}{\@gobble}{\csname #1mark\endcsname}{#7}%
974 \addcontentsline{toc}{#1}{%
975 \ifnum{#2}>\c@secnumdepth}{%
976 \protect\numberline{}}%
977 }{%
978 \protect\numberline{\csname the#1\endcsname}%
979 }%
980 #8}%
981 }{%
982 \def\@svsechd{%
983 #6{%
984 \ifundefined{@runin@to@#1}{\@runin@to}{\csname @runin@to@#1\endcsname}%
985 {\hskip#3\relax\H@svsec}{\@svsec}{#8}%
986 }%
987 \ifundefined{#1mark}{\@gobble}{\csname #1mark\endcsname}{#7}%
988 \addcontentsline{toc}{#1}{%
989 \ifnum{#2}>\c@secnumdepth}{%
990 \protect\numberline{}}%
991 }{%
992 \protect\numberline{\csname the#1\endcsname}%
993 }%
994 #8}%
995 }%
996 }%
997 \@xsect{#5}%
998 }%
999 \def\@changefrom#1#2#3{\@changefrom{#1#2}#3}%
1000 \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 `\@changefrom` 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`.

```

1001 \def\@ssect@ltx#1#2#3#4#5#6[#7]#8{%
    Removed \def\@currentlabelname{#8}
1002 \def\H@svsec{\phantomsection}%
1003 \@tempskipa #5\relax
1004 \@ifdim{\@tempskipa}>\z@}{%
1005 \begingroup
1006 \interlinepenalty \@M
1007 #6{%
1008 \ifundefined{@hangfroms@#1}{\@hangfroms}{\csname @hangfroms@#1\endcsname}%
    Removed {\hskip#3\relax\H@svsec}{\Sectionformat{#8}{#1}}

```

```

1009     {\hskip#3\relax\H@svsec}{#8}%
1010     }%
1011     \@@par
1012     \endgroup
1013     \@ifundefined{#1smark}{\@gobble}{\csname #1smark\endcsname}{#7}%
1014     \addcontentsline{toc}{#1}{\protect\numberline{#8}}%
1015     }{%
1016     \def\@svsechd{%
1017         #6%
1018         \@ifundefined{@runin@tos#1}{\@runin@tos}{\csname @runin@tos#1\endcsname}%
1019         Removed {\hskip#3\relax\H@svsec}{\Sectionformat{#8}{#1}}
1020     }%
1021     \@ifundefined{#1smark}{\@gobble}{\csname #1smark\endcsname}{#7}%
1022     \addcontentsline{toc}{#1}{\protect\numberline{#8}}%
1023     }%
1024     }%
1025     \@xsect{#5}%
1026 }%
1027 \def\@hang@froms#1#2{#1#2}%
1028 \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 type-setting 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.

```

1029 \def\init@hyperref{%
1030     \providecommand\phantomsection{}%
1031     \providecommand\hyper@makecurrent[1]{}%
1032     \providecommand\Hy@raisedlink[1]{}%
1033     \providecommand\hyper@anchorstart[1]{}%
1034     \providecommand\hyper@anchorend{}%
1035     \providecommand\hyper@linkstart[2]{}%
1036     \providecommand\hyper@linkend{}%
1037     \providecommand\@currentHref{}%
1038 }%
1039 \let\H@refstepcounter\refstepcounter
1040 \appdef\document@inithook{%
1041     \init@hyperref
1042 }%

```



`\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<sup>A</sup>T<sub>E</sub>X default.

```
1043 \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 `\endarray` `\endarray` 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<sup>A</sup>T<sub>E</sub>X array and tabular environments. I know of none that do so.

```
1044 \appdef\document@inithook{%
1045   \ifpackageloaded{array}{\switch@array}{\switch@tabular}%
1046   \prepdef\endtabular{\endtabular@hook}%
1047   \@provide\endtabular@hook}%
1048   \prepdef\endarray{\endarray@hook}%
1049   \@provide\endarray@hook}%
1050   \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.

```
1051 \prepdef\@tabular{\tabular@hook}%
1052 \@provide\@tabular@hook}%
1053 }%
```

`\switch@tabular` The two procedures `\switch@tabular` and `\switch@array` apply needed patches to the various tabular procedures, the former applying to the L<sup>A</sup>T<sub>E</sub>X kernel, the latter to the required `array` package (and to the number of other required packages that load it).

```
1054 \def\switch@tabular{%
1055   \let\@array@sw\@array@sw@array
1056   \@ifx{\@array\@array@LaTeX}{%
1057     \@ifx{\multicolumn\multicolumn@LaTeX}{%
1058       \@ifx{\@tabular\@tabular@LaTeX}{%
1059         \@ifx{\@tabarray\@tabarray@LaTeX}{%
1060           \@ifx{\array\array@LaTeX}{%
1061             \@ifx{\endarray\endarray@LaTeX}{%
1062               \@ifx{\endtabular\endtabular@LaTeX}{%
1063                 \@ifx{\@mkpream\@mkpream@LaTeX}{%
1064                   \@ifx{\@addamp\@addamp@LaTeX}{%
```

```

1065 \@ifx{\@arrayacol\@arrayacol@LaTeX}{%
1066 \@ifx{\@tabacol\@tabacol@LaTeX}{%
1067 \@ifx{\@arrayclassz\@arrayclassz@LaTeX}{%
1068 \@ifx{\@tabclassiv\@tabclassiv@LaTeX}{%
1069 \@ifx{\@arrayclassiv\@arrayclassiv@LaTeX}{%
1070 \@ifx{\@tabclassz\@tabclassz@LaTeX}{%
1071 \@ifx{\@classv\@classv@LaTeX}{%
1072 \@ifx{\hline\hline@LaTeX}{%
1073 \@ifx{\@tabularcr\@tabularcr@LaTeX}{%
1074 \@ifx{\@xtabularcr\@xtabularcr@LaTeX}{%
1075 \@ifx{\@xargarraycr\@xargarraycr@LaTeX}{%
1076 \@ifx{\@yargarraycr\@yargarraycr@LaTeX}{%
1077 \true@sw
1078 }{%
1079 \false@sw
1080 }%
1081 }{%
1082 \false@sw
1083 }%
1084 }{%
1085 \false@sw
1086 }%
1087 }{%
1088 \false@sw
1089 }%
1090 }{%
1091 \false@sw
1092 }%
1093 }{%
1094 \false@sw
1095 }%
1096 }{%
1097 \false@sw
1098 }%
1099 }{%
1100 \false@sw
1101 }%
1102 }{%
1103 \false@sw
1104 }%
1105 }{%
1106 \false@sw
1107 }%
1108 }{%
1109 \false@sw
1110 }%
1111 }{%
1112 \false@sw
1113 }%
1114 }{%

```

```

1115         \false@sw
1116     }%
1117 }{%
1118     \false@sw
1119     }%
1120 }{%
1121     \false@sw
1122     }%
1123 }{%
1124     \false@sw
1125     }%
1126 }{%
1127     \false@sw
1128     }%
1129 }{%
1130     \false@sw
1131     }%
1132 }{%
1133     \false@sw
1134     }%
1135 }{%
1136     \false@sw
1137     }%
1138 }{%
1139     \false@sw
1140     }%
1141 {%
1142 \class@info{Patching LaTeX tabular.}%
1143 }{%
1144 \class@info{Unrecognized LaTeX tabular. Please update this document class! (Proceeding with f
1145 }%
1146 \let\@array\@array@ltx
1147 \let\multicolumn\multicolumn@ltx
1148 \let\@tabular\@tabular@ltx
1149 \let\@tabarray\@tabarray@ltx
1150 \let\array\array@ltx
1151 \let\endarray\endarray@ltx
1152 \let\endtabular\endtabular@ltx
1153 \let\@mkpream\@mkpream@ltx
1154 \let\@addamp\@addamp@ltx
1155 \let\@arrayacol\@arrayacol@ltx
1156 \let\@tabacol\@tabacol@ltx
1157 \let\@arrayclassz\@arrayclassz@ltx
1158 \let\@tabclassiv\@tabclassiv@ltx
1159 \let\@arrayclassiv\@arrayclassiv@ltx
1160 \let\@tabclassz\@tabclassz@ltx
1161 \let\@classv\@classv@ltx
1162 \let\hline\hline@ltx
1163 \let\@tabularcr\@tabularcr@ltx
1164 \let\@xtabularcr\@xtabularcr@ltx

```

```

1165 \let\@xargarraycr\@xargarraycr@ltx
1166 \let\@yargarraycr\@yargarraycr@ltx
1167 }%

1168 \def\switch@array{%
1169 \@ifpackageloaded{colortbl}{\let\switch@array@info\colortbl@message}{\let\switch@array@info\ar
1170 \let\@array@sw\@array@sw@LaTeX
1171 \@ifx{\@array\@array@array}{%
1172 \@ifx{\@tabular\@tabular@array}{%
1173 \@ifx{\@tabarray\@tabarray@array}{%
1174 \@ifx{\array\array@array}{%
1175 \@ifx{\endarray\endarray@array}{%
1176 \@ifx{\endtabular\endtabular@array}{%
1177 \@ifx{\@mkpream\@mkpream@array}{%
1178 \@ifx{\@classx\@classx@array}{%
1179 \@ifx{\insert@column\insert@column@array}{%
1180 \@ifx{\@arraycr\@arraycr@array}{%
1181 \@ifx{\@xarraycr\@xarraycr@array}{%
1182 \@ifx{\@xargarraycr\@xargarraycr@array}{%
1183 \@ifx{\@yargarraycr\@yargarraycr@array}{%
1184 \true@sw
1185 }{%
1186 \false@sw
1187 }%
1188 }{%
1189 \false@sw
1190 }%
1191 }{%
1192 \false@sw
1193 }%
1194 }{%
1195 \false@sw
1196 }%
1197 }{%
1198 \false@sw
1199 }%
1200 }{%
1201 \false@sw
1202 }%
1203 }{%
1204 \false@sw
1205 }%
1206 }{%
1207 \false@sw
1208 }%
1209 }{%
1210 \false@sw
1211 }%
1212 }{%
1213 \false@sw

```

```

1214 }%
1215 }{%
1216 \false@sw
1217 }%
1218 }{%
1219 \false@sw
1220 }%
1221 }{%
1222 \false@sw
1223 }{%
1224 \class@info{Patching array package.}%
1225 }{%
1226 \switch@array@info
1227 }%
1228 \let\@array \@array@array@new
1229 \let\@@array \@array % Così fan tutti
1230 \let\@tabular \@tabular@array@new
1231 \let\@tabarray \@tabarray@array@new
1232 \let\array \array@array@new
1233 \let\endarray \endarray@array@new
1234 \let\endtabular\endtabular@array@new
1235 \let\@mkpream \@mkpream@array@new
1236 \let\@classx \@classx@array@new
1237 \let\@arrayacol\@arrayacol@ltx
1238 \let\@tabacol \@tabacol@ltx
1239 \let\insert@column\insert@column@array@new
1240 \expandafter\let\csname endtabular*\endcsname\endtabular % Così fan tutti
1241 \let\@arraycr \@arraycr@new
1242 \let\@xarraycr \@xarraycr@new
1243 \let\@xargarraycr\@xargarraycr@new
1244 \let\@yargarraycr\@yargarraycr@new
1245 }%
1246 \def\array@message{%
1247 \class@info{Unrecognized array package. Please update this document class! (Proceeding with fi
1248 }%
1249 \def\colortbl@message{%
1250 \class@info{colortbl package is loaded. (Proceeding with fingers crossed.)}%
1251 }%

```

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

```

1252 \def\@array@sw@LaTeX{\@ifx{\@tabularcr}}%
1253 \def\@array@sw@array{\@ifx{\dollarbegin\begin\group}}%

```

**\@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.

```

1254 \def\@tabular@LaTeX{%
1255 \leavevmode
1256 \hbox\bgroup$%
1257 \let\@acol\@tabacol
1258 \let\@classz\@tabclassz
1259 \let\@classiv\@tabclassiv
1260 \let\\\@tabularcr
1261 \@tabarray
1262 }%
1263 \def\@tabular@ltx{%
1264 \let\@acoll\@tabacoll
1265 \let\@acolr\@tabacolr
1266 \let\@acol\@tabacol
1267 \let\@classz\@tabclassz
1268 \let\@classiv\@tabclassiv
1269 \let\\\@tabularcr
1270 \@tabarray
1271 }%
1272 \def\@tabular@array{%
1273 \leavevmode
1274 \hbox\bgroup$%
1275 \colsep\tabcolsep
1276 \let\d@llarbegin\beginngroup
1277 \let\d@llarend\endngroup
1278 \@tabarray
1279 }%
1280 \def\@tabular@array@new{%
1281 \let\@acoll\@tabacoll
1282 \let\@acolr\@tabacolr
1283 \let\@acol\@tabacol
1284 \let\d@llarbegin\beginngroup
1285 \let\d@llarend\endngroup
1286 \@tabarray
1287 }%

```

`\@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<sup>A</sup>T<sub>E</sub>X. 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.

```

1288 \def\@tabarray@LaTeX{%
1289 \m@th\@ifnextchar[\@array{\@array[c]}%
1290 }%
1291 \def\@tabarray@ltx{%

```

```

1292 \m@th@ifnextchar[\@array{\expandafter\@array\expandafter[\array@default]}%
1293 }%
1294 \def\@tabarray@array{%
1295 \ifnextchar[{\@array}{\@array[c]}%
1296 }%
1297 \def\@tabarray@array@new{%
1298 \ifnextchar[{\@array}{\expandafter\@array\expandafter[\array@default]}%
1299 }%

```

`\@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.

`\@xtabularcr` 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 `\@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 `\@eqnrcr`: 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.

```

1300 \newcount\intertabularlinepenalty
1301 \intertabularlinepenalty=100
1302 \newcount\@tbpen
1303 \apptdef\samepage{\intertabularlinepenalty\@M}%
1304 \def\@tabularcr@LaTeX{\ifnum 0='}\fi \ifstar \@xtabularcr \@xtabularcr}%
1305 \def\@tabularcr<ltx{\ifnum 0='}\fi \ifstar {\global \@tbpen \@M \@xtabularcr }\global \@tbpen
1306 \def\@xtabularcr@LaTeX{\ifnextchar [\@argtabularcr {\ifnum 0='{ \fi }\cr }}%
1307 \def\@xtabularcr<ltx{\ifnextchar [\@argtabularcr {\ifnum 0='{ \fi }\cr \noalign {\penalty \@tbpen
1308 \def\@xargarraycr@LaTeX#1{\@tempdima #1\advance \@tempdima \dp \@arstrutbox \vrule \@height \z@ \
1309 \def\@xargarraycr<ltx#1{\@tempdima #1\advance \@tempdima \dp \@arstrutbox \vrule \@height \z@ \
1310 \def\@yargarraycr@LaTeX#1{\cr \noalign {\vskip #1}}%
1311 \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.

```

1312 \def\@arraycr@array{%
1313 \relax
1314 \iffalse{\fi\ifnum 0='}\fi
1315 \@ifstar \@xarraycr \@xarraycr
1316 }%
1317 \def\@arraycr@new{%
1318 \relax

```

```

1319 \iffalse{\fi\ifnum 0='}\fi
1320 \@ifstar {\global \@tbpen \@M \@xarraycr }{\global \@tbpen \intertabularlinepenalty \@xarraycr
1321 }%
1322 \def\@xarraycr@array{%
1323 \@ifnextchar [%]
1324 \@argarraycr {\ifnum 0='{}\fi\cr}%
1325 }%
1326 \def\@xarraycr@new{%
1327 \@ifnextchar [%]
1328 \@argarraycr {\ifnum 0='{}\fi\cr \noalign {\penalty \@tbpen }}%
1329 }%
1330 \def\@xargarraycr@array#1{%
1331 \unskip
1332 \@tempdima #1\advance\@tempdima \dp\@arstrutbox
1333 \vrule \@depth\@tempdima \@width\z@
1334 \cr
1335 }%
1336 \def\@xargarraycr@new#1{%
1337 \unskip
1338 \@tempdima #1\advance\@tempdima \dp\@arstrutbox
1339 \vrule \@depth\@tempdima \@width\z@
1340 \cr
1341 \noalign {\penalty \@tbpen }}%
1342 }%
1343 \def\@yargarraycr@array#1{%
1344 \cr
1345 \noalign{\vskip #1}%
1346 }%
1347 \def\@yargarraycr@new#1{%
1348 \cr
1349 \noalign{\penalty \@tbpen \vskip #1}%
1350 }%

```

`\array` We provide old and new versions of the `\array` procedure for both L<sup>A</sup>T<sub>E</sub>X 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.

```

1351 \def\array@LaTeX{%
1352 \let\@acol\@arrayacol
1353 \let\@classz\@arrayclassz
1354 \let\@classiv\@arrayclassiv
1355 \let\@arraycr
1356 \let\@halignto\@empty
1357 \let\@tabarray
1358 }%
1359 \def\array@ltx{%
1360 \@ifmmode{\@badmath}%
1361 \let\@acoll\@arrayacol
1362 \let\@acolr\@arrayacol
1363 \let\@acol\@arrayacol

