

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

Arthur Ogawa [‡]

June 21, 2022

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

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

The `ltxutil` package was commissioned by the American Physical Society and is distributed under the terms of the L^AT_EX Project Public License 1.3c, the same license under which all the portions of L^AT_EX itself are distributed. Please see <http://ctan.tug.org/macros/latex/base/lppl.txt> for details.

To use this document class, you must have a working T_EX installation equipped with L^AT_EX 2_ε and possibly pdf_{te}x and Adobe Acrobat Reader or equivalent.

To install, retrieve the distribution, unpack it into a directory on the target computer, and move the file `ltxutil.sty` into a location in your filesystem where it will be found by L^AT_EX.

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

Contents

1	Processing Instructions	2
1.1	Build Instructions	2
1.2	Change Log	3
1.3	Bill of Materials	3
1.3.1	Primary Source	3
1.3.2	Generated by <code>latex ltxutil.dtx</code>	3
1.3.3	Generated by <code>tex ltxutil.dtx</code>	3
1.3.4	Auxiliary	4
2	Code common to all modules	4
3	The driver module doc	4
3.1	The Preamble	5
3.1.1	Docstrip and info directives	5
3.2	The “Read Me” File	5
3.3	The Document Body	8

^{*}This file has version number 4.2f, last revised 2022/06/05.

[†]Version 4.2f © 2019–2022 American Physical Society

[‡]mailto:arthur_ogawa@sbcglobal.net

4	Using this package	8
4.1	Invoking the package	8
5	Compatibility with L^AT_EX's Required Packages	9
5.1	array	9
5.2	longtable	9
6	Implementation of package	9
6.1	Beginning of the package DOCSTRIP module	10
6.2	Banner and beginning of the kernel 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	19
6.9	Display Math	20
6.10	Footnotes	22
6.11	Floats	27
6.11.1	Usage notes	27
6.11.2	Robustifying fragile commands	28
6.11.3	Preparing for the hyperref package	28
6.11.4	Footnotes within floats, unfloating floats, float font	29
6.11.5	Writing floats out to a file	31
6.12	Counters	35
6.13	Customization of Sections	35
6.14	Patch the tabular and array Environments	39
6.15	Repair other broken parts of L ^A T _E X	59
6.16	Syntax	59
6.17	Auto-indented Contents	59
6.18	Lists	63
6.19	Hypertext capabilities	64
6.20	End of the kernel DOCSTRIP module	66
	Index	67

1 Processing Instructions

The package file `ltxutil.sty` is generated from this file, `ltxutil.dtx`, using the DOCSTRIP facility of L^AT_EX via `tex ltxutil.dtx` (Note: do *not* use L^AT_EX for this task). The typeset documentation that you are now reading is generated from the same file by typesetting it with L^AT_EX or pdftex via `latex ltxutil.dtx` or `pdflatex ltxutil.dtx`.

1.1 Build Instructions

You may bootstrap this suite of files solely from `ltxutil.dtx`. Prepare by installing L^AT_EX 2_ε (and either `tex` or `pdftex`) on your computer, then carry out the following steps:

1. Within an otherwise empty directory, typeset `ltxutil.dtx` with \LaTeX or `pdflatex`; you will obtain the typeset documentation you are now reading, along with the file `README-LTXUTIL`.
 Note: you will have to run \LaTeX , then `makeindex -s gind.ist ltxutil.idx`, then `makeindex -s gglo.ist -o ltxutil.gls ltxutil.glo`, then \LaTeX again in order to obtain a valid index and table of contents.
2. Now typeset `ltxutil.dtx` with \TeX (not \LaTeX), thereby generating the package file `ltxutil.sty`.
3. Install the following files into indicated locations within your TDS-compliant `texmf` tree (you may need root access):
 - `$TEXMF/tex/latex/revtex/ltxutil.sty`
 - `$TEXMF/source/latex/revtex/ltxutil.dtx`
 - `$TEXMF/doc/latex/revtex/ltxutil.pdf`
 where `$TEXMF/` stands for `texmf-local/`, or some other `texmf` tree in your installation.
4. Run `mktexlsr` on `$TEXMF/` (you may need root access).
5. Build and installation are now complete; now put a `\usepackage{ltxutil}` in your document preamble!

1.2 Change Log

1.3 Bill of Materials

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

1.3.1 Primary Source

One single file generates all.

```
%ltxutil.dtx
%
```

1.3.2 Generated by `latex ltxutil.dtx`

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

```
%README-LTXUTIL ltxutil.pdf
%
```

1.3.3 Generated by `tex ltxutil.dtx`

Typesetting this file with \TeX generates the package file.

```
%ltxutil.sty
%
```

1.3.4 Auxiliary

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

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

2 Code common to all modules

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

Now we declare that these files can only be used with L^AT_EX 2_ε. An appropriate message is displayed if a different T_EX format is used.

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

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

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

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

```
18 %<*doc|package|kernel>
19 %<version>
20 [2022/06/05 4.2f utilities package (portions licensed from W. E. Baxter web at superscript.
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 README-LTXUTIL 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 %\expandafter\ifx\cename package@font\endcename\@undefined\else
33 % \expandafter\RequirePackage\expandafter{\cename package@font\endcename}%
34 %\fi
35 \CodelineIndex\EnableCrossrefs % makeindex -s gind.ist ltxutil
36 \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.

```
37 \setcounter{StandardModuleDepth}{1}
```

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

```
38 \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.

```
39 \begin{filecontents*}{README-LTXUTIL}
40 \title{%
41   A \LaTeX\ Package of utility macros%
42   \thanks{%
43     This file has version number \fileversion,
44     last revised \filedate.%
45   }%
46   \thanks{%
47     Version \fileversion\ \copyright\ 2019--2022 American Physical Society
48   }%
49 }
```

```

49 }%
50 \author{%
51   Arthur Ogawa%
52   \thanks{\texttt{mailto:arthur\_ogawa at sbcglobal.net}}}%
53 }%
54 %\iffalse
55 % For version number and date,
56 % search on "\fileversion" in the .dtx file,
57 % or see the end of the README-LTXUTIL file.
58 %\fi
59 \maketitle
60
61 This file embodies the \classname{ltxutil} package,
62 the implementation and its user documentation.
63
64 The distribution point for this work is
65 \url{journals.aps.org/revtex},
66 which contains prebuilt runtime files, documentation, and full source,
67 ready to add to a TDS-compliant \TeX\ installation.
68
69 The \classname{ltxutil} package was commissioned by the American Physical Society
70 and is distributed under the terms of the \LaTeX\ Project Public License 1.3c,
71 the same license under which all the portions of \LaTeX\ itself are distributed.
72 Please see \url{http://ctan.tug.org/macros/latex/base/lppl.txt} for details.
73
74 To use this document class, you must have a working
75 \TeX\ installation equipped with \LaTeXe\
76 and possibly pdftex and Adobe Acrobat Reader or equivalent.
77
78 To install, retrieve the distribution,
79 unpack it into a directory on the target computer,
80 and move the file \file{ltxutil.sty}
81 into a location in your filesystem where it will be found by \LaTeX.
82
83 To use, read the user documentation \file{ltxutil.pdf}.
84
85 \tableofcontents
86
87 \section{Processing Instructions}
88
89 The package file \file{ltxutil.sty}
90 is generated from this file, \file{ltxutil.dtx},
91 using the {\sc docstrip} facility of \LaTeX
92 via |tex ltxutil.dtx| (Note: do \emph{not} use \LaTeX\ for this task).
93 The typeset documentation that you are now reading is generated from
94 the same file by typesetting it with \LaTeX\ or pdftex
95 via |latex ltxutil.dtx| or |pdflatex ltxutil.dtx|.
96
97 \subsection{Build Instructions}
98
99 You may bootstrap this suite of files solely from \file{ltxutil.dtx}.
100 Prepare by installing \LaTeXe\ (and either tex or pdftex) on your computer,
101 then carry out the following steps:
102 \begin{enumerate}