```



```

1364 \let\@classz\@arrayclassz
1365 \let\@classiv\@arrayclassiv
1366 \let\@arraycr
1367 \let\@halignto\@empty
1368 \@tabarray
1369 }%
1370 \def\array@array{%
1371 \col@sep\arraycolsep
1372 \def\d@llarbegin{$}\let\d@llarend\d@llarbegin\gdef\@halignto{}%
1373 \@tabarray
1374 }
1375 \def\array@array@new{%
1376 \@ifmode{}{\@badmath$}%
1377 \let\@acoll\@arrayacol
1378 \let\@acolr\@arrayacol
1379 \let\@acol\@arrayacol

Removed: \let\col@sep\@undefined
1380 \def\d@llarbegin{$}%
1381 \let\d@llarend\d@llarbegin
1382 \gdef\@halignto{}%
1383 \@tabarray
1384 }%

```

`\@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.

```

1385 \def\@array@LaTeX[#1]#2{%
1386 \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi\fi
1387 \bgroup
1388 \setbox\@arstrutbox\hbox{%
1389 \vrule \@height\arraystretch\ht\strutbox
1390 \quad \@depth\arraystretch \dp\strutbox
1391 \quad \@width\z@}%
1392 \@mkpream{#2}%
1393 \edef\@preamble{%
1394 \ialign \noexpand\@halignto
1395 \bgroup \@arstrut \@preamble \tabskip\z@skip \cr}%
1396 \let\@startpbox\@startpbox \let\@endpbox\@endpbox
1397 \let\tabularnewline\%
1398 \let\par\@empty
1399 \let\@sharp##%
1400 \set@typeset@protect
1401 \lineskip\z@skip\baselineskip\z@skip

```

```

1402 \ifhmode \@preamerr\z@ \@par\fi
1403 \@preamble
1404 }%
1405 \def\@array@ltx[#1]#2{%
1406 \@nameuse{\@array@align@#1}%
1407 \set@arstrutbox
1408 \@mkpream{#2}%
1409 \prepdef\@preamble{%
1410 \tabskip\@tabmid@skip
1411 \@arstrut
1412 }%
1413 \appdef\@preamble{%
1414 \tabskip\@tabright@skip
1415 \cr
1416 \array@row@pre
1417 }%
1418 % \let\@startpbox\@startpbox
1419 % \let\@endpbox\@endpbox
1420 \let\@tabularnewline\@
1421 \let\@par\@empty
1422 \let\@sharp##%
1423 \set@typeset@protect
1424 \lineskip\z@skip\baselineskip\z@skip
1425 \tabskip\@tableft@skip\relax
1426 \ifhmode \@preamerr\z@ \@par\fi
1427 \everycr{}%
1428 \expandafter\halign\expandafter\@halignto\expandafter\bgroup\@preamble
1429 }%
1430 %
1431 \def\set@arstrutbox{%
1432 \setbox\@arstrutbox\hbox{%
1433 \vrule \@height\arraystretch\ht\strutbox
1434 \quad \@depth\arraystretch\dp\strutbox
1435 \quad \@width\z@
1436 }%
1437 }%

```

\@array@array

```

1438 \def\@array@array[#1]#2{%
1439 \@tempdima \ht \strutbox
1440 \advance \@tempdima by\extrarowheight
1441 \setbox \@arstrutbox \hbox{\vrule
1442 \quad \@height \arraystretch \@tempdima
1443 \quad \@depth \arraystretch \dp \strutbox
1444 \quad \@width \z@}%
1445 \beginngroup
1446 \@mkpream{#2}%
1447 \xdef\@preamble{\noexpand \ialign \@halignto
1448 \quad \bgroup \@arstrut \@preamble
1449 \quad \tabskip \z@ \cr}%

```

```

1450 \endgroup
1451 \@arrayleft
1452 \if #1t\top \else \if#1b\vbox \else \vcenter \fi \fi
1453 \bgroup
1454 \let \@sharp ##\let \protect \relax
1455 \lineskip \z@
1456 \baselineskip \z@
1457 \m@th
1458 \let\\\@arraycr \let\tabularnewline\\\let\par\@empty \@preamble
1459 }%
1460 \def\@array@array@new[#1]#2{%
1461 \@tempdima\ht\strutbox
1462 \advance\@tempdima by\extrarowheight
1463 \setbox\@arstrutbox\hbox{%
1464 \vrule \@height\arraystretch\@tempdima
1465 \@depth \arraystretch\dp\strutbox
1466 \@width \z@
1467 }%
1468 \begingroup
1469 \mkpream{#2}%
1470 \xdef\@preamble{\@preamble}%
1471 \endgroup
1472 \prepdef\@preamble{%
1473 \tabskip\@tabmid@skip
1474 \@arstrut
1475 }%
1476 \appdef\@preamble{%
1477 \tabskip\@tabright@skip
1478 \cr
1479 \array@row@pre
1480 }%
1481 \@arrayleft
1482 \@nameuse{\@array@align@#1}%
1483 \m@th
1484 \let\\\@arraycr
1485 \let\tabularnewline\%
1486 \let\par\@empty
1487 \let\@sharp##%
1488 \set@typeset@protect
1489 \lineskip\z@\baselineskip\z@
1490 \tabskip\@tableft@skip
1491 \everycr{}%
1492 \expandafter\halign\expandafter\@halignto\expandafter\bgroup\@preamble
1493 }%

```

`\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`.

```

1494 \def\endarray@LaTeX{%
1495 \crr\egroup\egroup

```

```

1496 }%
1497 \def\endarray@ltx{%
1498 \crr\array@row@pst\egroup\egroup
1499 }%
1500 \def\endarray@array{%
1501 \crr \egroup \egroup \@arrayright \gdef\@preamble{%
1502 }%
1503 \def\endarray@array@new{%
1504 \crr\array@row@pst\egroup\egroup % Same as \endarray@ltx
1505 \@arrayright
1506 \global\let\@preamble\@empty
1507 }%

```

`\endtabular`

```

1508 \def\endtabular@LaTeX{%
1509 \crr\egroup\egroup $\egroup
1510 }%
1511 \def\endtabular@ltx{%
1512 \endarray
1513 }%
1514 \def\endtabular@array{%
1515 \endarray $\egroup
1516 }%
1517 \def\endtabular@array@new{%
1518 \endarray
1519 }%

```

`endtabular*` Here we provide a proper definition for the star-form of `\end{endtabular}`. It is one of the enduring curiosities that the L<sup>A</sup>T<sub>E</sub>X kernel continues to use dangerously and inappropriately “optimized” definitions for such commands.

```

1520 \@namedef{endtabular*}{\endtabular}%

```

`\multicolumn`

```

1521 \long\def\multicolumn@LaTeX#1#2#3{%
1522 \multispan{#1}\begingroup
1523 \mkpream{#2}%
1524 \def\@sharp{#3}\set@typeset@protect
1525 \let\@startpbox\@startpbox\let\@endpbox\@endpbox
1526 \arstrut \preamble\hbox{ }\endgroup\ignorespaces
1527 }%
1528 \long\def\multicolumn@ltx#1#2#3{%
1529 \multispan{#1}%
1530 \begingroup
1531 \mkpream{#2}%
1532 \def\@sharp{#3}%
1533 \set@typeset@protect
1534 %\let\@startpbox\@startpbox\let\@endpbox\@endpbox
1535 \arstrut
1536 \preamble

```

```

1537 \hbox{ }%
1538 \endgroup
1539 \ignorespaces
1540 }%

```

`\array@align@` Here are the various procedures for the vertical alignment options. The change  
`\array@default` from standard L<sup>A</sup>T<sub>E</sub>X 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.

```

1541 \def\array@align@t{\leavevmode\vtop\bgroup}%
1542 \def\array@align@b{\leavevmode\vbox\bgroup}%
1543 \def\array@align@c{\leavevmode\@ifmmode{\vcenter\bgroup}{\vcenter\bgroup\aftergroup$\aftergroup$}%
1544 \def\array@align@v{%
1545 \@ifmmode{%
1546 \@badmath
1547 \vcenter\bgroup
1548 }{%
1549 \@ifinner{%
1550 $\vcenter\bgroup\aftergroup$
1551 }{%
1552 \@@par\bgroup
1553 }%
1554 }%
1555 }%
1556 \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.

```

1557 \def\array@row@rst{%
1558 \let\array@align@v\array@align@c
1559 }%
1560 \def\array@row@pre{%
1561 \def\array@row@pst{%

```

```

\toprule Default definitions for \toprule, \colrule, \botrule
\colrule 1562 \newcommand\toprule{\tab@rule{\column@font}{\column@fil}{\frstrut}}%
\botrule 1563 \newcommand\colrule{\unskip\lrstrut\\\tab@rule{\body@font}{\frstrut}}%
1564 \newcommand\botrule{\unskip\lrstrut\\\noalign{\hline@rule}{}}%

```

`\hline`

```

1565 \def\hline@LaTeX{%
1566 \noalign{\ifnum0=‘}\fi\hrule \@height \arrayrulewidth \futurelet

```

```

1567   \reserved@a\@xhline
1568 }%
1569 \def\hline<ltx{%
1570   \noalign{%
1571     \ifnum0='}\fi
1572     \hline@rule
1573     \futurelet\reserved@a\@xhline
1574     % \noalign ended in \@xhline
1575 }%
1576 \def\@xhline@unneeded{%
1577   \say\reserved@a
1578   \ifx\reserved@a\hline
1579     \vskip\doublerulesep
1580     \vskip-\arrayrulewidth
1581   \fi
1582   \ifnum0='{\fi}%
1583 }%
1584 \def\tab@rule#1#2#3{%
1585   \crr
1586   \noalign{%
1587     \hline@rule
1588     \gdef\@arstrut@hook{%
1589       \global\let\@arstrut@hook\@empty
1590       #3%
1591     }%
1592     \gdef\cell@font{#1}%
1593     \gdef\cell@fil{#2}%
1594   }%
1595 }%
1596 \def\column@font{}%
1597 \def\column@fil{}%
1598 \def\body@font{}%
1599 \def\cell@font{}%
1600 \def\frstrut{}%
1601 \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@cinc` is a kloutch, a magic number.

```

1602 \def\@arstrut@hline{%
1603   \relax
1604   \@ifmode{\copy}{\unhcopy}\@arstrutbox@hline
1605   \@arstrut@hook

```

```

1606 }%
1607 %
1608 \let\@arstrut@org\@arstrut
1609 \def\@arstrut@hook{%
1610 \global\let\@arstrut\@arstrut@org
1611 }%
1612 %
1613 \newbox\@arstrutbox@hline
1614 \appdef\set@arstrutbox{%
1615 \setbox\@arstrutbox@hline\hbox{%
1616 \setbox\z@\hbox{\$0^{0}_{-}{}}%
1617 \dimen@ht\z@\advance\dimen@\@arstrut@hline@clnc
1618 \@ifdim{\dimen@<\arraystretch\ht\strutbox}{\dimen@=\arraystretch\ht\strutbox}{}%
1619 \vrule \@height\dimen@
1620 \depth\arraystretch \dp\strutbox
1621 \@width\z@
1622 }%
1623 }%
1624 %
1625 \def\hline@rule{%
1626 \hrule \@height \arrayrulewidth
1627 \global\let\@arstrut\@arstrut@hline
1628 }%
1629 \def\@arstrut@hline@clnc{2\p@}% % Klotch: magic number

\tableleft@skip
1630 \def\tableleft@skip{\z@skip}%
1631 \def\tabmid@skip{\z@skip}\@flushglue
1632 \def\tabright@skip{\z@skip}%
1633 \def\tableleftsep{\tabcolsep}%
1634 \def\tabmidsep{\tabcolsep}%
1635 \def\tabrightsep{\tabcolsep}%
1636 \def\cell@fil{}%
1637 \def\pbox@hook{}%

\@arstrut
1638 \appdef\@arstrut{\@arstrut@hook}%
1639 \let\@arstrut@hook\@empty
1640 \def\@addtopreamble{\appdef\@preamble}%

\@mkpream
1641 \def\@mkpream@LaTeX#1{%
1642 \@firstamptrue\@lastchclass6
1643 \let\@preamble\@empty
1644 \let\protect\@unexpandable@protect
1645 \let\@sharp\relax
1646 \let\@startpbox\relax\let\@endpbox\relax
1647 \@expast{#1}%
1648 \expandafter\@tfor \expandafter

```

```

1649 \nextchar \expandafter:\expandafter=\reserved@a\do
1650 {\testpach\nextchar
1651 \ifcase \@chclass \@classz \or \@classi \or \@classii \or \@classiii
1652 \or \@classiv \or \@classv \fi\@lastchclass\@chclass}%
1653 \ifcase \@lastchclass \@acol
1654 \or \or \@preamerr \@ne\or \@preamerr \tw@\or \or \@acol \fi
1655 }%
1656 \def\@mkpream@ltx#1{%
1657 \@firstamptrue
1658 \@lastchclass6
1659 \let\@preamble\@empty
1660 \let\protect\@unexpandable@protect
1661 \let\@sharp\relax
1662 %\let\@startpbox\relax\let\@endpbox\relax
1663 \@expast{#1}%
1664 \expandafter\@tfor\expandafter\nextchar\expandafter:\expandafter=\reserved@a
1665 \do{%
1666 \expandafter\testpach\expandafter{\nextchar}%
1667 \ifcase\@chclass
1668 \@classz
1669 \or
1670 \@classi
1671 \or
1672 \@classii
1673 \or
1674 \@classiii
1675 \or
1676 \@classiv
1677 \or
1678 \@classv
1679 \fi
1680 \@lastchclass\@chclass
1681 }%
1682 \ifcase\@lastchclass
1683 \@acolr % right-hand column
1684 \or
1685 \or
1686 \@preamerr\@ne
1687 \or
1688 \@preamerr\tw@
1689 \or
1690 \or
1691 \@acolr % right-hand column
1692 \fi
1693 }%

```

\insert@column

```

1694 \def\insert@column@array{%
1695 \the@toks \the \@tempcnta
1696 \ignorespaces \@sharp \unskip

```



```

1697   \the@toks \the \count@ \relax
1698 }%
1699 \def\insert@column@array@new{%
1700   \the@toks\the\@tempcnta
1701   \array@row@rst\cell@font
1702   \ignorespaces\@sharp\unskip
1703   \the@toks\the\count@
1704   \relax
1705 }%

```

`\@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 L<sup>A</sup>T<sub>E</sub>X, 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`, T<sub>E</sub>X'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`.

```

1706 \def\@mkpream@relax{%
1707   \let\tableleftsep \relax
1708   \let\abmidsep \relax
1709   \let\abrightsep \relax
1710   \let\array@row@rst\relax
1711   \let\cell@font \relax
1712   \let\@startpbox \relax
1713 }%

```

`\@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.

```

1714 \def\@mkpream@array#1{%
1715   \gdef\@preamble{}\@lastchclass 4 \@firstampttrue
1716   \let\@sharp\relax \let\@startpbox\relax \let\@endpbox\relax
1717   \@temptokena{#1}\@tempwattrue
1718   \@whilesw\if@tempswa\fi{\@tempswafalse\the\NC@list}%
1719   \count@m@ne
1720   \let\the@toks\relax
1721   \prepnext@tok
1722   \expandafter \tfor \expandafter \@nextchar
1723   \expandafter :\expandafter =\the\@temptokena \do
1724   {\@testpach
1725   \ifcase \@chclass \@classz \or \@classi \or \@classii
1726     \or \save@decl \or \or \@classv \or \@classvi

```

```

1727     \or \@classvii \or \@classviii
1728     \or \@classx
1729     \or \@classx \fi
1730 \@lastchclass\@chclass}%
1731 \ifcase\@lastchclass
1732 \@acol \or
1733 \or
1734 \@acol \or
1735 \@preamerr \thr@@ \or
1736 \@preamerr \tw@ \@addtopreamble\@sharp \or
1737 \or
1738 \else \@preamerr \@ne \fi
1739 \def\the@toks{\the\toks}%
1740 }%
1741 \def\@mkpream@array@new#1{%
1742 \gdef\@preamble{}%
1743 \@lastchclass\@f@ur
1744 \@firstampttrue
1745 \let\@sharp\relax
1746 \@mkpream@relax
1747 %\let\@startpbox\relax\let\@endpbox\relax
1748 \@temptokena{#1}\@tempswatrue
1749 \@whiles\if@tempswa\fi{\@tempswafalse\the\NC@list}%
1750 \count@m@ne
1751 \let\the@toks\relax
1752 \prepnext@tok
1753 \expandafter\@tfor\expandafter\@nextchar\expandafter:\expandafter=\the\@temptokena
1754 \do{%
1755 \@testpach
1756 \ifcase\@chclass
1757 \@classz
1758 \or
1759 \@classi
1760 \or
1761 \@classii
1762 \or
1763 \save@decl
1764 \or
1765 \or
1766 \@classv
1767 \or
1768 \@classvi
1769 \or
1770 \@classvii
1771 \or
1772 \@classviii
1773 \or
1774 \@classx
1775 \or
1776 \@classx

```

```

1777 \fi
1778 \@lastchclass\@chclass
1779 }%
1780 \ifcase\@lastchclass
1781 \@acolr % right-hand column
1782 \or
1783 \or
1784 \@acolr % right-hand column
1785 \or
1786 \@preamerr\thr@@
1787 \or
1788 \@preamerr\tw@\@addtopreamble\@sharp
1789 \or
1790 \or
1791 \else
1792 \@preamerr\@ne
1793 \fi
1794 \def\the@toks{\the\toks}%
1795 }%

```

`\@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.

```

1796 \appdef\@mkpream@relax{%
1797 \let\CT@setup \relax
1798 \let\CT@color \relax
1799 \let\CT@do@color \relax
1800 \let\color \relax
1801 \let\CT@column@color\relax
1802 \let\CT@row@color \relax
1803 \let\CT@cell@color \relax
1804 }%

```

`\@addamp`

```

1805 \def\@addamp@LaTeX{%
1806 \if@firstamp\@firstampfalse\else\edef\@preamble{\@preamble &}\fi
1807 }%
1808 \def\@addamp@ltx{%
1809 \if@firstamp\@firstampfalse\else\@addtopreamble{&}\fi
1810 }%

```

`\@arrayacol`

```

1811 \def\@arrayacol@LaTeX{%
1812 \edef\@preamble{\@preamble \hskip \arraycolsep}%
1813 }%
1814 \def\@arrayacol@ltx{%
1815 \@addtopreamble{\hskip\arraycolsep}%
1816 }%

```

```

\@tabacol
1817 \def\@tabacoll{%
1818 \@addtopreamble{\hskip\tableftsep\relax}%
1819 }%
1820 \def\@tabacol@LaTeX{%
1821 \edef\@preamble{\@preamble \hskip \tabcolsep}%
1822 }%
1823 \def\@tabacol@ltx{%
1824 \@addtopreamble{\hskip\tabmidsep\relax}%
1825 }%
1826 \def\@tabacolr{%
1827 \@addtopreamble{\hskip\tabrightsep\relax}%
1828 }%

\@arrayclassz
1829 \def\@arrayclassz@LaTeX{%
1830 \ifcase \@lastchclass \@acolampacol \or \@ampacol \or
1831 \or \or \@addamp \or
1832 \@acolampacol \or \@firstampfalse \@acol \fi
1833 \edef\@preamble{\@preamble
1834 \ifcase \@chnum
1835 \hfil$\relax\@sharp$\hfil \or $\relax\@sharp$\hfil
1836 \or \hfil$\relax\@sharp$\fi}%
1837 }%
1838 \def\@arrayclassz@ltx{%
1839 \ifcase\@lastchclass
1840 \@acolampacol
1841 \or
1842 \@ampacol
1843 \or
1844 \or
1845 \or
1846 \@addamp
1847 \or
1848 \@acolampacol
1849 \or
1850 \@firstampfalse\@acoll
1851 \fi
1852 \ifcase\@chnum
1853 \@addtopreamble{%
1854 \hfil\array@row@rst$\relax\@sharp$\hfil
1855 }%
1856 \or
1857 \@addtopreamble{%
1858 \array@row@rst$\relax\@sharp$\hfil
1859 }%
1860 \or
1861 \@addtopreamble{%
1862 \hfil\array@row@rst$\relax\@sharp$%
1863 }%

```

```

1864 \fi
1865 }%

\@tabclassz
1866 \def\@tabclassz@LaTeX{%
1867 \ifcase\@lastchclass
1868 \acolampacol
1869 \or
1870 \ampacol
1871 \or
1872 \or
1873 \or
1874 \addamp
1875 \or
1876 \acolampacol
1877 \or
1878 \@firstampfalse\acol
1879 \fi
1880 \edef\@preamble{%
1881 \@preamble{%
1882 \ifcase\@chnum
1883 \hfil\ignorespaces\@sharp\unskip\hfil
1884 \or
1885 \hskip1sp\ignorespaces\@sharp\unskip\hfil
1886 \or
1887 \hfil\hskip1sp\ignorespaces\@sharp\unskip
1888 \fi}}%
1889 }%
1890 \def\@tabclassz@ltx{%
1891 \ifcase\@lastchclass
1892 \acolampacol
1893 \or
1894 \ampacol
1895 \or
1896 \or
1897 \or
1898 \addamp
1899 \or
1900 \acolampacol
1901 \or
1902 \@firstampfalse\acoll
1903 \fi
1904 \ifcase\@chnum
1905 \@addtopreamble{%
1906 {\hfil\array@row@rst\cell@font\ignorespaces\@sharp\unskip\hfil}%
1907 }%
1908 \or
1909 \@addtopreamble{%
1910 {\cell@fil\hskip1sp\array@row@rst\cell@font\ignorespaces\@sharp\unskip\hfil}%
1911 }%

```

```

1912 \or
1913 \@addtopreamble{%
1914   {\hfil\hskip1sp\array@row@rst\cell@font\ignorespaces\@sharp\unskip\cell@fil}%
1915 }%
1916 \fi
1917 }%

\@tabclassiv
1918 \def\@tabclassiv@LaTeX{%
1919 \@addtopreamble\@nextchar
1920 }%
1921 \def\@tabclassiv@ltx{%
1922 \expandafter\@addtopreamble\expandafter{\@nextchar}%
1923 }%

\@arrayclassiv
1924 \def\@arrayclassiv@LaTeX{%
1925 \@addtopreamble{\@nextchar}%
1926 }%
1927 \def\@arrayclassiv@ltx{%
1928 \expandafter\@addtopreamble\expandafter{\expandafter\@nextchar}%
1929 }%

\@classv
1930 \def\@classv@LaTeX{%
1931 \@addtopreamble{\@startpbox{\@nextchar}\ignorespaces
1932 \@sharp\@endpbox}%
1933 }%
1934 \def\@classv@ltx{%
1935 \expandafter\@addtopreamble
1936 \expandafter{%
1937 \expandafter \@startpbox
1938 \expandafter {\@nextchar}%
1939 \pbox@hook\array@row@rst\cell@font\ignorespaces\@sharp\@endpbox
1940 }%
1941 }%

\@classx
1942 \def\@classx@array{%
1943 \ifcase \@lastchclass
1944 \@acolampacol \or
1945 \@addamp \@acol \or
1946 \@acolampacol \or
1947 \or
1948 \@acol \@firstampfalse \or
1949 \@addamp
1950 \fi
1951 }%
1952 \def\@classx@array@new{%

```

```

1953 \ifcase \@lastchclass
1954 \@acolampacol
1955 \or
1956 \@addamp \@acol
1957 \or
1958 \@acolampacol
1959 \or
1960 \or
1961 \@firstampfalse\@acoll
1962 \or
1963 \@addamp
1964 \fi
1965 }%

```

## 6.15 Repair other broken parts of L<sup>A</sup>T<sub>E</sub>X

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

```

1966 \def\@xbitor@LaTeX #1{\@tempcntb \count#1
1967   \ifnum \@tempcnta =\z@
1968     \else
1969       \divide\@tempcntb\@tempcnta
1970       \ifodd\@tempcntb \@testtrue\fi
1971     \fi}%
1972 \def\@xbitor@ltx#1{%
1973   \@tempcntb\count#1\relax
1974   \@ifnum{\@tempcnta=\z@}{-}{%
1975     \divide\@tempcntb\@tempcnta
1976     \@ifodd\@tempcntb{\@testtrue}{-}}%
1977   }%
1978 }%
1979 \@ifx{\@xbitor\@xbitor@LaTeX}{%
1980   \class@info{Repairing broken LaTeX \string\@xbitor}%
1981 }{%
1982   \class@info{Unrecognized LaTeX \string\@xbitor. Please update this document class! (Proceedin
1983 }%
1984 \let\@xbitor\@xbitor@ltx

```

## 6.16 Syntax

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

```

1985 \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.

```
1986 \def\starttoc#1{%
1987   \begingroup
1988   \toc@pre
1989   \makeatletter
1990   \@input{\jobname.#1}%
1991   \if@filesw
1992     \expandafter\newwrite\csname tf@#1\endcsname
1993     \immediate\openout \csname tf@#1\endcsname \jobname.#1\relax
1994   \fi
1995   \@nobreakfalse
1996   \toc@post
1997 \endgroup
1998 }%
1999 \def\toc@pre{}%
2000 \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.

```
2001 \def\toc@@font{}%
2002 \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.

```
2003 \let\tocdim@section      \leftmargini
2004 \let\tocdim@subsection   \leftmarginii
2005 \let\tocdim@subsubsection \leftmarginiii
```



```

2006 \let\tocdim@paragraph \leftmarginiv
2007 \let\tocdim@appendix \leftmarginv
2008 \let\tocdim@pagenum \leftmarginvi

```

`\toc@pre@auto` We patch `\@starttoc` to: 1) before TOC processing, initialize the max registers and set the needed dimensions from the values stored in the auxiliary file, and 2) `\toc@post@auto` 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<sup>A</sup>T<sub>E</sub>X processing, the substyle does:

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

```

2009 \def\toc@pre@auto{%
2010 \toc@@font
2011 \@tempdima\z@
2012 \toc@setindent\@tempdima{section}%
2013 \toc@setindent\@tempdima{subsection}%
2014 \toc@setindent\@tempdima{subsubsection}%
2015 \toc@setindent\@tempdima{paragraph}%
2016 \toc@letdimen{appendix}%
2017 \toc@letdimen{pagenum}%
2018 }%
2019 \def\toc@post@auto{%
2020 \if@filesw
2021 \begingroup
2022 \toc@writedimen{section}%
2023 \toc@writedimen{subsection}%
2024 \toc@writedimen{subsubsection}%
2025 \toc@writedimen{paragraph}%
2026 \toc@writedimen{appendix}%
2027 \toc@writedimen{pagenum}%
2028 \endgroup
2029 \fi
2030 }%

```

`\toc@setindent`

```

2031 \def\toc@setindent#1#2{%
2032 \csname tocdim@#2\endcsname\tocdim@min\relax
2033 \@ifundefined{tocmax@#2}{\@namedef{tocmax@#2}{\z@}}{}%
2034 \advance#1\@nameuse{tocmax@#2}\relax
2035 \expandafter\edef\csname tocleft@#2\endcsname{\the#1}%
2036 }%

```

`\toc@letdimen`

```

2037 \def\toc@letdimen#1{%
2038 \csname tocdim@#1\endcsname\tocdim@min\relax
2039 \@ifundefined{tocmax@#1}{\@namedef{tocmax@#1}{\z@}}{}%
2040 \expandafter\let\csname tocleft@#1\endcsname\expandafter\endcsname\csname tocmax@#1\endcsname
2041 }%

```

`\toc@writedimen`

```
2042 \def\toc@writedimen#1{%
2043 \immediate\write\@auxout{%
2044 \gdef\expandafter\string\csname tocmax@#1\endcsname{%
2045 \expandafter\the\csname tocdim@#1\endcsname
2046 }%
2047 }%
2048 }%
```

`\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

```
2049 \def\l@sections#1#2#3#4{%
2050 \begingroup
2051 \everypar{}%
2052 \set@tocdim@pagenum\@tempboxa{#4}%
2053 \global\@tempdima\csname tocdim@#2\endcsname
2054 \leftskip\csname tocleft@#2\endcsname\relax
2055 \dimen@\csname tocleft@#1\endcsname\relax
2056 \parindent-\leftskip\advance\parindent\dimen@
2057 \rightskip\tocleft@pagenum plus 1fil\relax
2058 \skip@\parfillskip\parfillskip\z@
2059 \let\numberline\numberline@@sections
2060 \@nameuse{1@f@#2}%
2061 \ignorespaces#3\unskip\nobreak\hskip\skip@
2062 \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`).

```
2063 \expandafter\par
2064 \expandafter\aftergroup\csname tocdim@#2%
2065 \expandafter\endcsname
2066 \expandafter\endgroup
2067 \the\@tempdima\relax
2068 }%
```

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

```
2069 \def\set@tocdim@pagenum#1#2{%
2070 \setbox#1\hbox{\ignorespaces#2}%
2071 \@ifdim{\tocdim@pagenum<\wd#1}{\global\tocdim@pagenum\wd#1}{}%
2072 }%
```

`\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!

```
2073 \def\numberline@@sections#1{%
2074 \leavevmode\hb@xt@-\parindent{%
2075 \hfil
2076 \@if@empty{#1}{-}{%
2077 \setbox\z@\hbox{#1.\kern\ltxu@dotsep}%
2078 \@ifdim{\@tempdima<\wd\z@}{\global\@tempdima\wd\z@}{}%
2079 \unhbox\z@
2080 }%
2081 }%
2082 \ignorespaces
2083 }%
2084 \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  $\text{\LaTeX}$  kernel procedure `\list`, which is the implementation of all lists.

```
2085 \def\list#1#2{%
2086 \ifnum \@listdepth >5\relax
2087 \toodeep
2088 \else
2089 \global\advance\@listdepth@ne
2090 \fi
2091 \rightmargin\z@
2092 \listparindent\z@
2093 \itemindent\z@
2094 \csname @list\romannumeral\the\@listdepth\endcsname
2095 \def\@itemlabel{#1}%
2096 \let\makelabel\@mklab
2097 \@nbrlistfalse
2098 #2\relax
```

```

2099 \@trivlist
2100 \parskip\parsep
2101 \set@listindent
2102 \ignorespaces
2103 }%
2104 \def\set@listindent@parshape{%
2105 \parindent\listparindent
2106 \advance\@totalleftmargin\leftmargin
2107 \advance\linewidth-\rightmargin
2108 \advance\linewidth-\leftmargin
2109 \parshape\@ne\@totalleftmargin\linewidth
2110 }%
2111 \def\set@listindent@{%
2112 \parindent\listparindent
2113 \advance\@totalleftmargin\leftmargin
2114 \advance\rightskip\rightmargin
2115 \advance\leftskip\@totalleftmargin
2116 }%
2117 \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 2118 \providecommand\href[0]{\begingroup\@sanitize@url\@href}%
\doi 2119 \def\@href#1{\@startlink{#1}\endgroup\@@href}%
\doibase 2120 \def\@@href#1{#1\@endlink}%
2121 \providecommand \url [0]{\begingroup\@sanitize@url \@url }%
2122 \def \@url #1{\endgroup\@href {#1}{\URL@prefix#1}}%
2123 \providecommand \URL@prefix [0]{URL }%
2124 \providecommand\doi[0]{\begingroup\@sanitize@url\@doi}%
2125 \def\@doi#1{\endgroup\@startlink{\doibase#1}doi:\discretionary {}{}#1\@endlink }%
2126 \providecommand \doibase [0]{http://dx.doi.org/}%
2127 \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 running under PDFLATEX. If so, and if PDF output is requested, then we use its primitives to implement hypertext, breaking out the link attributes in `\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.

```

2128 \def\@startlink#1{%
2129 \def\@endlink{}%
2130 \@ifundefined \pdfoutput {\true@sw}{\ifnum\z@=\pdfoutput{\true@sw}{\false@sw}}%
2131 {%
2132 \def\@startlink@hypertext#1{\leavevmode\special{html:<a href="#1">}}%

```

```

2133 \def\@@endlink@hypertext{\special{html:</a>}}%
2134 }{%
2135 \def\@@startlink@hypertext#1{%
2136   \leavevmode
2137   \pdfstartlink\pdfstartlink@attr
2138     user{/Subtype/Link/A<</Type/Action/S/URI/URI(#1)>>}}%
2139   \relax
2140 }%
2141 \def\@@endlink@hypertext{\pdfendlink}%
2142 \def\pdfstartlink@attr{attr{/Border[0 0 1 ]/H/I/C[0 1 1]}}%
2143 }%
2144 \def\hypertext@enable@ltx{%
2145   \let\@@startlink\@@startlink@hypertext
2146   \let\@@endlink\@@endlink@hypertext
2147 }%

```

`\href` The `\href` command of `hyperref` was extended 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.

```

2148 \def\href@Hy{\hyper@normalise \href@ }%
2149 \def\href@Hy@ltx{\@ifnextchar\bgroup\Hy@href{\hyper@normalise\href@}}%
2150 \def\Hy@href#\{\hyper@normalise\href@}%
2151 \begingroup
2152   \endlinechar=-1 %
2153   \catcode'\^^A=14 %
2154   \catcode'\^^M\active
2155   \catcode'\%\active
2156   \catcode'\#\active
2157   \catcode'\_\active
2158   \catcode'\$\active
2159   \catcode'\&\active
2160   \gdef\hyper@normalise@ltx{^^A
2161     \begingroup
2162     \catcode'\^^M\active
2163     \def^^M{ }^^A
2164     \catcode'\%\active
2165     \let%\@percentchar
2166     \let%\@percentchar
2167     \catcode'\#\active
2168     \def#\{\hyper@hash}^^A
2169     \def\#\{\hyper@hash}^^A
2170     \@makeother\^^A
2171     \edef&\string&}^^A
2172     \edef&\{\string&}^^A
2173     \edef\textunderscore{\string_}^^A
2174     \let\_ \textunderscore
2175     \catcode'\_\active