```

```

103 \item
104 Within an otherwise empty directory,
105 typeset \file{ltxutil.dtx} with \LaTeX or pdflatex;
106 you will obtain the typeset documentation you are now reading,
107 along with the file \file{README-LTXUTIL}.
108
109 Note: you will have to run \LaTeX, then
110 \file{makeindex} \texttt{-s gind.ist ltxutil.idx}, then
111 \file{makeindex} \texttt{-s gglo.ist -o ltxutil.gls ltxutil.glo}, then
112 \LaTeX again in order to obtain a valid index and table of contents.
113 \item
114 Now typeset \file{ltxutil.dtx} with \TeX (not \LaTeX),
115 thereby generating the package file \file{ltxutil.sty}.
116 \item
117 Install the following files into indicated locations within your
118 TDS-compliant \texttt{texmf} tree (you may need root access):
119 \begin{itemize}
120 \item
121 \file{$TEXMF/}\file{tex/}\file{latex/}\file{revtex/}\classname{ltxutil.sty}
122 \item
123 \file{$TEXMF/}\file{source/}\file{latex/}\file{revtex/}\classname{ltxutil.dtx}
124 \item
125 \file{$TEXMF/}\file{doc/}\file{latex/}\file{revtex/}\classname{ltxutil.pdf}
126 \end{itemize}
127 where \file{$TEXMF/} stands for \file{texmf-local/}, or some other \texttt{texmf} tree
128 in your installation.
129 \item
130 Run \texttt{mktexlsr} on \file{$TEXMF/} (you may need root access).
131 \item
132 Build and installation are now complete;
133 now put a \code{\usepackage\texttt{\{ltxutil\}}} in your document preamble!
134 \end{enumerate}
135
136 \subsection{Change Log}
137 \changes{4.0b}{1999/06/20}{A0: Fixed spurious \texttt{CR} and (return) characters in output}
138 \changes{4.0b}{1999/06/20}{A0: Removed superfluous \code{\cs{def}}s, changed to using \code{\cs{floats@sw}}
139 \changes{4.0b}{1999/06/20}{only execute if there really were floats of the given type}
140 \changes{4.0b}{1999/06/20}{Support the hack with \code{\cs{prepdef}}, and delay until \code{\cs{AtBeginDoc}}
141 \changes{4.0c}{1999/11/13}{(A0, 110) Install hooks for endfloats processing}
142 \changes{4.0c}{1999/11/13}{(A0, 116) Hyperref compatibility}
143 \changes{4.0c}{1999/11/13}{(A0, 130) Interference from array package}
144 \changes{4.0c}{1999/11/13}{*-form mandates pagebreak at each float; only print section head}
145 \changes{4.0d}{2000/04/10}{(A0, 127) Floats placed [h] to allow page breaks}
146 \changes{4.0d}{2000/04/10}{(A0, 174) kernel fix}
147 \changes{4.0d}{2000/05/19}{(A0, 224) Hyperref compatibility.}
148 \changes{4.0d}{2000/05/23}{Allow things to break over pages by setting array@default.}
149 \changes{4.0e}{2000/11/16}{(A0, 221) Remove samepage command from @xfloat@prep: If the float
150 \changes{4.0f}{2001/07/13}{(A0, 404) Hyperref compatibility}
151 \changes{4.1a}{2008/01/19}{(A0, 459) do not assume \code{\cs{class@name}} is defined}%
152 \changes{4.1a}{2008/01/19}{(A0, 461) Change the csname from \code{\cs{@dotsep}} to \code{\cs{ltxu@dotsep}}
153 \changes{4.1a}{2008/01/19}{(A0, 475) I had not properly reproduced the LaTeX macro \code{\cs{eqnarray}}
154 \changes{4.1a}{2008/01/19}{(A0, 479) Per: Dylan Thurston<dpt at math.harvard.edu>}%
155 \changes{4.1a}{2008/06/30}{(A0) Make \code{\cs{addtocontents}} a \code{\cs{long}} \code{\cs{def}}; gobble up \code{\cs{}}
156 \changes{4.1a}{2008/06/30}{(A0) Remove code that avoided changes to \code{\cs{@footnotemark}}}%

```

```

157 \changes{4.1a}{2008/06/30}{(A0, 438) Complete rewrite of footnote macros.}
158 \changes{4.1a}{2008/07/07}{\cs{xfloat@prep} calls \cs{ltx@footnote@pop} to restore the origi
159 \changes{4.1a}{2008/08/12}{\cs{class@documenthook} is the last \cs{AtBeginDocument} token no
160 \changes{4.1a}{2008/08/12}{Class extension mechanism \cs{pushfilename@ltx} and \cs{p@pfile
161 \changes{4.1a}{2008/08/12}{Class extension mechanism \cs{class@extension}, \cs{class@extensi
162 \changes{4.1a}{2008/08/12}{Get rid of \cs{set@typesize@hook} \cs{set@pica@hook} and the \cs{
163 \changes{4.1b}{2008/08/12}{(A0, 487) Support for video figures and the \cs{setfloatlink} com
164 \changes{4.1b}{2008/08/12}{(A0, 505) try to accommodate \classname{colortbl}.}
165 \changes{4.1b}{2008/08/12}{Acquire \classname{hyperref} savoirre}
166 \changes{4.1b}{2008/08/12}{Default assignment of \cs{float@sw} now, not at \cs{AtBeginDocume
167 \changes{4.1b}{2008/08/12}{If class option \classoption{lengthcheck} is in effect, log the h
168 \changes{4.1b}{2008/08/12}{No need to protect against undefined \cs{float@sw}}
169 \changes{4.1b}{2008/08/12}{Patch the array package even later: after all package patches go
170 \changes{4.1b}{2008/08/12}{Refine toc processing: provide default.}%
171 \changes{4.1b}{2008/08/12}{Tally and log the height of a float class}
172 \changes{4.1d}{2009/03/27}{(A0, 511) Compatibility with lineno.sty's erroneous way of detect
173 \changes{4.1f}{2009/07/07}{(A0, 515) Hook for setting the font of a footnote}
174 \changes{4.1f}{2009/07/10}{(A0, 518) Tally register overflow when locument is long}
175 \changes{4.1g}{2009/10/06}{(A0, 532) Both arguments of \cs{href} get sanitized}%
176 \changes{4.1g}{2009/10/07}{(A0, 525) Remove phantom paragraph above display math that is giv
177 \changes{4.1g}{2009/10/07}{(A0, 539) Use of double-backslash in argument of \cs{section} giv
178 \changes{4.1n}{2009/12/05}{(A0, 569) Use of \classname{hyperref} interferes with column bala
179 \changes{4.1n}{2009/12/06}{(A0) Incorporate change to ltmiscen.dtx v1.1i 2000/05/19}%
180 \changes{4.1n}{2009/12/09}{(A0, 569) execute \classname{atveryend}'s \cs{Call@AfterLastShipo
181 \changes{4.1n}{2009/12/13}{(A0, 574) protect against \classname{lineno.sty}, which forces a
182 \changes{4.1n}{2010/01/02}{(A0, 571) Interface \cs{set@footnotewidth} for determining the se
183 \changes{4.1n}{2010/01/02}{(A0, 571) allow split after last line of footnote}%
184 \changes{4.1n}{2010/01/06}{(A0, 572) title block footnotes numbered independently from body
185 \changes{4.1p}{2010/02/24}{(A0, 582) A patch of \classname{hyperref.sty} to provide backward
186 \changes{4.2a}{2017/11/21}{(MD) Use updated best practice to use https and doi.org}%
187 \changes{4.2a}{2018/12/12}{(MD) Updated name of README file and use standard fonts when type
188 \changes{4.2d}{2020/09/19}{(PHO) Adapt \cs{document} and \cs{enddocument} hooks to the 2020-
189
190 \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.