```

```

2176 \let_\textunderscore
2177 \let^hyper@tilde
2178 \let^\hyper@tilde
2179 \let\textasciitilde\hyper@tilde
2180 \let\\@backslashchar
2181 \edef$\string$^^A
2182 \Hy@safe@activestruel
2183 \hyper@n@rmalise
2184 }^^A
2185 \catcode'\#=6 ^^A
2186 \gdef\Hy@ActiveCarriageReturn@ltx{^^M}^^A
2187 \gdef\hyper@n@rmalise@ltx#1#2{^^A
2188 \def\Hy@tempa{#2}^^A
2189 \ifx\Hy@tempa\Hy@ActiveCarriageReturn
2190 \Hy@ReturnAfterElseFi{^^A
2191 \hyper@@normalise{#1}^^A
2192 }^^A
2193 \else
2194 \Hy@ReturnAfterFi{^^A
2195 \hyper@@normalise{#1}{#2}^^A
2196 }^^A
2197 \fi
2198 }^^A
2199 \gdef\hyper@@normalise@ltx#1#2{^^A
2200 \edef\Hy@tempa{^^A
2201 \endgroup
2202 \noexpand#1{\Hy@RemovePercentCr#2^^M\@nil}^^A
2203 }^^A
2204 \Hy@tempa
2205 }^^A
2206 \gdef\Hy@RemovePercentCr@ltx#1^^M#2\@nil{^^A
2207 #1^^A
2208 \ifx\limits#2\limits
2209 \else
2210 \Hy@ReturnAfterFi{^^A
2211 \Hy@RemovePercentCr #2\@nil
2212 }^^A
2213 \fi
2214 }^^A
2215 \endgroup
2216 \def\switch@hyperref@href{%
2217 \expandafter\@ifx\expandafter{\csname href \endcsname\href@Hy}{
2218 \class@info{Repairing hyperref 6.75r \string\href}%
2219 \let\hyper@normalise\hyper@normalise@ltx
2220 \let\hyper@@normalise\hyper@@normalise@ltx
2221 \let\hyper@n@rmalise\hyper@n@rmalise@ltx
2222 \let\Hy@ActiveCarriageReturn\Hy@ActiveCarriageReturn@ltx
2223 \let\Hy@RemovePercentCr\Hy@RemovePercentCr@ltx
2224 \let\href\href@Hy@ltx
2225 }-}%

```

```

2226 }%
2227 \appdef\document@inithook{\switch@hyperref@href}%

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

2228 \def\typeout@org#1{%
2229 \begingroup
2230 \set@display@protect
2231 \immediate\write\@unused{#1}%
2232 \endgroup
2233 }%
2234 \long\def\typeout@ltx#1{%
2235 \begingroup
2236 \set@display@protect
2237 \immediate\write\@unused{#1}%
2238 \endgroup
2239 }%
2240 \@ifx{\typeout\typeout@org}{%
2241 \let\typeout\typeout@ltx
2242 \true@sw
2243 }{%
2244 \rvtx@ifformat@geq{2020-10-01}%
2245 {\true@sw}{\false@sw}%
2246 }%
2247 {\class@info{Making \string\typeout\space \string\long}}%
2248 {}%

```

## 6.20 End of the kernel DOCSTRIP module

Here ends the module.

```
2249 %</kernel>
```





<code>\@arrayacol@LaTeX</code>	.. 1065, 1811	<code>\@bsphack</code>	..... 678, 893
<code>\@arrayacol@ltx</code>	1155, 1237, 1814	<code>\@caption</code>	..... 30
<code>\@arrayclassiv</code>	1069, 1159, 1354, 1365, <a href="#">1924</a>	<code>\@capttype</code>	..... 34
<code>\@arrayclassiv@LaTeX</code>	1069, 1924	<code>\@capttype</code>	..... 749
<code>\@arrayclassiv@ltx</code>	. 1159, 1927	<code>\@centercr</code>	..... 20, 21
<code>\@arrayclassz</code>	. 1067, 1157, 1353, 1364, <a href="#">1829</a>	<code>\@centercr</code>	..... <a href="#">497</a>
<code>\@arrayclassz@LaTeX</code>	1067, 1829	<code>\@centering</code>	. 504, 507, 510, 529, 535, 548
<code>\@arrayclassz@ltx</code>	.. 1157, 1838	<code>\@chclass</code>	1651, 1652, 1667, 1680, 1725, 1730, 1756, 1778
<code>\@arrayacr</code>	..... 47	<code>\@checkend</code>	... 369, 411, 423, 427
<code>\@arrayacr</code>	1180, 1241, <a href="#">1300</a> , 1355, 1366, 1458, 1484	<code>\@chnum</code>	.. 1834, 1852, 1882, 1904
<code>\@arrayacr@array</code>	.... 1180, 1312	<code>\@classi</code>	. 1651, 1670, 1725, 1759
<code>\@arrayacr@new</code>	..... 1241, 1317	<code>\@classii</code>	1651, 1672, 1725, 1761
<code>\@arrayleft</code>	..... 1451, 1481	<code>\@classiii</code>	..... 1651, 1674
<code>\@arrayright</code>	..... 1501, 1505	<code>\@classiv</code>	1259, 1268, 1354, 1365, 1652, 1676
<code>\@arstrut</code>	..... 54	<code>\@classv</code>	. 1071, 1161, 1652, 1678, 1726, 1766, <a href="#">1930</a>
<code>\@arstrut</code>	1395, 1411, 1448, 1474, 1526, 1535, 1608, 1610, 1627, <a href="#">1638</a>	<code>\@classv@LaTeX</code>	..... 1071, 1930
<code>\@arstrut@hline</code>	..... 54	<code>\@classv@ltx</code>	..... 1161, 1934
<code>\@arstrut@hline</code>	..... <a href="#">1602</a>	<code>\@classvi</code>	..... 1726, 1768
<code>\@arstrut@hline@clnc</code>	..... 54	<code>\@classvii</code>	..... 1727, 1770
<code>\@arstrut@hline@clnc</code>	1617, 1629	<code>\@classviii</code>	..... 1727, 1772
<code>\@arstrut@hook</code>	..... 54	<code>\@classx</code>	. 1178, 1236, 1728, 1729, 1774, 1776, <a href="#">1942</a>
<code>\@arstrut@hook</code>	1588, 1589, <a href="#">1602</a> , 1638, 1639	<code>\@classx@array</code>	..... 1178, 1942
<code>\@arstrut@org</code>	..... <a href="#">1602</a>	<code>\@classx@array@new</code>	. 1236, 1952
<code>\@arstrut@box</code>	..... 49, 54	<code>\@classz</code>	. 1258, 1267, 1353, 1364, 1651, 1668, 1725, 1757
<code>\@arstrut@box</code>	.....	<code>\@clsextension</code>	..... 19, 20
	.. 1308, 1309, 1332, 1338, 1388, 1432, 1441, 1463	<code>\@clsextension</code>	... 460, 477, 485
<code>\@arstrut@box@hline</code>	..... 54	<code>\@ctrerr</code>	..... 924
<code>\@arstrut@box@hline</code>	..... <a href="#">1602</a>	<code>\@currbox</code>	..... 34, 35
<code>\@auxout</code>	..... 670, 679, 2043	<code>\@currbox</code>	840, 843, 845, 846, 852, 853, 862
<code>\@backslashchar</code>	..... 2180	<code>\@currentHref</code>	..... 615, 1037
<code>\@boole@def</code>	..... 13	<code>\@currentlabel</code>	500, 516, 642, 680
<code>\@boole@def</code>	..... <a href="#">307</a> , 319–322, 324–334	<code>\@currentlabelname</code>	..... 39
<code>\@boolean</code>	..... 13	<code>\@currentxt</code>	..... 19
<code>\@boolean</code>	..... <a href="#">307</a>	<code>\@currentxt</code>	442, 459, 460, 476, 484
<code>\@booleanfalse</code>	..... 13	<code>\@currname</code>	..... 19
<code>\@booleanfalse</code>	... <a href="#">317</a> , 803, 848	<code>\@currname</code>	... 442, 458, 475, 483
<code>\@booleantrue</code>	..... 13	<code>\@currnamestack@ltx</code>	. 470, 474, 479, 487, 489
<code>\@booleantrue</code>	.... <a href="#">317</a> , 733, 846	<code>\@dblarg</code>	..... 940, 941

<code>\@dblfloat</code>	784, 786, 808	<code>\@hangfrom@section</code>	38
<code>\@dblfloat@LaTeX</code>	784, 808	<code>\@height</code>	1308, 1309, 1389, 1433, 1442, 1464, 1566, 1619, 1626
<code>\@defaultsubs</code>	392	<code>\@href</code>	2118, 2119, 2122
<code>\@depth</code>	1308, 1309, 1333, 1339, 1390, 1434, 1443, 1465, 1620	<code>\@ialph</code>	919, 920
<code>\@dofilelist</code>	385	<code>\@if@empty</code>	319, 2076
<code>\@doi</code>	2124, 2125	<code>\@if@sw</code>	322, 323, 377, 394, 396
<code>\@eha</code>	202	<code>\@ifcat</code>	319
<code>\@enddocumenthook</code>	368	<code>\@ifdim</code>	319, 386, 865, 964, 1004, 1618, 2071, 2078
<code>\@endpbox</code>	57	<code>\@ifempty</code>	13
<code>\@endpbox</code>	1396, 1419, 1525, 1534, 1646, 1662, 1716, 1747, 1932, 1939	<code>\@ifeof</code>	319
<code>\@eqcnt</code>	503, 508, 509, 511, 517, 530, 533, 536	<code>\@ifhbox</code>	319, 861, 880
<code>\@eqncr</code>	47	<code>\@ifhmode</code>	319
<code>\@eqncr</code>	505, 519	<code>\@ifinner</code>	319, 712, 1549
<code>\@eqnset</code>	507, 529	<code>\@ifl@t@r</code>	303
<code>\@eqnswtrue</code>	501, 517	<code>\@ifl@ter</code>	462
<code>\@eqpen</code>	47	<code>\@ifmmode</code>	319, 1360, 1376, 1543, 1545, 1604
<code>\@esphack</code>	682	<code>\@ifnextchar</code>	557, 566, 581, 1289, 1292, 1295, 1298, 1306, 1307, 1323, 1327, 2149
<code>\@expast</code>	1647, 1663	<code>\@ifnotempty</code>	13
<code>\@extension</code>	18, 19	<code>\@ifnotrelax</code>	13
<code>\@finalstrut</code>	648	<code>\@ifnotrelax</code>	297, 685, 860, 879
<code>\@firstampfalse</code>	1806, 1809, 1832, 1850, 1878, 1902, 1948, 1961	<code>\@ifnum</code>	319, 845, 946, 975, 989, 1974, 2130
<code>\@firstamptrue</code>	1642, 1657, 1715, 1744	<code>\@ifodd</code>	319, 1976
<code>\@firstofone</code>	367	<code>\@ifpackageloaded</code>	221, 606, 1045, 1169
<code>\@float</code>	29, 33, 35	<code>\@ifstar</code>	939, 1304, 1305, 1315, 1320
<code>\@float@LaTeX</code>	782, 783, 807	<code>\@ifundefined</code>	33, 955, 968, 973, 984, 987, 1008, 1013, 1018, 1021, 2033, 2039
<code>\@floatboxreset</code>	755	<code>\@ifvbox</code>	319
<code>\@flushglue</code>	495, 1631	<code>\@ifvmode</code>	319, 551
<code>\@font@warning</code>	387	<code>\@ifvoid</code>	319, 709, 720
<code>\@footnotemark</code>	26	<code>\@ifx</code>	13
<code>\@footnotetext</code>	23, 26	<code>\@ifx</code>	297, 298, 301, 319, 340, 395, 540, 619, 907, 1056–1076, 1171–1183, 1252, 1253, 1979, 2217, 2240
<code>\@footnotetext</code>	587, 607, 608, 623	<code>\@ifx@empty</code>	319
<code>\@gobble@opt@one</code>	63	<code>\@ifxundefined</code>	13
<code>\@gobble@opt@one</code>	1985		
<code>\@halignto</code>	1356, 1367, 1372, 1382, 1394, 1428, 1447, 1492		
<code>\@hang@from</code>	968, 999		
<code>\@hang@froms</code>	1008, 1027		
<code>\@hangfrom</code>	37–39		
<code>\@hangfrom</code>	999		

<code>\@ifxundefined</code> . . . 269, 274, 279, <a href="#">297</a> , 782, 2130	<code>\@nameuse</code> . 218, 737, 1406, 1482, 2034, 2060
<code>\@ifxundefined@cs</code> 301, 790, 801, 821	<code>\@ne</code> . . . . . <a href="#">11</a> , <a href="#">67</a>
<code>\@iiiparbox</code> . . . . . 731	<code>\@newl@bel</code> . . . . . 380
<code>\@itemlabel</code> . . . . . 2095	<code>\@nextchar</code> 1649, 1650, 1664, 1666, 1722, 1753, 1919, 1922, 1925, 1928, 1931, 1938
<code>\@lastchclass</code> . . . . . . . 1642, 1652, 1653, 1658, 1680, 1682, 1715, 1730, 1731, 1743, 1778, 1780, 1830, 1839, 1867, 1891, 1943, 1953	<code>\@nmbulistfalse</code> . . . . . 2097
<code>\@latex@warning@no@line</code> . . 397	<code>\@nobreakfalse</code> . . . . . 1995
<code>\@listdepth</code> . . . 2086, 2089, 2094	<code>\@normalcr</code> . . . . . <a href="#">20</a>
<code>\@mainaux</code> . . . . . <a href="#">15</a>	<code>\@normalcr</code> . . . . . 491
<code>\@mainaux</code> . . . . . 378	<code>\@nul</code> . . . . . 905, 916
<code>\@makecol</code> . . . . . <a href="#">9</a>	<code>\@onefilewithoptions</code> . . . . . <a href="#">18</a>
<code>\@makefnmark</code> . . . . . <a href="#">23</a> , <a href="#">27</a>	<code>\@outputdblcol</code> . . . . . <a href="#">9</a>
<code>\@makefnmark</code> . . . . 575, 623, 769	<code>\@p@pfilename</code> . . . . . <a href="#">473</a>
<code>\@makefntext</code> . . . . . <a href="#">27</a>	<code>\@p@pfilename@ltx</code> . . . . 470, 482
<code>\@makefntext</code> . . . . . 646	<code>\@parboxrestore</code> . . . . . 641, 754
<code>\@makeother</code> . . . . 900, 904, 2170	<code>\@pass@ptions</code> . . . . . 448
<code>\@minipagefalse</code> . . . . . 728	<code>\@percentchar</code> . . . . . 2165, 2166
<code>\@mklab</code> . . . . . 2096	<code>\@preamble</code> 1393, 1395, 1403, 1409, 1413, 1428, 1447, 1448, 1458, 1470, 1472, 1476, 1492, 1501, 1506, 1526, 1536, 1640, 1643, 1659, 1715, 1742, 1806, 1812, 1821, 1833, 1880, 1881
<code>\@mkpream</code> . . . . . <a href="#">57</a> , <a href="#">59</a>	<code>\@preamerr</code> 1402, 1426, 1654, 1686, 1688, 1735, 1736, 1738, 1786, 1788, 1792
<code>\@mkpream</code> 1063, 1153, 1177, 1235, 1392, 1408, 1446, 1469, 1523, 1531, <a href="#">1641</a> , <a href="#">1714</a>	<code>\@provide</code> . <a href="#">339</a> , 1047, 1049, 1052
<code>\@mkpream@LaTeX</code> . . . . 1063, 1641	<code>\@pushfilename</code> . . . . . <a href="#">473</a>
<code>\@mkpream@array</code> . . . . 1177, 1714	<code>\@pushfilename@ltx</code> . . . 453, 473
<code>\@mkpream@array@new</code> 1235, 1741	<code>\@refundefined</code> . . . . . 393
<code>\@mkpream@ltx</code> . . . . . 1153, 1656	<code>\@reset@ptions</code> . . . . . <a href="#">19</a>
<code>\@mkpream@relax</code> . . . . . <a href="#">57</a>	<code>\@reset@ptions</code> . . . . . 456, 471
<code>\@mkpream@relax</code> <a href="#">1706</a> , 1746, <a href="#">1796</a>	<code>\@resetactivechars</code> . . . . . 261
<code>\@mpargs</code> . . . . . 731	<code>\@rightskip</code> . . . . . 494
<code>\@mpfn</code> . . . . . <a href="#">23</a> , <a href="#">28</a>	<code>\@runin@to</code> . . . . . <a href="#">38</a>
<code>\@mpfn</code> . . . 560, 564, 573, 586, 592, 596, 599, 602, 643, 742	<code>\@runin@to</code> . . . . . 984, 1000
<code>\@mpfootins</code> . . . . . <a href="#">31</a>	<code>\@runin@tos</code> . . . . . 1018, 1028
<code>\@mpfootins</code> . 698, 701, 702, 709, 710, 713, 716, 720, 721	<code>\@sanitize</code> . . . . . 2127
<code>\@mpfootnotetext</code> . . . . . <a href="#">26</a>	<code>\@sanitize@url</code> 2118, 2121, 2124, 2127
<code>\@mpfootnotetext</code> . 609, 610, <a href="#">623</a>	<code>\@secCNTformat</code> . . . . . <a href="#">37</a> , <a href="#">38</a>
<code>\@mpmakefntext</code> . . . . . <a href="#">734</a>	<code>\@secCNTformat</code> . . . . . 956
<code>\@multiplelabels</code> . . . . 395, 403	<code>\@secpenalty</code> . . . . . 937
<code>\@namedef</code> . 814, 1520, 2033, 2039	<code>\@sect</code> . . . . . <a href="#">37</a>