```

191 \begin{document}%
192 \expandafter\DocInput\expandafter{\jobname.dtx}%
193 \end{document}
194 %</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

```
195 %<*package>
196 \def\package@name{ltxutil}%
197 \expandafter\PackageInfo\expandafter{\package@name}{%
198   Utility macros for \protect\LaTeXe,
199   by A. Ogawa (arthur_ogawa at sbcglobal.net)%
200 }%
201 %</package>
```

6.2 Banner and beginning of the kernel DOCSTRIP module

```
202 %<*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 203 \def\class@err#1{\ClassError{\class@name}{#1}\@eha}%
204 \def\class@warn#1{\ClassWarningNoLine{\class@name}{#1}}%
205 \def\class@info#1{\ClassInfo{\class@name}{#1}}%
206 \def\obsolete@command#1{%
207   \class@warn@end{Command \string#1\space is obsolete.^^JPlease remove from your document}%
208   \global\let#1\@empty
209   #1%
210 }%
211 \def\replace@command#1#2{%
212   \class@warn@end{Command \string#1\space is obsolete;^^JUse \string#2\space instead}%
213   \global\let#1#2%
214   #1%
215 }%
216 \def\replace@environment#1#2{%
217   \class@warn@end{Environment #1 is obsolete;^^JUse #2 instead}%
218   \glet@environment{#1}{#2}%
219   \@nameuse{#1}%
220 }%
221 \def\incompatible@package#1{%
222   \ifpackageloaded{#1}{%
223     \def\@tempa{I cannot continue. You must remove the \string\usepackage\ statement that causes
224     \ClassError{\class@name}{The #1 package cannot be used with \class@name}%
225     \@tempa\stop
226   }{%
227     \class@info{#1 was not loaded (OK!)}%
228   }%
229 }%
230 \def\class@warn@end#1{%
231   \gapdef\class@enddocumenthook{\class@warn{#1}}%
232 }%
```

Give `\class@name` a meaning if it does not already have one.

```
233 \ifx\undefined\class@name
234 \def\class@name{ltxutil}%
235 \class@warn{You should define the class name before reading in this package. Using default}
236 \fi
```

6.4 New Tools

```

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

\dimen@iii
238 \dimendef\dimen@iii\thr@@

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

\four Analogous to \@ne, \tw@, and \thr@@.
240 \chardef\four=4\relax
241 \chardef\cat@letter=11\relax
242 \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.
243 \def\let@environment#1#2{%
244 \expandafter\let
245 \csname#1\expandafter\endcsname\csname#2\endcsname
246 \expandafter\let
247 \csname end#1\expandafter\endcsname\csname end#2\endcsname
248 }%
249 \def\glet@environment#1#2{%
250 \global\expandafter\let
251 \csname#1\expandafter\endcsname\csname#2\endcsname
252 \global\expandafter\let
253 \csname end#1\expandafter\endcsname\csname end#2\endcsname
254 }%

\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.
255 \newcommand\tracingplain{%
256 \tracingonline\z@\tracingcommands\z@\tracingstats\z@
257 \tracingpages\z@\tracingoutput\z@\tracinglostchars\@ne
258 \tracingmacros\z@\tracingparagraphs\z@\tracingrestores\z@
259 \showboxbreadth5\showboxdepth3\relax %\errorstopmode
260 }%
261 \newcommand\traceoutput{%
262 \appdef\@resetactivechars{\showoutput}%
263 }%

\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.
264 \newcommand\say[1]{\typeout{<\noexpand#1=\meaning#1>}}%
265 \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.
266 \def\fullinterlineskip{\prevdepth\z@}%

```

```

\count@i
\count@ii 267 \countdef\count@i\@ne
           268 \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.

```

269 \long\def\prepdef#1#2{%
270   \@ifxundefined#1{\toks@{}}{\toks@\expandafter{#1}}%
271   \toks@ii{#2}%
272   \edef#1{\the\toks@ii\the\toks@}%
273 }%
274 \long\def\appdef#1#2{%
275   \@ifxundefined#1{\toks@{}}{\toks@\expandafter{#1}}%
276   \toks@ii{#2}%
277   \edef#1{\the\toks@\the\toks@ii}%
278 }%
279 \long\def\gappdef#1#2{%
280   \@ifxundefined#1{\toks@{}}{\toks@\expandafter{#1}}%
281   \toks@ii{#2}%
282   \global\edef#1{\the\toks@\the\toks@ii}%
283 }%
284 \long\def\appdef@val#1#2{%
285   \appdef#1{{#2}}%
286 }%
287 \long\def\appdef@e#1#2{%
288   \expandafter\appdef
289   \expandafter#1%
290   \expandafter{#2}%
291 }%
292 \long\def\appdef@eval#1#2{%
293   \expandafter\appdef@val
294   \expandafter#1%
295   \expandafter{#2}%
296 }%
297 \toksdef\toks@ii=\tw@

```

`\@ifxundefined` Certain utility procedures use `\@ifxundefined`, which is defined here in terms of `\@ifnotrelax`. 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.

```

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

```

`\rvtx@ifformat@geq` Some changes in the L^AT_EX 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^AT_EX 2_ε kernel is greater or equal to the argument (the argument should be in the format yyyy-mm-dd).

```

303 \if\IfFormatAtLeastTF\undefined
304   \def\rvtx@ifformat@geq{\@ifl@t@r\fmtversion}%
305 \else
306   \let\rvtx@ifformat@geq\IfFormatAtLeastTF
307 \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.

```

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

```

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

```

318 \def\@booleantrue#1{\let#1\true@sw}%
319 \def\@booleanfalse#1{\let#1\false@sw}%

```

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

```

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

```

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

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

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

```
338 \long\def\loopuntil#1{#1}{\loopuntil{#1}}}%
339 \long\def\loopwhile#1{#1{\loopwhile{#1}}{}}%
```

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

```
340 \def\@provide#1{%
341   \@ifx{\undefined#1}{\true@sw}{\@ifx{\relax#1}{\true@sw}{\false@sw}}}%
342   {\def#1}{\def\j@nk}%
343 }%
```

6.6 Begin Document Structure

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

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

```
344 \rvtx@ifformat@geq{2020/10/01}%
345   {%
346     \AddToHook{begindocument/before}{\document@inithook}%
347   }{%
348     \prepdef\document{%
349       \endgroup
350       \document@inithook
351       \true@sw}%
352   }%
353 }
```

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

```
354 \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`.

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

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

This approach, however, will only be done for older versions of the L^AT_EX kernel:

```
363 \rvtx@ifformat@geq{2020/10/01}{%
364   % <definitions for newer LaTeX later>
365 }{%
366   % <definitions for older LaTeX>
367 \def\enddocument{%
```

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

```
368 \let\AtEndDocument\@firstofone
369 \@enddocumenthook
370 \@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`.

```
371 \clear@document
```

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

```
372 \check@aux
373 \deadcycles\z@
374 \@@end
375 }%
376 \def\check@aux{\do@check@aux}%
377 \def\do@check@aux{%
378   \if@sw\if@files\fi{%
379     \immediate\closeout\@mainaux
```

```

380 \let\@setckpt\@gobbletwo
381 \let\@newl@bel\@testdef
382 \@tempswafalse
383 \makeatletter
384 \input\jobname.aux\relax
385 }{}%
386 \@dofilelist
387 \@ifdim{\font@submax >\fontsubfuzz\relax}{%
388 \@font@warning{%
389   Size substitutions with differences\MessageBreak
390   up to \font@submax\space have occurred.\@gobbletwo
391 }%
392 }{}%
393 \@defaultsubs
394 \@refundefined
395 \@if@sw@if@files\fi{%
396 \@ifx{\@multiplelabels\relax}{%
397 \@if@sw@if@temp\sw\fi{%
398 \@latex@warning@no@line{%
399   Label(s) may have changed.
400   Rerun to get cross-references right%
401 }%
402 }{}%
403 }{}%
404 \@multiplelabels
405 }%
406 }{}%
407 }%
408 }

```

`\rvtx@enddocument@patch` For newer L^AT_EX we'll try to be a bit more future-proof (no miracle though). The code for `\enddocument` (in pre-2020-10-01 L^AT_EX) 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:

```

409 \rvtx@ifformat@geq{2020/10/01}{%

```



```

410 \AddToHook{enddocument}{\rvtx@enddocument@patch{}}%
411 }{}

```

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:

```

412 \protected\long\def\rvtx@enddocument@patch#1#2\@checkend#3{%
413   \begingroup
414   \edef\x{\detokenize{#3}}%
415   \edef\y{\detokenize{document}}%
416   \expandafter\endgroup
417   \ifx\x\y
418     \expandafter\rvtx@enddocument@patch@end
419   \else
420     \expandafter\rvtx@enddocument@patch@more
421   \fi
422   {#1#2}{#3}}
423 \def\rvtx@enddocument@patch@more#1#2{%
424   \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`:

```

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

```

Then execute the code consumed in the previous step:

```

427   #1%
428   \@checkend{#2}%

```

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

```

429   \clear@document
430   \check@aux}
431 \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.

```

432 \def\clear@document{%
433   \clearpage
434   \do@output@cc1v{%
435     \Call@AfterLastShipout
436   }%
437 }%

```

```

438 \appdef\class@documenthook{%
439 \providecommand\Call@AfterLastShipout{%
440 }%

```

6.7 Class Extensions

The L^AT_EX procedure `\onefilewithoptions` is the vehicle for reading in a L^AT_EX class or package. The APS RevTeX class implements the use of what are called “substyles”, actually extensions to the class itself. Any document class can do likewise.

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

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

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

```

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

```

```

469 #1%
470 \let#1\@undefined
471 \expandafter\@p@pfilename@ltx\@currnamestack@ltx\@nil
472 \@reset@options
473 }%

```

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

```

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

```

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

6.8 Type Tools

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

```

491 \def\flushing{%
492   \let\\\@normalcr
493   \leftskip\z@skip
494   \rightskip\z@skip
495   \@rightskip\z@skip
496   \parfillskip\@flushglue
497 }%

```

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

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

```
%\documentclass{article}%

```

```
%\usepackage[overload]{textcase}
%\begin{document}
%\centering
%\section{\MakeTextUppercase{Section\\title}}
%Text
%\end{document}
%
```

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

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

6.9 Display Math

`\eqnarray@LaTeX` Team L^AT_EX has stated they will never repair Leslie's broken definition of `\eqnarray@fleqn@fixed 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.

```
499 \def\rvtx@tmpa#1{%
500 \def\eqnarray@LaTeX{%
501   \stepcounter{equation}%
502   \def\@currentlabel{\p@equation\theequation}%
503   #1% \def\@currentcounter{equation} on newer LaTeX
504   \global\@eqnswtrue
505   \m@th
506   \global\@eqcnt\z@
507   \tabskip\@centering
508   \let\\\@eqnocr
509   $$\everycr{}\halign to\displaywidth\bgroup
510     \hskip\@centering$\displaystyle\tabskip\z@skip{####}$\@eqnsele
511     &\global\@eqcnt\@ne\hskip \tw@\arraycolsep \hfil${####}$\hfil
512     &\global\@eqcnt\tw@ \hskip \tw@\arraycolsep
513     $\displaystyle{####}$\hfil\tabskip\@centering
514     &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss####\egroup
515     \tabskip\z@skip
516   \cr
517 }%
518 \long\def\eqnarray@fleqn@fixed{%
519   \stepcounter{equation}\def\@currentlabel{\p@equation\theequation}%
520   #1% \def\@currentcounter{equation} on newer LaTeX
521   \global\@eqnswtrue\m@th\global\@eqcnt\z@
522   \tabskip\ltx@mathindent
523   \let\=\@eqnocr
524   \setlength\abovedisplayskip{\topsep}%
525   \ifvmode\addtolength\abovedisplayskip{\partopsep}\fi
526   \addtolength\abovedisplayskip{\parskip}%
527   \setlength\belowdisplayskip{\abovedisplayskip}%
528   \setlength\belowdisplayshortskip{\abovedisplayskip}%
529   \setlength\abovedisplayshortskip{\abovedisplayskip}%
530   $$$
531   \everycr{}%
532   \halign to\linewidth\bgroup
```

```

533 \hskip\@centering$\displaystyle\tabskip\z@skip{####}$\@eqnse1
534 &\global\@eqcnt\@ne
535 \hskip\tw\@eqncolsep
536 \hfil${}\####{}}$\hfil
537 &\global\@eqcnt\tw@
538 \hskip\tw\@eqncolsep
539 $\displaystyle{####}$\hfil\tabskip\@centering
540 &\global\@eqcnt\thr@@\hb@xt@\z@\bgroup\hss####\egroup
541 \tabskip\z@skip
542 \cr
543 }%
544 }
545 \rvtx@tmpa{}% older LaTeX
546 \@ifx{\eqnarray\eqnarray@LaTeX}{\@firstofone}
547 {%
548 \rvtx@tmpa{\def\@currentcounter{equation}}% newer LaTeX
549 \@ifx{\eqnarray\eqnarray@LaTeX}{\@firstofone}
550 {\@gobble}
551 }
552 {%
553 \class@info{Repairing broken LaTeX eqnarray}%
554 \let\eqnarray\eqnarray@fleqn@fixed
555 \newlength\eqncolsep
556 \setlength\eqncolsep\z@
557 \let\eqnarray@LaTeX\relax
558 \let\eqnarray@fleqn@fixed\relax
559 }%

```

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

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

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

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

```

560 \def\ltx@mathindent{\@centering}%
561 \def\set@eqnarray@skips{}%

```

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

```
562 \def\prep@math{%
563   \@ifvmode{\everypar{\setbox\z@\lastbox}}{\}%
564 }%
565 \def\prep@math@patch{%
566   \prepdef\equation{\prep@math}%
567   \prepdef\eqnarray{\prep@math}%
568 }%
```

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.
\footnotetext 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 `\footnotetext` procedures, achieving a cleaner separation of syntax and semantics. Note that the `\@footnotetext` 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@@footnotetext`. We anticipate this, and do our context switching on `\H@@footnotetext`.

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\@footnotetext\@mpfootnotetext
%\let\@makefnmark\@mpmakefnmark
%\c@mpfootnote\z@
```

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

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

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

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

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

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

```

569 \def\footnote{\@ifnextchar[\ltx@xfootnote\ltx@yfootnote]}%
570 \def\ltx@xfootnote[#1]{%
571   \ltx@def@footproc\ltx@footmark[#1]%
572   \expandafter\ltx@foottext\expandafter{\the\csname c@\@mpfn\endcsname}%
573 }%
574 \def\ltx@yfootnote{%
575   \ltx@stp@footproc\ltx@footmark
576   \expandafter\ltx@foottext\expandafter{\the\csname c@\@mpfn\endcsname}%
577 }%

```

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.

```

578 \def\footnotemark{\@ifnextchar[\ltx@xfootmark\ltx@yfootmark]}%
579 \def\ltx@xfootmark{\ltx@def@footproc\ltx@footmark}%
580 \def\ltx@yfootmark{\ltx@stp@footproc\ltx@footmark}%
581 \def\ltx@footmark#1{%
582   \leavevmode
583   \ifhmode\edef\@xsf{\the\spacefactor}\nobreak\fi
584   \begingroup
585     \expandafter\ltx@make@current@footnote\expandafter{\@mpfn}{#1}%
586     \expandafter\@argswap@val\expandafter{\Hy@footnote@currentHref}{\hyper@linkstart {link}}%
587     \@makefnmark
588     \hyper@linkend
589   \endgroup
590   \ifhmode\spacefactor\@xsf\fi
591   \relax
592 }%

```

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.

```

593 \def\footnotetext{\@ifnextchar[\ltx@xfoottext\ltx@yfoottext]}%
594 \def\ltx@xfoottext{\ltx@def@footproc\ltx@foottext}%
595 \def\ltx@yfoottext{\ltx@stp@footproc\ltx@foottext}%
596 \long\def\ltx@foottext#1#2{%
597   \begingroup
598     \expandafter\ltx@make@current@footnote\expandafter{\@mpfn}{#1}%
599     \@footnotetext{#2}%

```

```

600 \endgroup
601 }%

```

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.

```

602 \def\ltx@def@footproc#1[#2]{%
603 \begingroup
604   \csname c@\mpfn\endcsname #2\relax
605   \unrestored@protected@xdef\thefnmark{\thempfn}%
606 \expandafter\endgroup
607 \expandafter#1%
608 \expandafter{\the\csname c@\mpfn\endcsname}%
609 }%
610 \def\ltx@stp@footproc#1{%
611 \expandafter\stepcounter\expandafter{\mpfn}%
612 \protected@xdef\thefnmark{\thempfn}%
613 \expandafter#1%
614 \expandafter{\the\csname c@\mpfn\endcsname}%
615 }%

```

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 gentler and without hypertext functionality.