<code>\@sect</code> .....	944	<code>\@tabclassz</code> ..	1070, 1160, 1258, 1267, <u>1866</u>
<code>\@sect@ltx</code> .....	941, 945	<code>\@tabclassz@LaTeX</code> ..	1070, 1866
<code>\@setckpt</code> .....	379	<code>\@tabclassz@ltx</code> ....	1160, 1890
<code>\@sharp</code> ..	1399, 1422, 1454, 1487, 1524, 1532, 1645, 1661, 1696, 1702, 1716, 1736, 1745, 1788, 1835, 1836, 1854, 1858, 1862, 1883, 1885, 1887, 1906, 1910, 1914, 1932, 1939	<code>\@tabular</code> .....	<u>45</u>
<code>\@ssect</code> .....	<u>37</u> , <u>39</u>	<code>\@tabular</code>	1051, 1058, 1148, 1172, 1230, <u>1254</u>
<code>\@ssect@ltx</code> .....	<u>1001</u>	<code>\@tabular@LaTeX</code> ....	1058, 1254
<code>\@ssect@ltx</code> .....	940, 1001	<code>\@tabular@array</code> ....	1172, 1272
<code>\@startcolumn</code> .....	<u>9</u>	<code>\@tabular@array@new</code>	1230, 1280
<code>\@startpbox</code> .....	<u>57</u>	<code>\@tabular@ltx</code> .....	1148, 1263
<code>\@startpbox</code> .....	.. 1396, 1418, 1525, 1534, 1646, 1662, 1712, 1716, 1747, 1931, 1937	<code>\@tabularacr</code> .....	<u>47</u>
<code>\@startsection</code> .....	<u>37</u> , <u>39</u>	<code>\@tabularacr</code> ..	1073, 1163, 1252, 1260, 1269, <u>1300</u>
<code>\@startsection</code> .....	<u>925</u>	<code>\@tabularacr@LaTeX</code> .....	<u>47</u>
<code>\@startsection@hook</code> ..	926, 943	<code>\@tabularacr@LaTeX</code> ..	1073, 1304
<code>\@starttoc</code> .....	<u>65</u>	<code>\@tabularacr@ltx</code> ....	1163, 1305
<code>\@starttoc</code> .....	<u>1986</u>	<code>\@tbpopen</code> .....	<u>47</u>
<code>\@svsec</code> .....	<u>38</u>	<code>\@tbpopen</code> .....	<u>1300</u>
<code>\@svsec</code> ..	948, 954, 956, 959, 969, 985	<code>\@temp@sw</code> .....	846, 848, 852
<code>\@svsechd</code> .....	982, 1016	<code>\@tempa</code> .....	222, 224, 858, 860, 861, 866, 867, 869, 870, 877, 879–882, 906, 907
<code>\@tabacol</code>	1066, 1156, 1238, 1257, 1266, 1283, <u>1817</u>	<code>\@tempb</code> .....	897, 907
<code>\@tabacol@LaTeX</code> ....	1066, 1820	<code>\@tempboxa</code> .....	<u>66</u>
<code>\@tabacol@ltx</code> ..	1156, 1238, 1823	<code>\@tempc</code> .....	895, 913
<code>\@tabacoll</code> ....	1264, 1281, 1817	<code>\@tempcnta</code> .....	<u>35</u>
<code>\@tabacolr</code> ....	1265, 1282, 1826	<code>\@tempcntb</code>	882, 1966, 1969, 1970, 1973, 1975, 1976
<code>\@tabarray</code> .....	<u>46</u>	<code>\@tempdima</code> .....	<u>66</u> , <u>67</u>
<code>\@tabarray</code>	1059, 1149, 1173, 1231, 1261, 1270, 1278, 1286, <u>1288</u> , 1357, 1368, 1373, 1383	<code>\@tempskipa</code> ..	929, 931, 932, 937, 963, 964, 1003, 1004
<code>\@tabarray@LaTeX</code> ...	1059, 1288	<code>\@temptokena</code> ..	1717, 1723, 1748, 1753
<code>\@tabarray@array</code> ...	1173, 1294	<code>\@testdef</code> .....	380
<code>\@tabarray@array@new</code>	1231, 1297	<code>\@testpach</code>	1650, 1666, 1724, 1755
<code>\@tabarray@ltx</code> .....	1149, 1291	<code>\@testtrue</code> .....	1970, 1976
<code>\@tabclassiv</code> ..	1068, 1158, 1259, 1268, <u>1918</u>	<code>\@textsuperscript</code> .....	625
<code>\@tabclassiv@LaTeX</code> ..	1068, 1918	<code>\@tfor</code> ...	1648, 1664, 1722, 1753
<code>\@tabclassiv@ltx</code> ...	1158, 1921	<code>\@thefnmark</code> ..	593, 600, 626, 643
		<code>\@toodeep</code> .....	2087
		<code>\@totalleftmargin</code> ..	2106, 2109, 2113, 2115
		<code>\@tpfootnotetext</code> .....	<u>623</u>
		<code>\@trivlist</code> .....	2099
		<code>\@twopowerfourteen</code> .....	<u>34</u>

<code>\@twopowerfourteen</code>	...	836, 864
<code>\@twopowertwo</code>	.....	<a href="#">34</a>
<code>\@twopowertwo</code>	...	837, 881, 882
<code>\@undefined</code>	.....	<a href="#">45</a>
<code>\@undefined</code>	.....	469
<code>\@unexpandable@protect</code>	..	1644, 1660
<code>\@unused</code>	.....	2231, 2237
<code>\@unusedoptionlist</code>	...	448, 449
<code>\@url</code>	.....	2121, 2122
<code>\@whilesw</code>	.....	1718, 1749
<code>\@width</code>	..	1308, 1309, 1333, 1339, 1391, 1435, 1444, 1466, 1621
<code>\@write@floatline</code>	.....	<a href="#">36</a>
<code>\@write@floatline</code>	.....	<a href="#">903</a>
<code>\@writefile</code>	.....	670
<code>\@x@sf</code>	.....	571, 578
<code>\@xargarraycr</code>	.....	<a href="#">47</a>
<code>\@xargarraycr</code>	..	1075, 1165, 1182, 1243, <a href="#">1300</a>
<code>\@xargarraycr@LaTeX</code>		1075, 1308
<code>\@xargarraycr@array</code>		1183, 1343
<code>\@xargarraycr@ltx</code>	..	1166, 1309
<code>\@xargarraycr@new</code>	..	1243, 1336
<code>\@xarraycr</code>	.....	<a href="#">47</a>
<code>\@xarraycr</code>	....	1181, 1242, <a href="#">1300</a>
<code>\@xarraycr@array</code>	...	1181, 1322
<code>\@xarraycr@new</code>	.....	1242, 1326
<code>\@xbitor</code>	.....	<a href="#">1966</a>
<code>\@xbitor@LaTeX</code>	.....	1966, 1979
<code>\@xbitor@ltx</code>	.....	1972, 1984
<code>\@xfloat</code>	.....	<a href="#">734</a>
<code>\@xfloat@LaTeX</code>	.....	734, 738
<code>\@xfloat@anchored</code>	....	738, 748
<code>\@xfloat@prep</code>	.....	736, 740
<code>\@xfootnotemark</code>	.....	<a href="#">23</a>
<code>\@xfootnotenext</code>	.....	<a href="#">26</a>
<code>\@xhline</code>	.....	1567, 1573, 1574
<code>\@xhline@unneeded</code>	.....	1576
<code>\@xifempty</code>	.....	<a href="#">13</a>
<code>\@xssect</code>	.....	997, 1025
<code>\@xtabularcr</code>	.....	<a href="#">47</a>
<code>\@xtabularcr</code>	..	1074, 1164, <a href="#">1300</a>
<code>\@xtabularcr@LaTeX</code>	..	1074, 1306
<code>\@xtabularcr@ltx</code>	...	1164, 1307
<code>\@xvi</code>	.....	835, 845, 846
<code>\@xxxii</code>	.....	841, 842, 844
<code>\@yargarraycr</code>	.....	<a href="#">47</a>
<code>\@yargarraycr</code>	..	1076, 1166, 1183, 1244, <a href="#">1300</a>
<code>\@yargarraycr@LaTeX</code>		1076, 1310
<code>\@yargarraycr@array</code>		1183, 1343
<code>\@yargarraycr@ltx</code>	..	1166, 1311
<code>\@yargarraycr@new</code>	..	1244, 1347
<code>\[</code>	.....	<a href="#">22</a> , <a href="#">23</a>
<code>\[</code>	.....	904
<code>\</code>	.....	<a href="#">20</a> , <a href="#">21</a> , <a href="#">47</a>
<code>\{</code>	.....	134, 900, 904
<code>\}</code>	.....	134, 900, 904
<code>\]</code>	.....	<a href="#">22</a>
<code>\]</code>	.....	904
<code>\^</code>	....	899, 912, 2153, 2154, 2162
<code>\_</code>	.....	53, 2157, 2174, 2175
<code>\~</code>	.....	2178
<code>00readme</code>	.....	<a href="#">4</a>
<code>10pt document class option</code>	...	<a href="#">89</a>
<code>11pt document class option</code>	...	<a href="#">89</a>
<code>12pt document class option</code>	...	<a href="#">89</a>
<code>□</code>	.....	<a href="#">90</a> , <a href="#">92</a>
<code>\□</code>	..	42, 48, 68, 71, 72, 76, 93, 95, 101, 106, 113, 222, 2127

## A

<code>\abovedisplayshortskip</code>	...	525
<code>\abovedisplayskip</code>	....	520–525
<code>\active</code>	...	899, 2154–2159, 2162, 2164, 2167, 2175
<code>\addcontentsline</code>	.....	<a href="#">39</a>
<code>\addcontentsline</code>	..	<a href="#">672</a> , 974, 988, 1014, 1022
<code>\addtocontents</code>	.....	<a href="#">660</a> , 673
<code>\AddToHook</code>	.....	345, 409
<code>\aftergroup</code>	.....	<a href="#">53</a>
<code>\aftergroup</code>	...	1543, 1550, 2064
<code>\AfterLastShipout</code>	.....	<a href="#">16</a> , <a href="#">18</a>
<code>\appdef</code>	.....	<a href="#">12</a> , <a href="#">15</a> , <a href="#">16</a> , <a href="#">18</a> , <a href="#">29</a>
<code>\appdef</code>	.....	261, <a href="#">268</a> , 354, 437, 604, 690, 694, 1040, 1044, 1303, 1413,

1476, 1614, 1638, 1640, 1796,  
2227

`\appdef@e` ..... 286

`\appdef@eval` ..... 291

`\appdef@val` ..... 283, 292

`aps.rtx` ..... 18, 19

argument

- `class` ..... 18
- `float` ..... 36
- `text` ..... 24, 41
- `your_tokens_here` ..... 15

argument, optional

- `number` ..... 24

`\array` ..... 48

`\array` ... 1060, 1150, 1174, 1232,  
1351

array document class 9, 41, 45, 47,  
48, 57, 59

array environment .... 2, 41, 46

`\array@array` ..... 1174, 1370

`\array@array@new` ... 1232, 1375

`\array@default` . 806, 1292, 1298,  
1541

`\array@hook` ..... 1050

`\array@LaTeX` ..... 1060, 1351

`\array@ltx` ..... 1150, 1359

`\array@message` ..... 1169, 1246

`\array@row@pre` . 1416, 1479, 1557

`\array@row@pst` . 1498, 1504, 1557

`\array@row@rst` ..... 53

`\array@row@rst` .....

- .. 1557, 1701, 1710, 1854,
- 1858, 1862, 1906, 1910, 1914,
- 1939

`\arraycolsep` 508, 509, 1371, 1812,  
1815

`\arrayrulewidth` 1566, 1580, 1626

`\arraystretch` .....

- .. 1389, 1390, 1433, 1434,
- 1442, 1443, 1464, 1465, 1618,
- 1620

`\AtBeginDocument` . 14, 15, 29, 46

`\AtBeginDocument` ..... 355

`\AtEndDocument` ..... 15, 16

`\AtEndDocument` ..... 357, 367

`\AtEndOfClass` ..... 19

`atveryend document class` 18, 93

`auguide.tex` ..... 2

`\author` ..... 51

## B

`\baselineskip` ..... 12, 22

`\baselineskip` . 1401, 1424, 1456,  
1489

`\baselineskip` ..... 12

`\begin` ..... 15

`\begin@float@pagebreak` ... 750,  
764, 796, 822

`\belowdisplayshortskip` ... 524

`\belowdisplayskip` ..... 523

`\bgroup` ..... 26, 53

`\body@font` ..... 1563, 1598

`\botrule` ..... 53

`\botrule` ..... 1562

`\box` ..... 66

## C

`\c@mpfootnote` .... 617, 618, 744

`\c@secnumdepth` ... 946, 975, 989

`\Call@AfterLastShipout` .... 18

`\Call@AfterLastShipout` 434, 438

`\caption` ..... 30

`\caption` ..... 694

`\cat@letter` ..... 240

`\cat@space` ..... 2127

`\catcode` ..... 19

`\catcode` ..... 478, 486, 899,  
904, 2127, 2153–2159, 2162,  
2164, 2167, 2175, 2185

`\cell@fil` 1593, 1636, 1910, 1914

`\cell@font` 1592, 1599, 1701, 1711,  
1906, 1910, 1914, 1939

`\centering` ..... 20

`\changes` ..... 138–187

`\chardef` 239–241, 835, 895, 2127

`\check@aux` ..... 18

`\check@aux` ..... 362

`<class>` placeholder ..... 18, 19

`class`, argument ..... 18

`\class@documenthook` ... 15, 33

`\class@documenthook` . 354, 437,  
604, 694

<code>\class@enddocumenthook</code>	. 15, 29	<code>\columnwidth</code>	..... 26
<code>\class@enddocumenthook</code>	230, 354	<code>\columnwidth</code>	..... 753
<code>\class@err</code>	..... 202	<code>\contentsline</code>	..... 30, 63, 64
<code>\class@ext@hook</code>	..... 440	<code>\contentsline</code>	.... 674, 691, 692
<code>\class@extension</code>	..... 440	<code>\contentsline@latex</code>	.. 688, 691
<code>\class@extensionfile</code>	.. 18, 19	<code>\copy</code>	..... 1604
<code>\class@extensionfile</code>	.... 440	<code>\copyright</code>	..... 48
<code>\class@info</code>	..... 202,	<code>\count</code>	..... 34, 35
	463, 465, 541, 620, 851, 883,	<code>\count</code>	... 840, 843, 845, 846, 852,
	944, 1142, 1144, 1224, 1247,		1966, 1973
	1250, 1980, 1982, 2218, 2247	<code>\count@</code>	.. 1697, 1703, 1719, 1750
<code>\class@name</code>	..... 10	<code>\count@i</code>	..... 266
<code>\class@name</code>	. 202–204, 223, 232,	<code>\count@ii</code>	..... 266
	233	<code>\countdef</code>	..... 266, 267
<code>\class@warn</code>	..... 202, 234	<code>\crrc</code>	1495, 1498, 1501, 1504, 1509,
<code>\class@warn@end</code>	.. 206, 211, 216,		1585
	229	<code>\cs</code>	. 138, 139, 141, 152–154, 156,
<i>(class customization commands)</i>			157, 159–164, 167, 169, 176,
holder	..... 9		178, 181–183, 187
<code>\ClassError</code>	..... 202, 223	<code>\csname</code>	..... 13, 49
<code>classes.dtx</code>	..... 29	<code>\csname</code>	..... 34, 244, 246, 250,
<code>classes.dtx</code>	document class .. 33		252, 301, 322, 442, 444, 450,
<code>\ClassInfo</code>	..... 204		560, 564, 592, 596, 602, 614,
<code>\classname</code>	..... 62,		615, 643, 685, 686, 775, 777,
	70, 122, 124, 126, 141, 165,		780, 781, 812, 813, 817, 826,
	166, 178, 179, 181, 182, 186		830, 832, 859, 878, 895, 898,
<code>\classoption</code>	..... 168		960, 968, 973, 978, 984, 987,
<code>\ClassWarningNoLine</code>	..... 203		992, 1008, 1013, 1018, 1021,
<code>\clear@document</code>	..... 16–18		1240, 1992, 1993, 2032, 2035,
<code>\clear@document</code>	.. 370, 428, 431		2038, 2040, 2044, 2045, 2053–
<code>\clearpage</code>	..... 17, 18		2055, 2064, 2094, 2217
<code>\clearpage</code>	..... 424, 432	<code>\CT@cell@color</code>	..... 1803
<code>\closeout</code>	..... 378, 816	<code>\CT@color</code>	..... 1798
<code>\cmd</code>	..... 134	<code>\CT@column@color</code>	..... 1801
<code>\col@sep</code>	..... 45	<code>\CT@do@color</code>	..... 1799
<code>\col@sep</code>	..... 1275, 1371	<code>\CT@end</code>	..... 46
<code>\color</code>	..... 1800	<code>\CT@row@color</code>	..... 1802
<code>\color@begingroup</code>	..... 645	<code>\CT@setup</code>	..... 1797
<code>\color@endgroup</code>	..... 650, 729	<code>\CT@start</code>	..... 46
<code>colortbl</code>	document class . 45, 46,	<code>\CurrentOption</code>	..... 455
	57, 59, 91		
<code>\colortbl@message</code>	.. 1169, 1249		
<code>\colrule</code>	..... 53		
<code>\colrule</code>	..... 1562		
<code>\column@fil</code>	..... 1562, 1597		
<code>\column@font</code>	..... 1562, 1596		