```

616 \appdef\class@documenthook{%
617 \let\footnote@latex\footnote
618 \@ifpackageloaded{hyperref}{}%
619 \let\H@@footnotetext\@footnotetext
620 \def\@footnotetext{\H@@footnotetext}%
621 \let\H@@mpfootnotetext\@mpfootnotetext
622 \def\@mpfootnotetext{\H@@mpfootnotetext}%
623 }%
624 }%

```

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

```

625 \def\ltx@make@current@footnote#1#2{%
626   \csname c@#1\endcsname#2\relax
627   \protected@edef\Hy@footnote@currentHref{\@currentHref-#1.\csname the#1\endcsname}%
628 }%
629 \def\thempfootnote@latex{\itshape \@alph \c@mpfootnote }%
630 \def\ltx@thempfootnote{\@alph \c@mpfootnote}%
631 \@ifx{\thempfootnote\thempfootnote@latex}{%
632 \class@info{Repairing hyperref-unfriendly LaTeX definition of \string\mpfootnote}%
633 \let\thempfootnote\ltx@thempfootnote
634 }{}%

```

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

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

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

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

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

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

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

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

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

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

<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> <p>Difference from L^AT_EX: here we do not set <code>\floatingpenalty</code> to infinity. Doing this must date back to a time when L^AT_EX could not accomodate split insertions (footnotes). I cannot think of any other reason to do have done this. At any rate, with the <code>ltxgrid</code> package, split insertions are properly taken care of, so we allow it.</p>
---	---

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.

```

635 \def\@makefnmark{%
636   \hbox{%
637     \@textsuperscript{%
638       \normalfont\itshape\@thefnmark
639     }%

```

```

640 }%
641 }%

642 \long\def\@footnotetext{%
643   \insert\footins\bgroup
644   \make@footnotetext
645 }%

646 \long\def\@mpfootnotetext{%
647   \minipagefootnote@pick
648   \make@footnotetext
649 }%

```

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@makefnmark` instead of `\@makefnmark`.

Dependency note: the `\@makefnmark` 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\@makefnmark[1]{%
%  \parindent 1em\noindent
%  \hb@xt@1.8em{\hss\@makefnmark}%
%  #1%
%}%

%\def\@makefnmark{%
%  \hbox{\@textsuperscript{\normalfont\@thefnmark}}}%
%}%
%

```

```

650 \long\def\make@footnotetext#1{%
651   \set@footnotefont

```

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

```

652   \set@footnotewidth
653   \@parboxrestore
654   \protected@edef\@currentlabel{%

```

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

```

655   \csname p@\@mpfn\endcsname\@thefnmark
656 }%
657 \color@begingroup
658   \@makefnmark{%
659     \rule{z@\footnotesep}{\ignorespaces#1}

```

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

```

660   \@finalstrut\strutbox\vadjust{\vskip\z@skip}%
661 }%
662 \color@endgroup

```

```

663 \minipagefootnote@drop
664 }%

```

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

```

665 \def\set@footnotefont{%
666   \reset@font\footnotesize
667   \interlinepenalty\interfootnotelinepenalty
668   \splittopskip\footnotesep
669   \splitmaxdepth\dp\strutbox
670 }%

```

`\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.

```

671 \def\set@footnotewidth{\set@footnotewidth@one}%

```

6.11 Floats

6.11.1 Usage notes

We extend the L^AT_EX kernel for three purposes:

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

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

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

```

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

```

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

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

```

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

```

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

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

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

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

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

6.11.2 Robustifying fragile commands

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

```
\addtocontents
\robustify@contents 672 \def\robustify@contents{%
673   \let \label \gobble
674   \let \index \gobble
675   \let \glossary \gobble
676   \let \footnote \gobble
677   \def\({\string\}%
678   \def\){\string\}%
679   \def\\{\string\\}%
680 }%
681 \long\def\addtocontents#1#2{%
682   \protected@write\@auxout{\robustify@contents}{\string \@writefile {#1}{#2}}%
683 }%
```

6.11.3 Preparing for the hyperref package

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

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

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

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

```
684 \def\addcontentsline#1#2#3{%
685   \addtocontents{#1}{%
686     \protect\contentsline{#2}{#3}{\thepage}{}%

```

```

687 }%
688 }%
689 \def\label#1{%
690   \bsphack
691   \protected@write\@auxout{}\@auxout{%
692     \string\newlabel{#1}{\@currentlabel}{\thepage}{-}{-}}%
693   }%
694   \esphack
695 }%
696 \def\ltx@contentsline#1{%
697   \expandafter\ifnotrelax\csname l@#1\endcsname{}\@auxout{%
698     \expandafter\let\csname l@#1\endcsname\gobbletwo
699   }%
700   \contentsline@latex{#1}%
701 }%
702 \appdef\document@inithook{%
703   \let\contentsline@latex\contentsline
704   \let\contentsline\ltx@contentsline
705 }%

```

6.11.4 Footnotes within floats, unfloating floats, float font

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

```

706 \appdef\class@documenthook{%
707   \prepdef\caption{\minipagefootnote@here}%
708 }%

```

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

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

```

%\def\caption{%
%   \H@refstepcounter\@captype
%   \hyper@makecurrent{\@captype}%
%   \@dblarg{\H@caption\@captype}%
%}%
%\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`
`\minipagefootnote@drop`

```

709 \def\minipagefootnote@init{%
710 \setbox\@mpfootins\box\voidb@x
711 }%
712 \def\minipagefootnote@pick{%
713 \global\setbox\@mpfootins\vbox\bgroup
714 \unvbox\@mpfootins
715 }%
716 \def\minipagefootnote@drop{%
717 \egroup
718 }%
719 \def\minipagefootnote@here{%
720 \par
721 \@ifvoid\@mpfootins{}{%
722 \vskip\skip\@mpfootins
723 \fullinterlineskip
724 \@ifinner{%
725 \vtop{\unvcopy\@mpfootins}%
726 {\setbox\z@\lastbox}%
727 }{}%
728 \unvbox\@mpfootins
729 }%
730 }%
731 \def\minipagefootnote@foot{%
732 \@ifvoid\@mpfootins{}{%
733 \insert\footins\bgroup\unvbox\@mpfootins\egroup
734 }%
735 }%
736 \def\endminipage{%
737 \par
738 \unskip
739 \minipagefootnote@here
740 \@minipagefalse %% added 24 May 89
741 \color@endgroup
742 \egroup
743 \expandafter\@iiiparbox\@mpargs{\unvbox\@tempboxa}%
744 }%

```

`\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.

```

745 \@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.

```

746 \let\@xfloat@LaTeX\@xfloat
747 \def\@xfloat#1[#2]{%
748 \@xfloat@prep
749 \@nameuse{fp@proc@#2}%
750 \floats@sw{\@xfloat@LaTeX{#1}[#2]}\@xfloat@anchored{#1}[]}%
751 }%

```

```

752 \def\@xfloat@prep{%
753   \ltx@footnote@pop
754   \def\@mpfn{mpfootnote}%
755   \def\thempfn{\thempfootnote}%
756   \c@mpfootnote\z@
757   \let\H@@footnotetext\H@@mpfootnotetext
758 }%
759 \let\ltx@footnote@pop\@empty
760 \def\@xfloat@anchored#1[#2]{%
761   \def\@capttype{#1}%
762   \begin@float@pagebreak
763   \let\end@float\end@float@anchored
764   \let\end@dblfloat\end@float@anchored
765       \hsize\columnwidth
766       \@parboxrestore
767       \@floatboxreset
768   \minipagefootnote@init
769 }%
770 \def\end@float@anchored{%
771   \minipagefootnote@here
772   \par\vskip\z@skip
773   \par
774   \end@float@pagebreak
775 }%
776 \def\begin@float@pagebreak{\par\addvspace\intextsep}%
777 \def\end@float@pagebreak{\par\addvspace\intextsep}%
778 \def\@mpmakefnmark#1{%
779   \parindent=1em
780   \noindent
781   \hb@xt@1em{\hss\@makefnmark}%
782   #1%
783 }%

```

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.

```

784 \def\do@if@floats#1#2{%
785   \floats@sw{ }{%

```

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

```

786   \expandafter\newwrite
787       \csname#1write\endcsname
788   \expandafter\def
789       \csname#1@stream\endcsname{\jobname#2}%
790   \expandafter\immediate
791   \expandafter\openout
792       \csname#1write\endcsname
793       \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`.