## D

<code>\dollarbegin</code>	. 1253, 1276, 1284,
	1372, 1380, 1381
<code>\dollarend</code>	1277, 1285, 1372, 1381
<code>\deadcycles</code>	..... 372

<code>\DeclareRobustCommand</code> . . . . .	20
<code>\DeclareRobustCommand</code> . . . . .	497
<code>\detokenize</code> . . . . .	413, 414
<code>\dimen</code> . . . . .	64
<code>\dimen@</code> . . . . .	863, 864, 866, 867, 869, 870, 1617–1619, 2055, 2056
<code>\dimen@iii</code> . . . . .	237
<code>\dimendef</code> . . . . .	237
<code>\discretionary</code> . . . . .	2125
<code>displaymath</code> environment . . . . .	22
<code>\displaystyle</code> . . . . .	507, 510, 529, 535
<code>\displaywidth</code> . . . . .	506
<code>\do</code> . . . . .	1649, 1665, 1723, 1754
<code>\do@check@aux</code> . . . . .	17, 18
<code>\do@check@aux</code> . . . . .	362
<code>\do@if@floats</code> . . . . .	29, 33
<code>\do@if@floats</code> . . . . .	772
<code>\do@output@cclv</code> . . . . .	433
<code>doc</code> . . . . .	4
<code>\DocInput</code> . . . . .	8
<code>\document</code> . . . . .	14, 15
<code>\document</code> . . . . .	343
document class	
<code>array</code> . . . . .	9, 41, 45, 47, 48, 57, 59
<code>atveryend</code> . . . . .	18, 93
<code>classes.dtx</code> . . . . .	33
<code>colortbl</code> . . . . .	45, 46, 57, 59, 91
<code>ftnright</code> . . . . .	9
<code>hyperref</code> . . . . .	21, 23, 25, 26, 30, 39, 40, 63, 64, 67–69, 89, 91, 93
<code>hyperref<sub>l</sub>package</code> . . . . .	2, 30
<code>hyperref.sty</code> . . . . .	94
<code>lineno</code> . . . . .	92, 94
<code>lineno.sty</code> . . . . .	66, 93
<code>longtable</code> . . . . .	9, 57
<code>ltxdoc</code> . . . . .	4, 8
<code>ltxfront</code> . . . . .	27
<code>ltxgrid</code> . . . . .	9, 15, 18, 26, 94
<code>ltxutil</code> . . . . .	9, 18
<code>multicol</code> . . . . .	89
<code>revtex4</code> . . . . .	18
<code>revtex4-1</code> . . . . .	1
<code>textcase</code> . . . . .	20, 93
document class option	
<code>10pt</code> . . . . .	89
<code>11pt</code> . . . . .	89
<code>12pt</code> . . . . .	89
<code>groupauthors</code> . . . . .	89
<code>hypertext</code> . . . . .	93
<code>lengthcheck</code> . . . . .	91
document environment . . . . .	4, 15
<code>\document@inithook</code> . . . . .	15
<code>\document@inithook</code> . . . . .	345, 349, 353, 354, 690, 1040, 1044, 2227
<code>\doi</code> . . . . .	68
<code>\doi</code> . . . . .	2118
<code>\doibase</code> . . . . .	2118
<code>\doublerulesep</code> . . . . .	1579
<code>\dp</code> . . . . .	657, 853, 869, 870, 882, 1308, 1309, 1332, 1338, 1390, 1434, 1443, 1465, 1620
<b>E</b>	
<code>\edef</code> . . . . .	57
<code>\edef</code> . . . . .	271, 276, 281, 413, 414, 571, 1393, 1806, 1812, 1821, 1833, 1880, 2035, 2171–2173, 2181, 2200
<code>\emph</code> . . . . .	93
<code>\end</code> . . . . .	36
<code>\end@float@anchored</code> . . . . .	751, 752, 758
<code>\end@float@pagebreak</code> . . . . .	762, 765, 796
<code>\endarray</code> . . . . .	46, 51
<code>\endarray</code> . . . . .	1044, 1061, 1151, 1175, 1233, 1494, 1512, 1515, 1518
<code>\endarray@array</code> . . . . .	1175, 1500
<code>\endarray@array@new</code> . . . . .	1233, 1503
<code>\endarray@hook</code> . . . . .	1048, 1049
<code>\endarray@LaTeX</code> . . . . .	1061, 1494
<code>\endarray@ltx</code> . . . . .	51
<code>\endarray@ltx</code> . . . . .	1151, 1497, 1504
<code>\endcsname</code> . . . . .	34, 244, 246, 250, 252, 301, 322, 442, 444, 450, 560, 564, 592, 596, 602, 614, 615, 643, 685, 686, 775, 777, 780, 781, 812, 813, 817, 826, 830, 832, 859, 878, 895, 898, 960, 968, 973, 978, 984, 987, 992, 1008, 1013, 1018, 1021, 1240, 1992, 1993, 2032, 2035,



2038, 2040, 2044, 2045, 2053–  
2055, 2065, 2094, 2217

`\enddocument` . . . . . 17, 18

`\enddocument` . . . . . 362

`\endgroup` . . . . . 15, 17

`\endlinechar` . . . . . 2152

`\endminipage` . . . . . 724

`\endtabular` . . . . .  
. . . 1044, 1062, 1152, 1176,  
1234, 1240, 1508, 1520

endtabular environment . . . . . 52

`\endtabular*` . . . . . 1520

`\endtabular@array` . . . 1176, 1514

`\endtabular@array@new` . . . 1234,  
1517

`\endtabular@hook` . . . 1046, 1047

`\endtabular@LaTeX` . . . 1062, 1508

`\endtabular@ltx` . . . . 1152, 1511

`\endwrite@float` . . . . . 888, 898

`\endwrite@floats` . . . . . 890

environment

- array . . . . . 2, 41, 46
- displaymath . . . . . 22
- document . . . . . 4, 15
- endtabular . . . . . 52
- eqnarray . . . . . 21, 22, 47
- equation . . . . . 22
- figure . . . . . 28
- figure@write . . . . . 28, 33
- longtable . . . . . 29, 36
- minipage . . . . . 23, 28
- tabular . . . . . 2, 41, 46, 47
- thebibliography . . . . . 94

`\eqnarray` . . . . . 21

`\eqnarray` . . . . . 540, 542, 555

eqnarray environment . . . 21, 22, 47

`\eqnarray@fleqn@fixed` . . . . 498

`\eqnarray@LaTeX` . . . . . 498

`\eqncolsep` . . . 531, 534, 543, 544

`\equation` . . . . . 23

`\equation` . . . . . 554

equation environment . . . . . 22

`\errorstopmode` . . . . . 258

`\extrarowheight` . . . . . 1440, 1462

## F

`\f@ur` . . . . . 239, 1743

`\false@sw` . . . 312, 318, 323, 335,  
340, 1079, 1082, 1085, 1088,  
1091, 1094, 1097, 1100, 1103,  
1106, 1109, 1112, 1115, 1118,  
1121, 1124, 1127, 1130, 1133,  
1136, 1139, 1186, 1189, 1192,  
1195, 1198, 1201, 1204, 1207,  
1210, 1213, 1216, 1219, 1222,  
2130, 2245

figure environment . . . . . 28

figure@write environment 28, 33

`\figuresname` . . . . . 29

file

- .aux . . . . . 40
- .cls . . . . . 19
- .dtx . . . . . 4
- .rtx . . . . . 19
- 00readme . . . . . 4
- . . . . . 90, 92
- aps.rtx . . . . . 18, 19
- auguide.tex . . . . . 2
- classes.dtx . . . . . 29
- doc . . . . . 4
- kernel . . . . . 2, 10, 71
- ltxfloat.dtx . . . . . 23
- nameref.sty . . . . . 39
- package . . . . . 2, 9
- pdfmark.def . . . . . 39
- README . . . . . 2
- template.aps . . . . . 2

`\file` . 29, 81, 84, 90, 91, 100, 106,  
108, 111, 112, 115, 116, 122,  
124, 126, 128, 131

float, argument . . . . . 36

`\float@end@tag` . . . . . 36

`\float@end@tag` . . . . . 903

`\floatingpenalty` . . . . . 26, 27

`\floats@sw` . . . . . 32, 33

`\floats@sw` . . . 733, 738, 773, 803

`\flushing` . . . . . 490

`\fmtversion` . . . . . 303

`\font@submax` . . . . . 386, 389

`\fontsubfuzz` . . . . . 386

`\footins` . . . . . 26

<code>\footins</code>	631, 721	<code>\hline@rule</code>	1564, 1572, 1587, 1602
<code>\footmark</code>	24	<i>(hooks and bookkeeping)</i>	placeholder
<code>\footnote</code>	23, 24, 28		17
<code>\footnote</code>	557, 664	<code>\href</code>	68, 69
<code>\footnote@latex</code>	605	<code>\href</code>	2118, 2148
<code>\footnotemark</code>	23, 24, 26	<code>\href@</code>	2148–2150
<code>\footnotemark</code>	557	<code>\href@Hy</code>	2148, 2217
<code>\footnotesep</code>	647, 656	<code>\href@Hy@ltx</code>	2149, 2224
<code>\footnotesize</code>	654	<code>\hrule</code>	12
<code>\footnotetest</code>	557	<code>\hsize</code>	14
<code>\footnotetext</code>	23, 24	<code>\ht</code>	853, 863, 866, 867, 881, 1389, 1433, 1439, 1461, 1617, 1618
<code>\footnotetext</code>	581	<code>\Hy@ActiveCarriageReturn</code>	2189, 2222
<code>\frontmatter@footnotetext</code>	27	<code>\Hy@ActiveCarriageReturn@ltx</code>	2186, 2222
<code>\frontmatter@makefntext</code>	27	<code>\Hy@footnote@currentHref</code>	574, 615
<code>\frustrut</code>	1562, 1563, 1600	<code>\Hy@href</code>	2149, 2150
<code>ftnright</code> document class	9	<code>\Hy@raisedlink</code>	1032
<code>\ftype@</code>	34	<code>\Hy@RemovePercentCr</code>	2202, 2211, 2223
<code>\fullinterlineskip</code>	265, 711	<code>\Hy@RemovePercentCr@ltx</code>	2206, 2223
<code>\futurelet</code>	1566, 1573	<code>\Hy@ReturnAfterElseFi</code>	2190
<b>G</b>			
<code>\gapptdef</code>	230, 268	<code>\Hy@ReturnAfterFi</code>	2194, 2210
<code>\GetFileInfo</code>	39	<code>\Hy@safe@activestruel</code>	2182
<code>\glet@environment</code>	217, 242	<code>\Hy@tempa</code>	2188, 2189, 2200, 2204
<code>\glossary</code>	663	<code>\hyper@@normalise</code>	2191, 2195, 2220
<code>groupauthors</code> document class option	89	<code>\hyper@@normalise@ltx</code>	2199, 2220
<b>H</b>			
<code>\H@@footnotetext</code>	23	<code>\hyper@anchorend</code>	40
<code>\H@@footnotetext</code>	607, 608, 745	<code>\hyper@anchorend</code>	1034
<code>\H@mpfootnotetext</code>	609, 610, 745	<code>\hyper@anchorstart</code>	40
<code>\H@refstepcounter</code>	38	<code>\hyper@anchorstart</code>	1033
<code>\H@refstepcounter</code>	950, 1039	<code>\hyper@hash</code>	2168, 2169
<code>\H@svsec</code>	947, 951, 969, 985, 1002, 1009, 1019	<code>\hyper@last</code>	40
<code>\halign</code>	51, 57	<code>\hyper@linkend</code>	40
<code>\halign@</code>	238, 528	<code>\hyper@linkend</code>	576, 1036
<code>\hangindent</code>	67	<code>\hyper@linkstart</code>	40
<code>\hb@xt@</code>	511, 536, 769, 2062, 2074	<code>\hyper@linkstart</code>	574, 1035
<code>\hbox</code>	22, 67	<code>\hyper@makecurrent</code>	1031
<code>\hline</code>	54	<code>\hyper@n@rmalise</code>	2183, 2221
<code>\hline</code>	1072, 1162, 1565		
<code>\hline@LaTeX</code>	1072, 1565		
<code>\hline@ltx</code>	1162, 1569		
<code>\hline@rule</code>	54		

`\hyper@normalise@ltx` 2187, 2221  
`\hyper@normalise` ... 2148–2150, 2219  
`\hyper@normalise@ltx` 2160, 2219  
`\hyper@tilde` ..... 2177–2179  
`\hyperanchor` ..... 40  
`hyperref` document class . 21, 23, 25, 26, 30, 39, 40, 63, 64, 67–69, 89, 91, 93  
`hyperref_lpackage` document class ..... 2, 30  
`hyperref.sty` document class . 94  
`hypertext` document class option 93  
`\hypertext@enable@ltx` ..... 68  
`\hypertext@enable@ltx` ... 2128

## I

`\ialign` ..... 49  
`\ialign` ..... 1394, 1447  
`\if` ..... 49  
`\if` ..... 309, 321, 1386, 1452  
`\if@filesw` . 377, 394, 1991, 2020  
`\if@firstamp` ..... 1806, 1809  
`\if@nobreak` ..... 934  
`\if@noskipsec` ..... 927  
`\ifeof` ..... 325  
`\iffalse` ..... 55, 1314, 1319  
`\IfFileExists` ..... 441  
`\IfFormatAtLeastTF` ... 302, 305  
`\ifhbox` ..... 326  
`\ifinner` ..... 328  
`\ifmode` ..... 329  
`\ifodd` ..... 331, 1970  
`\ifvbox` ..... 332  
`\ifvoid` ..... 334  
`\ignorespaces` .....  
... 647, 1526, 1539, 1696, 1702, 1883, 1885, 1887, 1906, 1910, 1914, 1931, 1939, 2061, 2070, 2082, 2102  
`\immediate` .. 378, 778, 815, 913, 1993, 2043, 2231, 2237  
`\incompatible@package` .... 220  
`\index` ..... 662  
`\init@hyperref` ..... 1029  
`\insert` ..... 27

`\insert` ..... 631, 721  
`\insert@column` . 1179, 1239, 1694  
`\insert@column@array` 1179, 1694  
`\insert@column@array@new` 1239, 1699  
`\interdisplaylinepenalty` .. 47  
`\interfootnotelinepenalty` 655  
`\interlinepenalty` 655, 966, 1006  
`\interlineskip` ..... 27  
`\intertabularlinepenalty` .. 47  
`\intertabularlinepenalty` 1300, 1301, 1303, 1305, 1320  
`\intextsep` ..... 764, 765  
`\item` 104, 114, 117, 121, 123, 125, 130, 132  
`\itemindent` ..... 2093  
`\itshape` ..... 617, 626

## J

`\j@nk` ..... 341

## K

`kernel` ..... 2, 10, 71

## L

`\l@` ..... 64  
`\l@sections` ..... 64  
`\l@sections` ..... 2049  
`\l@section` ..... 2003  
`\label` ..... 661, 672  
`\lastbox` ..... 551, 714, 820  
`\LaTeX` . 42, 71, 72, 82, 92, 93, 95, 106, 110, 113, 115, 187  
`\LaTeXe` ..... 76, 101, 197  
`\leftmargin` ... 2106, 2108, 2113  
`\leftmargini` ..... 2003  
`\leftmarginii` ..... 2004  
`\leftmarginiii` ..... 2005  
`\leftmarginiv` ..... 2006  
`\leftmarginv` ..... 2007  
`\leftmarginvi` ..... 2008  
`\leftskip` ..... 67  
`\leftskip` . 492, 2054, 2056, 2115  
`lengthcheck` document class option ..... 91  
`\lengthcheck@sw` ..... 798

<code>\let</code> .....	45, 65	<code>ltxfront</code> document class	.... 27
<code>\let@environment</code> .....	11	<code>ltxgrid</code> document class	9, 15, 18, 26, 94
<code>\let@environment</code> .	242, 788, 789, 791, 810, 811	<code>\ltxu@dotsep</code> .....	2002, 2077
<code>\limits</code> .....	2208	<code>ltxutil</code> document class	... 9, 18
<code>lineno</code> document class	... 92, 94	<b>M</b>	
<code>lineno.sty</code> document class	66, 93	<code>\make@footnotetext</code> ....	26, 27
<code>\lineskip</code>	1401, 1424, 1455, 1489	<code>\make@footnotetext</code> .....	623
<code>\linewidth</code> .....	528, 2107–2109	<code>\makeatletter</code> .	32, 382, 454, 1989
<code>\list</code> .....	67	<code>\makeatother</code> .....	35
<code>\list</code> .....	2085	<code>\makelabel</code> .....	2096
<code>\listparindent</code> .	2092, 2105, 2112	<code>\maketitle</code> .....	26
<code>\LoadClass</code> .....	19	<code>\maketitle</code> .....	60
<code>\long</code> .....	71	<code>\mark</code> .....	29
<code>longtable</code> document class	. 9, 57	<code>\mathchardef</code> .....	836, 837
<code>longtable</code> environment	.. 29, 36	<code>\mathindent</code> .....	22
<code>\loop</code> .....	14	<code>\meaning</code> .....	263
<code>\loopuntil</code> .....	337	<code>\MessageBreak</code> .....	388
<code>\loopwhile</code> .....	337	<code>minipage</code> environment	... 23, 28
<code>\lrstrut</code> .....	1563, 1564, 1601	<code>\minipagefootnote@drop</code> ....	26
<code>\LT@array</code> .....	9	<code>\minipagefootnote@drop</code>	651, 697
<code>ltxfloat.dtx</code> .....	23	<code>\minipagefootnote@foot</code> ...	697
<code>\ltx@contentsline</code> .....	672	<code>\minipagefootnote@here</code> ....	31
<code>\ltx@def@footproc</code> .....	24, 25	<code>\minipagefootnote@here</code> ...	695, 697, 759
<code>\ltx@def@footproc</code>	559, 567, 582, 590	<code>\minipagefootnote@init</code>	697, 756
<code>\ltx@footmark</code> .....	24	<code>\minipagefootnote@pick</code>	635, 697
<code>\ltx@footmark</code> .....	557	<code>\mpfootnote</code> .....	620
<code>\ltx@footnote@pop</code> ....	741, 747	<code>multicol</code> document class	.... 89
<code>\ltx@foottext</code> .....	24	<code>\multicolumn</code> ..	1057, 1147, 1521
<code>\ltx@foottext</code> .....	557	<code>\multicolumn@LaTeX</code> .	1057, 1521
<code>\ltx@make@current@footnote</code>	557	<code>\multicolumn@ltx</code> ...	1147, 1528
<code>\ltx@mathindent</code> .....	22	<code>\multiply</code> .....	842
<code>\ltx@mathindent</code> .....	518, 548	<code>\multispan</code> .....	1522, 1529
<code>\ltx@stp@footproc</code> .....	24, 25	<b>N</b>	
<code>\ltx@stp@footproc</code>	563, 568, 583, 598	<code>nameref.sty</code> .....	39
<code>\ltx@thempfootnote</code> ...	618, 621	<code>\NC@list</code> .....	1718, 1749
<code>\ltx@xfootmark</code> .....	566, 567	<code>\newbox</code> .....	1613
<code>\ltx@xfootnote</code> .....	24	<code>\newcount</code> .....	1300, 1302
<code>\ltx@xfootnote</code> .....	557	<code>\newenvironment</code> .....	36
<code>\ltx@xfoottext</code> .....	581, 582	<code>\newlabel</code> .....	30
<code>\ltx@yfootmark</code> .....	566, 568	<code>\newlabel</code> .....	680
<code>\ltx@yfootnote</code> .....	557, 562	<code>\newlinechar</code> .....	912
<code>\ltx@yfoottext</code> .....	581, 583	<code>\newwrite</code> .....	774, 1992
<code>ltxdoc</code> document class	.... 4, 8		

<code>\noalign</code>	. 1307, 1309–1311, 1328, 1341, 1345, 1349, 1564, 1566, 1570, 1574, 1586
<code>\noexpand</code>	. 263, 264, 1394, 1447, 2202
<code>\normalfont</code>	..... 626
number, optional argument	... 24
<code>\numberline</code>	..... 66, 67
<code>\numberline@sections</code>	... 2059, <a href="#">2073</a>
<b>O</b>	
<code>\obeylines</code>	..... 909
<code>\obsolete@command</code>	..... 205
<code>\oneapage</code>	..... 796
<code>\openout</code>	..... 779, 1993
<code>\OptionNotUsed</code>	..... 19
<code>\other</code>	..... 241
<b>P</b>	
<code>\p@equation</code>	..... 500, 516
package	..... 2, 9
<code>\package@name</code>	..... 195, 196
<code>\PackageInfo</code>	..... 196
<code>\par</code>	..... 66, 71
<code>\parindent</code>	..... 22, 67
<code>\parindent</code>	767, 2056, 2074, 2105, 2112
<code>\parsep</code>	..... 2100
<code>\parshape</code>	..... 67
<code>\parshape</code>	..... 2109
<code>\parskip</code>	..... 22
<code>\parskip</code>	..... 522, 2100
<code>\parskipfillskip</code>	..... 22
<code>\partopsep</code>	..... 521
<code>\pbox@hook</code>	..... 1637, 1939
<code>\pdfendlink</code>	..... 2141
<code>pdfmark.def</code>	..... 39
<code>\pdfoutput</code>	..... 2130
<code>\pdfstartlink</code>	..... 2137
<code>\pdfstartlink@attr</code>	..... 68
<code>\pdfstartlink@attr</code>	..... <a href="#">2128</a>
<code>\pdfstringdefDisableCommands</code>	..... 28
<code>\phantomsection</code>	..... 39
<code>\phantomsection</code>	. 947, 952, 1002, 1030
placeholder	<i>&lt;class&gt;</i> ..... 18, 19 <i>&lt;class customization commands&gt;</i> ..... 9 <i>&lt;hooks and bookkeeping&gt;</i> .. 17 <i>&lt;read main .aux and final checks&gt;</i> ..... 17 <i>&lt;your document here&gt;</i> ..... 8
<code>\prep@math</code>	..... 22
<code>\prep@math</code>	..... <a href="#">550</a>
<code>\prep@math@patch</code>	..... 22, 23
<code>\prep@math@patch</code>	..... <a href="#">550</a>
<code>\prepdef</code>	..... 12
<code>\prepdef</code>	..... 268, 347, 554, 555, 695, 812, 813, 956, 958, 1046, 1048, 1051, 1409, 1472
<code>\prepnex@tok</code>	..... 1721, 1752
<code>\prevdepth</code>	..... 12
<code>\prevdepth</code>	..... 265
<code>\print@float</code>	..... 33
<code>\print@float</code>	..... <a href="#">795</a>
<code>\printfigures</code>	..... 29
<code>\ProcessOptions</code>	..... 19
<code>\protected</code>	..... 411
<code>\protected@edef</code>	..... 25
<code>\protected@edef</code>	.. 615, 642, 954
<code>\protected@write</code>	..... 670, 679
<code>\protected@xdef</code>	..... 600
<code>\providecommand</code>	..... 438, 1030–1037, 1050, 2118, 2121, 2123, 2124, 2126, 2127
<code>\ProvidesFile</code>	..... 6
<code>\ProvidesPackage</code>	..... 19
<b>R</b>	
<code>\raggedleft</code>	..... 20
<code>\raggedright</code>	..... 20
<i>&lt;read main .aux and final checks&gt;</i> placeholder	..... 17
README	..... 2
<code>\realfootnote</code>	..... 26
<code>\RecordChanges</code>	..... 37

<code>\refstepcounter</code>	38	<code>\set@listindent@</code>	2111		
<code>\relax</code>	38, 41	<code>\set@listindent@parshape</code>	2104, 2117		
<code>\repeat</code>	14	<code>\set@tocdim@pagenum</code>	66, 67		
<code>\replace@command</code>	210	<code>\set@tocdim@pagenum</code>	2052, 2069		
<code>\replace@environment</code>	215	<code>\set@typeset@protect</code>	1400, 1423, 1488, 1524, 1533		
<code>\RequirePackage</code>	25, 27, 34	<code>\shipout</code>	11		
<code>\reserved@a</code>	1567, 1573, 1577, 1578, 1649, 1664	<code>\show@box@size@sw</code>	850		
<code>\reset@font</code>	654	<code>\showboxbreadth</code>	258		
<code>revtex4</code> document class	18	<code>\showboxdepth</code>	258		
<code>revtex4-1</code> document class	1	<code>\showoutput</code>	261		
<code>\robustify@contents</code>	660	<code>\skip@</code>	2058, 2061		
<code>\romannumeral</code>	2094	<code>\spacefactor</code>	571, 578		
<code>\rule</code>	647	<code>\special</code>	2132, 2133		
<code>\rvtx@enddocument@patch</code>	408	<code>\splitmaxdepth</code>	657		
<code>\rvtx@enddocument@patch@end</code>	417, 424	<code>\splittopskip</code>	656		
<code>\rvtx@enddocument@patch@more</code>	419, 422	<code>\stepcounter</code>	499, 516, 599		
<code>\rvtx@ifformat@geq</code>	13	<code>\stop</code>	224		
<code>\rvtx@ifformat@geq</code>	302, 343, 362, 408, 2244	<code>\StopEventually</code>	5		
<b>S</b>				<code>\string</code>	206, 211, 222, 620, 665–667, 670, 680, 944, 1980, 1982, 2044, 2171–2173, 2181, 2218, 2247
<code>\samepage</code>	47	<code>\strutbox</code>	648, 657, 1389, 1390, 1433, 1434, 1439, 1443, 1461, 1465, 1618, 1620		
<code>\samepage</code>	1303	<code>\subsection</code>	98, 137		
<code>\save@decl</code>	1726, 1763	<code>\substyle@ext</code>	19		
<code>\say</code>	11	<code>\switch@array</code>	41, 47		
<code>\say</code>	263, 1577	<code>\switch@array</code>	1045, 1054		
<code>\saythe</code>	11	<code>\switch@array@info</code>	1169, 1226		
<code>\saythe</code>	263	<code>\switch@hyperref@href</code>	2216, 2227		
<code>\sc</code>	30, 92	<code>\switch@tabular</code>	41, 47		
<code>\sec@upcase</code>	41	<code>\switch@tabular</code>	1045, 1054		
<code>\sec@upcase</code>	1043	<b>T</b>			
<code>\section</code>	88, 823	<code>\t@</code>	236, 238		
<code>\Sectionformat</code>	39	<code>\tab@rule</code>	54		
<code>\set@arstrutbox</code>	54	<code>\tab@rule</code>	1562, 1563, 1584		
<code>\set@arstrutbox</code>	1407, 1431, 1602	<code>\tabcolsep</code>	1275, 1633–1635, 1821		
<code>\set@display@protect</code>	2230, 2236	<code>\tableft@skip</code>	1425, 1490, 1630		
<code>\set@eqnarray@skips</code>	549	<code>\tableftsep</code>	1633, 1707, 1818		
<code>\set@footnotefont</code>	28	<code>\tableofcontents</code>	86		
<code>\set@footnotefont</code>	639, 653				
<code>\set@footnotewidth</code>	26, 28				
<code>\set@footnotewidth</code>	623				
<code>\set@footnotewidth@one</code>	659				
<code>\set@listindent</code>	2101, 2117				

<code>\tabmid@skip</code>	.. 1410, 1473, 1631	<code>\toc@post@auto</code>	..... 2009
<code>\tabmidsep</code>	.... 1634, 1708, 1824	<code>\toc@pre</code>	..... 65
<code>\tabright@skip</code>	. 1414, 1477, 1632	<code>\toc@pre</code>	..... 1988, 1999
<code>\tabrightsep</code>	.. 1635, 1709, 1827	<code>\toc@pre@auto</code>	..... 65
<code>\tabskip</code>	..... 22, 49	<code>\toc@pre@auto</code>	..... 2009
<code>\tabskip</code>	..... 504,	<code>\toc@setindent</code>	. 2012–2015, 2031
	507, 510, 512, 518, 529, 535,	<code>\toc@writedimen</code>	2022–2027, 2042
	537, 1395, 1410, 1414, 1425,	<code>\tocdim@appendix</code>	..... 2007
	1449, 1473, 1477, 1490	<code>\tocdim@min</code>	... 2032, 2038, 2084
<code>\tabular</code>	..... 41	<code>\tocdim@pagenum</code>	.... 2008, 2071
tabular environment	2, 41, 46, 47	<code>\tocdim@paragraph</code>	..... 2006
<code>\tabular@hook</code>	..... 1051, 1052	<code>\tocdim@section</code>	..... 66
<code>\tabularnewline</code>	1397, 1420, 1458,	<code>\tocdim@section</code>	..... 2003
	1485	<code>\tocdim@subsection</code>	..... 2004
<code>\tally@float</code>	..... 34	<code>\tocdim@subsubsection</code>	... 2005
<code>\tally@float</code>	..... 835	<code>\tocleft@pagenum</code>	..... 2057
template.aps	..... 2	<code>\tocleft@section</code>	..... 66
<code>\TeX</code>	..... 68, 76, 115, 186	<code>\tocmax@section</code>	..... 66
text, argument	..... 24, 41	<code>\toks</code>	..... 1739, 1794
<code>\textasciitilde</code>	..... 2179	<code>\toks@</code>	... 269, 271, 274, 276, 279,
textcase document class	. 20, 93		281, 896, 913, 915
<code>\texttt</code>	.. 53, 111, 112, 119, 128,	<code>\toks@ii</code>	..... 12
	131, 134, 138	<code>\toks@ii</code>	. 270, 271, 275, 276, 280,
<code>\textunderscore</code>	2173, 2174, 2176		281, 296
<code>\textwidth</code>	..... 865	<code>\toksdef</code>	..... 296
<code>\thanks</code>	..... 43, 47, 53	<code>\toprule</code>	..... 53
<code>\the@toks</code>	1695, 1697, 1700, 1703,	<code>\toprule</code>	..... 1562
	1720, 1739, 1751, 1794	<code>\topsep</code>	..... 520
thebibliography environment	94	<code>\total@float</code>	..... 34
<code>\theequation</code>	..... 500, 516	<code>\total@float</code>	..... 799, 835
<code>\thefootnote</code>	..... 23	<code>\traceoutput</code>	..... 11
<code>\thempfn</code>	..... 23	<code>\tracefn</code>	..... 260
<code>\thempfn</code>	..... 593, 600, 743	<code>\tracingall</code>	..... 11
<code>\thempfootnote</code>	..... 25	<code>\tracingcommands</code>	..... 255
<code>\thempfootnote</code>	... 619, 621, 743	<code>\tracinglostchars</code>	..... 256
<code>\thempfootnote@latex</code>	. 617, 619	<code>\tracingmacros</code>	..... 257
<code>\thepage</code>	..... 674, 680	<code>\tracingonline</code>	..... 255
<code>\thr@@</code>	..... 11	<code>\tracingoutput</code>	..... 11
<code>\thr@@</code>	. 237, 511, 536, 1735, 1786	<code>\tracingoutput</code>	..... 256
<code>\title</code>	..... 41	<code>\tracingpages</code>	..... 256
<code>\toc@font</code>	..... 64	<code>\tracingparagraphs</code>	..... 257
<code>\toc@font</code>	..... 2001, 2010	<code>\tracingplain</code>	..... 11
<code>\toc@letdimen</code>	.. 2016, 2017, 2037	<code>\tracingplain</code>	..... 254
<code>\toc@post</code>	..... 65	<code>\tracingrestores</code>	..... 257
<code>\toc@post</code>	..... 1996, 2000	<code>\tracingstats</code>	..... 255
<code>\toc@post@auto</code>	..... 65		

`\trigger@float@par` .. 809, 812, 813, 819  
`\triggerpar` ..... 795, 809  
`\true@sw` .... 310, 317, 323, 335, 340, 350, 1077, 1184, 2130, 2242, 2245  
`\tw@` ..... 11  
`\typeout` ..... 71  
`\typeout` ..... 2228  
`\typeout@ltx` ..... 2234, 2241  
`\typeout@org` ..... 2228, 2240

### U

`\unhbox` ..... 2062, 2079  
`\unhcopy` ..... 1604  
`\unrestored@protected@xdef` 593  
`\unskip` 726, 1331, 1337, 1563, 1564, 1696, 1702, 1883, 1885, 1887, 1906, 1910, 1914, 2061  
`\unvbox` ..... 702, 716, 721, 731  
`\unvcopy` ..... 713, 862  
`\uppercase` ..... 41  
`\url` ..... 68  
`\url` ..... 26, 66, 73, 2118  
`\URL@prefix` ..... 2118

### V

`\vadjust` ..... 648  
`\vbox` .. 701, 862, 1386, 1452, 1542  
`\vcenter` ..... 53  
`\vcenter` . 1386, 1452, 1543, 1547, 1550  
`\voidb@x` ..... 698  
`\vrule` 1308, 1309, 1333, 1339, 1389, 1433, 1441, 1464, 1619  
`\vsize` ..... 14  
`\vsplit` ..... 27  
`\vtop` ..... 713, 1386, 1452, 1541

### W

`\write` .... 913, 2043, 2231, 2237  
`\write@@float` ..... 887, 891  
`\write@float` ..... 29, 33, 36  
`\write@float` ..... 785, 887  
`\write@floatline` ..... 36  
`\write@floatline` ..... 901, 903

`\write@floats` ..... 786, 887

### X

`\x` ..... 413, 416  
`\xdef` ..... 458, 474, 1447, 1470

### Y

`\y` ..... 414, 416  
`your_tokens_here`, argument . 15  
*<your document here>* placeholder 8

### Z

`\z@skip` .. 492–494, 507, 512, 529, 537, 648, 760, 1395, 1401, 1424, 1630–1632, 2062



## Change History

4.0a	General: 4.0d had twoside option setting twoside switch to false . . . 2	General: Added localization of <code>\figuresname</code> . . . . . 2
	comma not space between email and homepage . . . . . 2	Added localization of <code>\tablesname</code> . . . . . 2
	Initial version . . . . . 2	AO: all code for 10pt is in this module. . . . . 2
	Move after process options, so <code>\clearpage</code> not in scope of <code>twocolumn</code> . . . . . 2	AO: all code for 11pt is in this module. . . . . 2
	Move after process options, so the following test works . . . . . 2	AO: all code for 12pt is in this module. . . . . 2
	multiple preprint commands . . . . . 2	AO: made <code>aps.rtx</code> part of <code>revtex4.dtx</code> . . . . . 2
	print homepage . . . . . 2	AO: remove duplicates . . . . . 2
	protect against hyperref revtex kludges which are not needed now . . . . . 2	call <code>\print@floats</code> . . . . . 2
	Rearrange the ordering so numerical ones come first. AO: David, what does this mean? . . . 2	Defer assignment until <code>\AtBeginDocument</code> time. . . . . 2
	single space footnotes . . . . . 2	Defer decision until <code>\AtBeginDocument</code> time . . . . . 2
	use font-dependent spacing . . . . . 2	Define three separate environments, defer assignment to <code>\AtBeginDocument</code> time. . . . . 2
4.0b	<code>\@mpmakefntext</code> : 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. . . . . 32	First modifications by Arthur Ogawa (mailto:arthur_ogawa at sbcglobal dot net) . . . . . 2
	<code>\caption</code> : Support the hack with <code>\prepdef</code> , and delay until <code>\AtBeginDocument</code> time, since hyperref clobbers <code>\caption</code> . . . 30	Frank Mittelbach, has stated in <code>multicol</code> : “The kernel command <code>\@footnotetext</code> should not be modified.” Thus, I have removed David Carlisle's redefinition of that command. Note, however, that later versions of <code>multicol</code> do not require this workaround. Belt and suspenders. . . . . 2
	<code>\print@float</code> : only execute if there really were floats of the given type . . . . . 33	Move this “complex” option to the front, where it can be overridden by “simple” options. . . 2
	<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. . . . . 36	New option . . . . . 2
		One-line caption sets flush left. . . 2
		only execute if appropriate . . . . . 2
		Processing delayed to <code>\AtBeginDocument</code> time . . . . . 2
		Removed invocation of nonexistent class option <code>groupauthors</code> and all other class options that should only be invoked by the document.