```

794 \ifxundefined\@float@LaTeX{%
795   \let\@float@LaTeX\@float
796   \let\@dblfloat@LaTeX\@dblfloat
797   \let\@float\write@float
798   \let\@dblfloat\write@floats
799 }{}%
800 \let@environment{#1@float}{#1}%
801 \let@environment{#1@floats}{#1*}%
802 \ifxundefined@cs{#1@write}{}%
803   \let@environment{#1}{#1@write}%
804 }%
805 }%
806 }%
```

`\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.

```

807 \def\triggerpar{\leavevmode\@par}%
808 \def\oneapage{\def\begin@float@pagebreak{\newpage}\def\end@float@pagebreak{\newpage}}%
809 \def\print@float#1#2{%
810   \lengthcheck@sw{%
811     \total@float{#1}%
812   }{}%
813   \ifxundefined@cs{#1@write}{}%
814     \begingroup
815       \booleanfalse\floats@sw
816       #2%
817       \raggedbottom
818       \def\array@default{v}% floats must
819       \let\@float\@float@LaTeX
820       \let\@dblfloat\@dblfloat@LaTeX
821       \let\trigger@float@par\triggerpar
822       \let@environment{#1}{#1@float}%
823       \let@environment{#1*}{#1@floats}%
824       \expandafter\prepdef\csname#1\endcsname{\trigger@float@par}%
825       \expandafter\prepdef\csname#1*\endcsname{\trigger@float@par}%
826       \@namedef{fps@#1}{h!}%
827       \expandafter\immediate
828       \expandafter\closeout
829         \csname#1@write\endcsname
830     \everypar{%
831       \global\let\trigger@float@par\relax
832       \global\everypar{}\setbox\z@\lastbox
833       \ifxundefined@cs{#1@name}{}%
834         \begin@float@pagebreak
835         \expandafter\section
836         \expandafter*%
837         \expandafter{%
838           \csname#1@name\endcsname
839         }%
840     }%
841   }%
```



```

842   \input{\csname#1@stream\endcsname}%
843   \endgroup
844   \global\expandafter\let\csname#1write\endcsname\relax
845 }%
846 }%

```

`\tally@float` If we are tallying column inches, `\tally@float` tallies a contribution to `\ftype@`
`\total@float` `\captype`, 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 .

```

847 \chardef\@xvi=16\relax
848 \mathchardef\@twopowerfourteen="4000
849 \mathchardef\@twopowertwo="4
850 \def\tally@float#1{%
851   \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`.

```

852   \@tempcnta\count\@currbox
853   \divide\@tempcnta\@xxxii
854   \multiply\@tempcnta\@xxxii
855   \advance\count\@currbox-\@tempcnta
856   \divide\@tempcnta\@xxxii
857   \ifnum\count\@currbox>\@xvi}{%
858     \advance\count\@currbox-\@xvi\@booleantrue\@temp@sw
859   }{%
860     \@booleanfalse\@temp@sw
861   }%

```

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

```

862   \show@box@size@sw{%
863     \class@info{Float #1
864       (\the\@tempcnta)[\@temp@sw{16+}{}]\the\count\@currbox]^^J%
865       (\the\ht\@currbox+\the\dp\@currbox)X\the\wd\@currbox
866     }%
867   }{}%
868 \endgroup

```

Here we tally the height of this float object.

```

869 \expandafter\let
870 \expandafter\@tempa
871   \csname fbox@\csname ftype@#1\endcsname\endcsname
872 \ifnotrelax\@tempa{%
873   \ifhbox\@tempa{%
874     \setbox\@tempboxa\vbox{\unvcopy\@currbox\hrule}%
875     \dimen@ \ht\@tempboxa
876     \divide\dimen@\@twopowerfourteen
877     \ifdim\wd\@tempboxa<\textwidth}{%
878       \advance\dimen@\ht\@tempa
879       \global\ht\@tempa\dimen@
880     }{%
881       \advance\dimen@\dp\@tempa

```

```

882 \global\dp\@tempa\dimen@
883 }%
884 }{}%
885 }{}%
886 }%
887 \def\total@float#1{%
888 \expandafter\let
889 \expandafter\@tempa
890 \csname fbox@\csname ftype@#1\endcsname\endcsname
891 \ifnotrelax\@tempa{%
892 \ifhbox\@tempa{%
893 \@tempdima\the\ht\@tempa\divide\@tempdima\@twopowertwo\@tempcnta\@tempdima
894 \@tempdimb\the\dp\@tempa\divide\@tempdimb\@twopowertwo\@tempcntb\@tempdimb
895 \class@info{Total #1: Column(\the\@tempcnta pt), Page(\the\@tempcnta pt)}%
896 }{}%
897 }{}%
898 }%

```

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

```

899 \def\write@float#1{\write@@float{#1}{#1}}%
900 \def\endwrite@float{\@Esphack}%
901 \def\write@floats#1{\write@@float{#1*}{#1}}%
902 \def\endwrite@floats{\@Esphack}%

```

`\write@@float`

```

903 \def\write@@float#1#2{%
904 \ifhmode
905 \@bsphack
906 \fi
907 \chardef\@tempc\csname#2write\endcsname
908 \toks@{\begin{#1}}%
909 \def\@tempb{#1}%
910 \expandafter\let\csname end#1\endcsname\endwrite@float
911 \catcode'\^M\active
912 \@makeother\{\@makeother\}\@makeother\%
913 \write@floatline
914 }%

```

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

```

915 \begingroup
916 \catcode'\[the\catcode'\{\catcode'\]\the\catcode'\}\@makeother\{\@makeother\}%
917 \gdef\float@end@tag#1\end{#2}#3\@nul[%
918 \def\@tempa{#2}%
919 \ifx[\@tempa\@tempb][\end{#2}][\write@floatline]%
920 ]%
921 \obeylines%
922 \gdef\write@floatline#1^M[%
923 \begingroup%
924 \newlinechar'\^M%

```

```

925 \toks@ \expandafter [\the\toks@#1] \immediate \write \@tempc [\the\toks@] %
926 \endgroup %
927 \toks@ [] %
928 \float@end@tag#1 \end{} \@nul %
929 ] %
930 \endgroup

```

6.12 Counters

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

```

931 \def\@alph#1{\ifcase#1\or a\or b\or c\or d\else\@ialph{#1}\fi}
932 \def\@ialph#1{\ifcase#1\or \or \or \or \or e\or f\or g\or h\or i\or j\or
933 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
934 y\or z\or aa\or bb\or cc\or dd\or ee\or ff\or gg\or hh\or ii\or jj\or
935 kk\or ll\or mm\or nn\or oo\or pp\or qq\or rr\or ss\or tt\or uu\or
936 vv\or ww\or xx\or yy\or zz\else\@ctrerr\fi}

```

6.13 Customization of Sections

Patch the standard L^AT_EX sectioning procedure to:

- Allow a sectioning command to trigger the title page, or more generally to recognize that it is the first object in the document, so we headpatch `\@startsection`.
- Allow a tail command in #6 to uppercase the title, so we retain DPC's braces.
- Allow each type of sectioning command to format its number differently, so we generalize `\@secntformat`.
- Allow each type of sectioning command to format its argument differently, so we generalize `\@changfrom`.
- 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.

```

937 \def\@startsection#1#2#3#4#5#6{%
938 \@startsection@hook
939 \if@noskipsec \leavevmode \fi
940 \par
941 \@tempskipa #4\relax
942 \@afterindenttrue
943 \ifdim \@tempskipa <\z@
944 \@tempskipa -\@tempskipa \@afterindentfalse

```

```

945 \fi
946 \if@nobreak
947 \everypar{}%
948 \else
949 \addpenalty\@secpenalty\addvspace\@tempskipa
950 \fi
951 \@ifstar
952 {\@dblarg{\@sect@ltx{#1}{#2}{#3}{#4}{#5}{#6}}}%
953 {\@dblarg{\@sect@ltx {#1}{#2}{#3}{#4}{#5}{#6}}}%
954 }%
955 \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^AT_EX procedure.