	(Otherwise precedence of class options does not work.) . . . . .	2		eprint takes an optional argument, syntactical only in this case. . . . .	2
	Restore all media size class option of <code>\_classes.dtx</code> . . . . .	2		make longtable trigger the head, too . . . . .	2
	Stack <code>\preprint</code> args flush right at right margin. . . . .	2		More features and bug fixes: compatability with longtable and array packages. Now certainly incompatible with multicol. . . . .	2
4.0c	<code>\@mpmakefnctext</code> : (AO, 110) Install hooks for endfloats processing	32		New option . . . . .	2
	<code>\@ssect</code> : (AO, 116) Hyperref compatibility . . . . .	39	4.0e	<code>\@mpmakefnctext</code> : (AO, 221) Remove samepage command from <code>@xfloat@prep</code> : If the float can break over pages, we want better control. . . . .	32
	<code>\endarray</code> : (AO, 130) Interference from array package . . . . .	41		General: adornments above and below. . . . .	2
	<code>\print@float</code> : *-form mandates pagebreak at each float; only print section head if there is something there. . . . .	33		Bug fixes and minor new features: title block affiliations can have ancillary data, just like authors; clearpage processing revamped, with floats staying in order; widetext ornaments. . . . .	2
	General: (AO, 115) If three or more preprints specified, set on single line, with commas. . . . .	2		New option <code>showkeys</code> . . . . .	2
	(AO, 129) section* within appendix was producing <code>appendixname</code> . . . . .	2	4.0f	<code>\@ssect</code> : (AO, 404) Hyperref compatibility . . . . .	39
	*-form mandates pagebreak . . . . .	2		General: Last bug fixes before release. . . . .	2
	also spelled “acknowledgements”. . . . .	2	4.0rc1	General: grid changes with push and pop . . . . .	2
	Do not put by REVTeX in every page foot . . . . .	2		Running headers always as if two-sided . . . . .	2
	grid changes via <code>ltxgrid</code> procedures . . . . .	2	4.0rc4	General: hyperref is no longer loaded via class option: use a <code>usepackage</code> statement instead . . . . .	2
	grid changes with <code>ltxgrid</code> . . . . .	2	4.1a	<code>\@mpmakefnctext</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. . . . .	32
	Insert procedure <code>\checkindate</code> . . . . .	2			
	Lose compatability mode. . . . .	2			
	New <code>ltxgrid</code> -based code, other bug fixes . . . . .	2			
	New option “checkin” . . . . .	2			
	Prevent an inner footnote from performing twice . . . . .	2			
4.0d	<code>\@mpmakefnctext</code> : (AO, 127) Floats placed [h] to allow page breaks	32			
	(AO, 224) Hyperref compatibility. . . . .	32			
	<code>\print@float</code> : Allow things to break over pages by setting <code>array@default</code> . . . . .	33			
	General: (AO, 174) kernel fix . . . . .	23			
	Also alter how lists get indented. . . . .	2			
	But alternative spelling is deprecated. . . . .	2			

<code>\@p@pfilename</code> : Class extension mechanism	(AO, 478) <code>\ds@letterpaper</code> , so that “letterpaper really is the default” . . . . .	2
<code>\@pushfilename@ltx</code> and <code>\@p@pfilename@ltx</code> . . . . .	(AO, 488) Change processing of options to allow an unused option to specify society and journal . . . . .	2
<code>\class@enddocumenthook</code> : <code>\class@documenthook</code> is the last <code>\AtBeginDocument</code> token now . . . . .	Class extension mechanism <code>\class@extension</code> , <code>\class@extensionfile</code> , and <code>\class@ext@hook</code> . . . . .	18
<code>\document</code> : Get rid of <code>\set@typesize@hook</code> , <code>\set@pica@hook</code> and the <code>\normalsize</code> directive . . . . .	For natbib versions before 8.21, <code>\NAT@sort</code> was consulted only as natbib was being read in. Now it is fully dynamic. . . . .	2
<code>\eqnarray@fleqn@fixed</code> : (AO, 475) I had not properly reproduced the LaTeX macro <code>\eqnarray</code> . . . . .	4.1b	21
<code>\footnote</code> : (AO) Remove code that avoided changes to <code>\@xfootnotemark</code> . . . . .	<code>\@mkpream</code> : (AO, 505) try to accommodate <code>colortbl</code> . . . . .	57
<code>\ltx@make@current@footnote</code> : (AO, 438) Complete rewrite of footnote macros. . . . .	<code>\@mkpream@relax</code> : (AO, 505) try to accommodate <code>colortbl</code> . . . . .	59
<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.) . . . . .	<code>\@mpmakefntext</code> : No need to protect against undefined <code>\float@sw</code> . . . . .	32
<code>\robustify@contents</code> : (AO) Make <code>\addtocontents</code> a <code>\long\def</code> ; gobble up <code>\footnote</code> . . . . .	<code>\@tabarray</code> : (AO, 505) try to accommodate <code>colortbl</code> . . . . .	46
<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.) . . . . .	<code>\@tabular</code> : (AO, 505) try to accommodate <code>colortbl</code> . . . . .	45
General: (AO, 451) “Cannot have more than 256 cites in a document” . . . . .	<code>\array</code> : (AO, 505) try to accommodate <code>colortbl</code> . . . . .	49
(AO, 457) Endnotes to be sorted in with numerical citations. . . . .	<code>\do@if@floats</code> : No need to protect against undefined <code>\float@sw</code> . . . . .	33
(AO, 459) do not assume <code>\class@name</code> is defined . . . . .	<code>\doibase</code> : (AO, 487) Support for video figures and the <code>\setfloatlink</code> command . . . . .	68
(AO, 460) “Proper style is “FIG. 1. . . . .” (no colon)” . . . . .	<code>\endarray</code> : Patch the array package even later: after all package patches go in. . . . .	41
(AO, 461) Change the csname <code>revtex</code> uses from <code>@dotsep</code> to <code>ltxu@dotsep</code> . The former is understood in mu. (What we wanted was a dimension.) . . . . .	<code>\floats@sw</code> : Default assignment of <code>\float@sw</code> now, not at <code>\AtBeginDocument</code> time. . . . .	32
	<code>\init@hyperref</code> : Acquire <code>hyperref</code> <code>savoire</code> . . . . .	40
	<code>\ltx@contentsline</code> : Refine toc processing: provide default. . . . .	30
	<code>\print@float</code> : If class option <code>lengthcheck</code> is in effect, log the height of this float class. . . . .	33
	<code>\switch@array</code> : (AO, 505) Try to accommodate <code>colortbl</code> . . . . .	44

<code>\total@float</code> : Tally and log the height of a float class . . . . .	34		
General: (AO) Implement bibnotes through			
<code>\frontmatter@footnote@produce</code> instead of <code>\bibnotes@sw</code> . . . . .	2		
(AO) No longer need to test <code>\chapter</code> as of <code>natbib</code> version 8.2 . . . . .	2		
(AO) No longer use <code>\secnumarabic@sw</code> , instead use <code>\setup@secnums</code> . . . . .	2		
(AO) Provide more diagnostics when <code>\@society</code> is assigned. . . . .	2		
(AO) Structure the Abstract using the <code>bibliography</code> environment . . . . .	2		
(AO) coordinate <code>\if@twoside</code> with <code>\twoside@sw</code> . . . . .	2		
(AO) make settings at class time instead of deferring them to later. . . . .	2		
(AO) provide option <code>longbibliography</code> . . . . .	2		
(AO, 455) Be nice to a list within the abstract (assign <code>\totalleftmargin</code> ). . . . .	2		
Add <code>\@hangfroms@section</code> . . . . .	2		
Add option reprint, opposite of preprint, and preferred alternative to <code>twocolumn</code> . . . . .	2		
As with author formatting, rag the right more, and assign <code>\totalleftmargin</code> . Also neutralize			
<code>\def@after@address</code> . . . . .	2		
Break out			
<code>\@caption@fignum@sep</code> . . . . .	2		
Class option galley sets			
<code>\preprintsty@sw</code> to false . . . . .	2		
Code relating to new syntax for <code>frontmatter</code> has been placed in <code>ltxfront.dtx</code> . . . . .	2		
Package <code>textcase</code> is now simply a required package . . . . .	2		
Procedures			
<code>\@parse@class@options@society</code> and			
<code>\@parse@class@options@journal</code> and friends . . . . .	2		
			Rag the right even more:
			<code>.8\hsize</code> . Also, assign
			<code>\totalleftmargin</code> . . . . .
			Read in all required packages together . . . . .
			Remove options <code>newabstract</code> and <code>oldabstract</code> . . . . .
			Section numbering via procedures <code>\secnums@rtx</code> and <code>\secnums@arabic</code> . . . . .
			The <code>rmp</code> journal substyle selects <code>groupedaddress</code> by default. . . . .
			The <code>cname</code> substyle@ext is now defined without a dot ( <code>.</code> ), to be compatible with $\LaTeX$ usage (see <code>@clsexextension</code> and <code>@pkgextension</code> ). . . . .
			Use <code>\setup@hook</code> to initialize all. . . . .
	4.1c		General: Document class option <code>longbibliography</code> via <code>\substyle@post</code> . . . . .
	4.1d		<code>\eqnarray@fleqn@fixed</code> : (AO, 511) Compatibility with <code>lineno.sty</code> 's erroneous way of detecting <code>fleqn.clo</code> . . . . .
			General: Definition of <code>\@fnsymbol</code> follows <code>fixltx2e.sty</code> . . . . .
	4.1e		General: (AO, 455) be nice to a list within the abstract . . . . .
	4.1f		<code>\set@footnotewidth</code> : (AO, 515) Hook for setting the font of a footnote . . . . .
			<code>\total@float</code> : (AO, 518) Tally register overflow when document is long . . . . .
			General: (AO, 513) Add class option <code>linenumbers: number</code> the lines a la <code>lineno</code> . . . . .
			(AO, 516) Merged references are separated with a semicolon . . . . .
			(AO, 520) Automatically produce <code>\bibliography</code> command when needed . . . . .
			(AO, 521) Lonely bibliography head . . . . .

(AO, 522) Warn if software is expired . . . . .	2	<code>\newlabel</code> command syntax appropriate to the <code>hyperref</code> package . . . . .	2
(AO, 523) Add class option <code>nomerge</code> , to turn off new <code>natbib</code> 8.3 syntax . . . . .	2	4.1n <code>\clear@document</code> : (AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page . . . . .	18
(AO, 524) Makes no sense if citations are superscript numbers and so are footnotes . . . . .	2	(AO, 569) execute <code>atveryend</code> 's <code>\Call@AfterLastShipout</code> at the proper time . . . . .	18
(AO, 530) <code>\fnsymbol</code> : Failed to import <code>fixltx2e.sty</code> technology. Return to LaTeX core. . . . .	2	<code>\do@check@aux</code> : (AO) Incorporate change to <code>ltnmiscn.dtx v1.li 2000/05/19</code> . . . . .	16
4.1g <code>\doibase</code> : (AO, 532) Both arguments of <code>\href</code> get sanitized . . . . .	68	(AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page . . . . .	15
General: (AO, 525) Remove phantom paragraph above display math that is given in vertical mode . . . . .	2, 22	<code>\l@sections</code> : (AO, 574) protect against <code>lineno.sty</code> , which forces a visit to the output routine, which appears to destroy the value of <code>\@tempdima</code> . . . . .	66
(AO, 538) <code>\MakeTextUppercase</code> inappropriately expands the double backslash . . . . .	2	<code>\set@footnotewidth</code> : (AO, 571) Interface <code>\set@footnotewidth</code> for determining the set width of footnotes . . . . .	28
(AO, 539) Use of double-backslash in argument of <code>\section</code> gives error. The <code>textcase</code> package is involved. . . . .	20	(AO, 571) allow split after last line of footnote . . . . .	28
4.1h General: (AO) Remove expiry code in the release software . . . . .	2	(AO, 572) title block footnotes numbered independently from body footnotes . . . . .	27
4.1i General: (AO, 541) Defer assignment of <code>\cite</code> until after <code>natbib</code> loads . . . . .	2	General: (AO) fine-tune spacing above and below <code>widetext</code> . . . . .	2
4.1j <code>\doibase</code> : (AO, 545) Provide definition for <code>\doi</code> that does hypertext . . . . .	68	(AO, 565) restore 4.0 behavior: invoking class option <code>preprint</code> implies class option <code>preprintnumbers</code> . . . . .	2
<code>\hypertext@enable@ltx</code> : (AO, 545) hypertext capabilities off by default; enable with <code>hypertext</code> . . . . .	68	(AO, 566) restore 4.0 behavior: flush column bottoms . . . . .	2
General: (AO, 545) hypertext capabilities off by default; enable with <code>hypertext</code> . . . . .	2	(AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page . . . . .	2
(AO, 549) Repairing <code>natbib</code> 's <code>\BibitemShut</code> and <code>\bibAnnote</code> . . . . .	2	(AO, 569) execute the after-last-shipout procedures from within the safety of the output routine . . . . .	2
(AO, 552) Repair spacing in <code>\onlincite</code> . . . . .	2	(AO, 571) Interface <code>\set@footnotewidth</code> for determining the set width of footnotes . . . . .	2
4.1k General: (AO, 554) give the			

(AO, 571) class file must set	a stop character. . . . .	2
\splittopskip; fine tune	4.1p	
\skip\footins;	\href: (AO, 582) A patch of	
\footnoterule defined in	hyperref.sty to provide	
terms of \skip\footins . . . . .	backward compatibility to	
(AO, 572) Independent footnote	TeXLive 2007's version 6.75r . . .	69
counter for title block.	General: (AO, 583) Provide	
Abstract footnote counter	interface to ltxgrid	
shared with body. . . . .	\onecolumn@grid@setup and	
(AO, 572) \@makefnstext and	\twocolumn@grid@setup . . . . .	2
\frontmatter@makefnstext	(AO, 584) Per MD, remove	
must be defined harmoniously . . .	trailing space character from	
(AO, 573) arrange to load	each journal abbreviation: it	
lineno after any other	had caused an extraneous	
packages. . . . .	space in the .bbl . . . . .	2
(AO, 575) the default for journal	4.1q	
prstper is longbibliography . . .	General: (AO, 586) When .bbl is	
(AO, 576) In .bst files, remove	pasted into the document,	
support for the annotate field . . .	prevent automatic bibliography	
4.1o	inclusion. . . . .	2
General: (AO, 549) Remove patch	(AO, 588) Only write	
to natbib, which is now at	REVTeX-specific BibTeX .bib	
version 8.31a . . . . .	data if the .bst style is set by	
(AO, 575) Automatically	REVTeX. . . . .	2
incorporate the	4.1r	
(BibTeX-generated) .bbl into	General: (AO, 595) Provide	
an explicit thebibliography . . .	\lovname along with other List	
(AO, 578) accommodate the	of Videos definitions. . . . .	2
possible space character	4.2d	
preceding \BibitemShut. . . . .	\do@check@aux: (PHO) Only	
(AO, 579) Endnote shall	redefine \enddocument in older	
comprise their own BibTeX	versions. . . . .	15
entry type: @FOOTNOTE. . . . .	(PHO) Patch \enddocument at	
(AO, 580) Control .bst at run	runtime in newer versions. . . . .	17
time. . . . .	\document: (PHO) Use L <sup>A</sup> TeX's	
(AO, 580) Provide a document	hook management system, if	
class option to turn off	possible (from 4.2). . . . .	15
production of eprint field in	\rvtx@ifformat@geq: (PHO) Add	
bibliography. . . . .	\rvtx@ifformat@geq (from	
(AO, 581) Handle case: merged	4.2). . . . .	13
references, with first ending in		