```

956 \class@info{Repairing broken LaTeX \string\@sect}%
957 \def\@sect@ltx#1#2#3#4#5#6[#7]#8{%
958   \@ifnum{#2}>\c@secnumdepth}{%
959     \def\H@svsec{\phantomsection}%
960     \let\@svsec\@empty
961   }{%
962     \H@refstepcounter{#1}%
963     \def\H@svsec{%
964       \phantomsection
965     }%
966     \protected@edef\@svsec{{#1}}%
967     \@ifundefined{@#1cntformat}{%
968       \prepdef\@svsec\@secntformat
969     }{%
970       \expandafter\prepdef
971       \expandafter\@svsec
972       \csname @#1cntformat\endcsname
973     }%
974   }%
975   \@tempskipa #5\relax
976   \@ifdim{\@tempskipa>\z@}{%
977     \begingroup
978       \interlinepenalty \@M
979       #6{%
980         \@ifundefined{@hangfrom@#1}{\@hang@from}{\csname @hangfrom@#1\endcsname}%
981         {\hskip#3\relax\H@svsec}{\@svsec}{#8}%
982       }%
983     \@@par
984   \endgroup
985   \@ifundefined{#1mark}{\@gobble}{\csname #1mark\endcsname}{#7}%

```

```

986 \addcontentsline{toc}{#1}{%
987 \ifnum{#2}>\c@secnumdepth}{%
988 \protect\numberline{}%
989 }{%
990 \protect\numberline{\csname the#1\endcsname}%
991 }%
992 #8}%
993 }{%
994 \def\@svsechd{%
995 #6{%
996 \ifundefined{@runin@to@#1}{\@runin@to}{\csname @runin@to@#1\endcsname}%
997 {\hskip#3\relax\H@svsec}{\@svsec}{#8}%
998 }%
999 \ifundefined{#1mark}{\@gobble}{\csname #1mark\endcsname}{#7}%
1000 \addcontentsline{toc}{#1}{%
1001 \ifnum{#2}>\c@secnumdepth}{%
1002 \protect\numberline{}%
1003 }{%
1004 \protect\numberline{\csname the#1\endcsname}%
1005 }%
1006 #8}%
1007 }%
1008 }%
1009 \@xsect{#5}%
1010 }%
1011 \def\@hangfrom#1#2#3{\@hangfrom{#1#2}#3}%
1012 \def\@runin@to #1#2#3{#1#2#3}%

```

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

Note that, for compatibility with the `hyperref` package, we need to provide the interface required by that package (actually required by `pdfmark.def` and `nameref.sty`), namely the definition of `\@currentlabelname` (but now removed), the insertion of the procedure `\Sectionformat` (but why is this needed?), and the call to `\phantomsection` (which must precede the call to `\addcontentsline`). We also have to sidestep the patch to `\@ssect` in that same file, therefore we use a different control sequence name in the call from `\@startsection`.

```

1013 \def\@ssect@ltx#1#2#3#4#5#6[#7]#8{%
Removed \def\@currentlabelname{#8}
1014 \def\H@svsec{\phantomsection}%
1015 \@tempskipa #5\relax
1016 \ifdim\@tempskipa>\z@{%
1017 \begingroup
1018 \interlinepenalty \@M
1019 #6{%
1020 \ifundefined{@hangfroms@#1}{\@hangfroms}{\csname @hangfroms@#1\endcsname}%
Removed {\hskip#3\relax\H@svsec}{\Sectionformat{#8}{#1}}
1021 {\hskip#3\relax\H@svsec}{#8}%
1022 }%
1023 \@@par
1024 \endgroup
1025 \ifundefined{#1mark}{\@gobble}{\csname #1mark\endcsname}{#7}%

```

```

1026 \addcontentsline{toc}{#1}{\protect\numberline{#8}}%
1027 }{%
1028 \def\@svsechd{%
1029 #6{%
1030 \ifundefined{@runin@tos@#1}{\@runin@tos}{\csname @runin@tos@#1\endcsname}%
Removed {\hskip#3\relax\H@svsec}{\Sectionformat{#8}{#1}}
1031 {\hskip#3\relax\H@svsec}{#8}}%
1032 }%
1033 \ifundefined{#1smark}{\@gobble}{\csname #1smark\endcsname}{#7}}%
1034 \addcontentsline{toc}{#1}{\protect\numberline{#8}}%
1035 }%
1036 }%
1037 \xsect{#5}}%
1038 }%
1039 \def\@hang@froms#1#2{#1#2}%
1040 \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.

```

1041 \def\init@hyperref{%
1042 \providecommand\phantomsection{}%
1043 \providecommand\hyper@makecurrent[1]{}%
1044 \providecommand\Hy@raisedlink[1]{}%
1045 \providecommand\hyper@anchorstart[1]{}%
1046 \providecommand\hyper@anchorend{}%
1047 \providecommand\hyper@linkstart[2]{}%
1048 \providecommand\hyper@linkend{}%
1049 \providecommand\@currentHref{}%
1050 }%
1051 \let\H@refstepcounter\refstepcounter
1052 \appdef\document@inithook{%
1053 \init@hyperref
1054 }%

```

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

The following setting gives the L^AT_EX default.

```
1055 \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` tabular and array environments. A document class can define these hooks as needed.

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

```
1056 \appdef\document@inithook{%
1057   \ifpackageloaded{array}{\switch@array}{\switch@tabular}%
1058   \prepdef\endtabular{\endtabular@hook}%
1059   \@provide\endtabular@hook{}%
1060   \prepdef\endarray{\endarray@hook}%
1061   \@provide\endarray@hook{}%
1062   \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.

```
1063 \prepdef\@tabular{\tabular@hook}%
1064 \@provide\tabular@hook{}%
1065 }%
```

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

```
1066 \def\switch@tabular{%
1067   \let\@array@sw\@array@sw@array
1068   \ifx{\@array\@array@LaTeX}{%
1069     \ifx{\multicolumn\multicolumn@LaTeX}{%
1070       \ifx{\@tabular\@tabular@LaTeX}{%
1071         \ifx{\@tabarray\@tabarray@LaTeX}{%
1072           \ifx{\array\array@LaTeX}{%
1073             \ifx{\endarray\endarray@LaTeX}{%
1074               \ifx{\endtabular\endtabular@LaTeX}{%
1075                 \ifx{\@mkpream\@mkpream@LaTeX}{%
1076                   \ifx{\@addamp\@addamp@LaTeX}{%
1077                     \ifx{\@arrayacol\@arrayacol@LaTeX}{%
1078                       \ifx{\@tabacol\@tabacol@LaTeX}{%
1079                         \ifx{\@arrayclassz\@arrayclassz@LaTeX}{%
1080                           \ifx{\@tabclassiv\@tabclassiv@LaTeX}{%
1081                             \ifx{\@arrayclassiv\@arrayclassiv@LaTeX}{%
1082                               \ifx{\@tabclassz\@tabclassz@LaTeX}{%
1083                                 \ifx{\@classv\@classv@LaTeX}{%
1084                                   \ifx{\hline\hline@LaTeX}{%
1085                                     \ifx{\@tabularcr\@tabularcr@LaTeX}{%
1086                                       \ifx{\@xtabularcr\@xtabularcr@LaTeX}{%
1087                                         \ifx{\@xargarraycr\@xargarraycr@LaTeX}{%
1088                                           \ifx{\@yargarraycr\@yargarraycr@LaTeX}{%
1089                                             \true@sw
```

```

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 \false@sw
1143 } %

```



```

1144 }{%
1145   \false@sw
1146 }%
1147 }{%
1148   \false@sw
1149 }%
1150 }{%
1151   \false@sw
1152 }%
1153 {%
1154   \class@info{Patching LaTeX tabular.}%
1155 }{%
1156   \class@info{Unrecognized LaTeX tabular. Please update this document class! (Proceeding with
1157 }%
1158 \let\@array\@array@ltx
1159 \let\multicolumn\multicolumn@ltx
1160 \let\@tabular\@tabular@ltx
1161 \let\@tabarray\@tabarray@ltx
1162 \let\array\array@ltx
1163 \let\endarray\endarray@ltx
1164 \let\endtabular\endtabular@ltx
1165 \let\@mkpream\@mkpream@ltx
1166 \let\@addamp\@addamp@ltx
1167 \let\@arrayacol\@arrayacol@ltx
1168 \let\@tabacol\@tabacol@ltx
1169 \let\@arrayclassz\@arrayclassz@ltx
1170 \let\@tabclassiv\@tabclassiv@ltx
1171 \let\@arrayclassiv\@arrayclassiv@ltx
1172 \let\@tabclassz\@tabclassz@ltx
1173 \let\@classv\@classv@ltx
1174 \let\hline\hline@ltx
1175 \let\@tabularcr\@tabularcr@ltx
1176 \let\@xtabularcr\@xtabularcr@ltx
1177 \let\@xarraycr\@xarraycr@ltx
1178 \let\@yarraycr\@yarraycr@ltx
1179 }%

1180 \def\switch@array{%
1181   \ifpackageloaded{colortbl}{\let\switch@array@info\colortbl@message}{\let\switch@array@info
1182   \let\@array@sw\@array@sw@LaTeX
1183   \ifx{\@array\@array@array}{}%
1184     \ifx{\@tabular\@tabular@array}{}%
1185       \ifx{\@tabarray\@tabarray@array}{}%
1186         \ifx{\array\array@array}{%
1187           \ifx{\endarray\endarray@array}{%
1188             \ifx{\endtabular\endtabular@array}{%
1189               \ifx{\@mkpream\@mkpream@array}{%
1190                 \ifx{\@classx\@classx@array}{%
1191                   \ifx{\insert@column\insert@column@array}{%
1192                     \ifx{\@arraycr\@arraycr@array}{%
1193                       \ifx{\@xarraycr\@xarraycr@array}{%
1194                         \ifx{\@xarraycr\@xarraycr@array}{%
1195                           \ifx{\@yarraycr\@yarraycr@array}{%
1196                             \true@sw

```

```

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 }{ %
1225 \false@sw
1226 } %
1227 }{ %
1228 \false@sw
1229 } %
1230 }{ %
1231 \false@sw
1232 } %
1233 }{ %
1234 \false@sw
1235 }{ %
1236 \class@info{Patching array package.}%
1237 }{ %
1238 \switch@array@info
1239 } %
1240 \let\@array \@array@array@new
1241 \let\@@array \@array % Così fan tutti
1242 \let\@tabular \@tabular@array@new
1243 \let\@tabarray \@tabarray@array@new
1244 \let\array \@array@array@new
1245 \let\endarray \@endarray@array@new
1246 \let\endtabular\endtabular@array@new
1247 \let\@mkpream \@mkpream@array@new
1248 \let\@classx \@classx@array@new
1249 \let\@arrayacol\@arrayacol@ltx
1250 \let\@tabacol \@tabacol@ltx

```

```

1251 \let\insert@column\insert@column@array@new
1252 \expandafter\let\csname endtabular*\endcsname\endtabular % Così fan tutti
1253 \let\@arraycr \@arraycr@new
1254 \let\@xarraycr \@xarraycr@new
1255 \let\@xargarraycr\@xargarraycr@new
1256 \let\@yargarraycr\@yargarraycr@new
1257 }%
1258 \def\array@message{%
1259 \class@info{Unrecognized array package. Please update this document class! (Proceeding with
1260 }%
1261 \def\colortbl@message{%
1262 \class@info{colortbl package is loaded. (Proceeding with fingers crossed.)}%
1263 }%

\@array@sw The Boolean \@array@sw must be different depending on whether the array pack-
age is loaded.
1264 \def\@array@sw@LaTeX{\@ifx{\@tabularcr}}%
1265 \def\@array@sw@array{\@ifx{\dollarbegin\beginngroup}}%

\@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.
1266 \def\@tabular@LaTeX{%
1267 \leavevmode
1268 \hbox\bgroup$%
1269 \let\@acol\@tabacol
1270 \let\@classz\@tabclassz
1271 \let\@classiv\@tabclassiv
1272 \let\@tabularcr
1273 \@tabarray
1274 }%
1275 \def\@tabular@ltx{%
1276 \let\@acoll\@tabacoll
1277 \let\@acolr\@tabacolr
1278 \let\@acol\@tabacol
1279 \let\@classz\@tabclassz
1280 \let\@classiv\@tabclassiv
1281 \let\@tabularcr
1282 \@tabarray
1283 }%
1284 \def\@tabular@array{%
1285 \leavevmode
1286 \hbox\bgroup$%
1287 \col@sep\tabcolsep
1288 \let\dollarbegin\beginngroup
1289 \let\dollarend\endgroup
1290 \@tabarray
1291 }%
1292 \def\@tabular@array@new{%
1293 \let\@acoll\@tabacoll
1294 \let\@acolr\@tabacolr
1295 \let\@acol\@tabacol

```

sepundefined

```
1296 \let\dollarbegin\beginngroup
1297 \let\dollarend\endgroup
1298 \@tabarray
1299 }%
```

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

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

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

```
1300 \def\@tabarray@LaTeX{%
1301 \m@th\@ifnextchar[\@array{\@array[c]}%
1302 }%
1303 \def\@tabarray@ltx{%
1304 \m@th\@ifnextchar[\@array{\expandafter\@array\expandafter[\array@default]]}%
1305 }%
1306 \def\@tabarray@array{%
1307 \@ifnextchar[{\@array}{\@array[c]}%
1308 }%
1309 \def\@tabarray@array@new{%
1310 \@ifnextchar[{\@array}{\expandafter\@array\expandafter[\array@default]]}%
1311 }%
```

`\@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 `\@xargarraycr` tabular will cleave together.

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

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

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

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

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

```
1312 \newcount\intertabularlinepenalty
1313 \intertabularlinepenalty=100
1314 \newcount\@tbpen
1315 \appdef\samepage{\intertabularlinepenalty\@M}%
1316 \def\@tabularcr@LaTeX{{\ifnum 0='}\fi \@ifstar \@xtabularcr \@tabularcr}%
1317 \def\@tabularcr@ltx{{\ifnum 0='}\fi \@ifstar {\global \@tbpen \@M \@xtabularcr }{\global \@t
1318 \def\@xtabularcr@LaTeX{\@ifnextchar [\@argtabularcr {\ifnum 0='{ \fi } \cr }}%
1319 \def\@xtabularcr@ltx{\@ifnextchar [\@argtabularcr {\ifnum 0='{ \fi } \cr \noalign {\penalty \@
1320 \def\@xargarraycr@LaTeX#1{\@tempdima #1\advance \@tempdima \dp \@arstrutbox \vrule \@height
```

```

1321 \def\@xargarraycr@ltx#1{\@tempdima #1\advance \@tempdima \dp \@arstrutbox \vrule \@height \z@
1322 \def\@yargarraycr@LaTeX#1{\cr \noalign {\vskip #1}}%
1323 \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.

1324 \def\@arraycr@array{%
1325   \relax
1326   \iffalse{\fi\ifnum 0='}\fi
1327   \@ifstar \@xarraycr \@xarraycr
1328 }%
1329 \def\@arraycr@new{%
1330   \relax
1331   \iffalse{\fi\ifnum 0='}\fi
1332   \@ifstar {\global \@tbpen \@M \@xarraycr }{\global \@tbpen \intertabularlinepenalty \@xarra
1333 }%
1334 \def\@xarraycr@array{%
1335   \@ifnextchar [%]
1336   \@argarraycr {\ifnum 0='{ }\fi\cr}%
1337 }%
1338 \def\@xarraycr@new{%
1339   \@ifnextchar [%]
1340   \@argarraycr {\ifnum 0='{ }\fi\cr \noalign {\penalty \@tbpen }}%
1341 }%
1342 \def\@xargarraycr@array#1{%
1343   \unskip
1344   \@tempdima #1\advance\@tempdima \dp\@arstrutbox
1345   \vrule \@depth\@tempdima \@width\z@
1346   \cr
1347 }%
1348 \def\@xargarraycr@new#1{%
1349   \unskip
1350   \@tempdima #1\advance\@tempdima \dp\@arstrutbox
1351   \vrule \@depth\@tempdima \@width\z@
1352   \cr
1353   \noalign {\penalty \@tbpen }%
1354 }%
1355 \def\@yargarraycr@array#1{%
1356   \cr
1357   \noalign{\vskip #1}%
1358 }%
1359 \def\@yargarraycr@new#1{%
1360   \cr
1361   \noalign{\penalty \@tbpen \vskip #1}%
1362 }%

```

\array We provide old and new versions of the `\array` procedure for both \LaTeX 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.

```

1363 \def\array@LaTeX{%
1364   \let\@acol\@arrayacol
1365   \let\@classz\@arrayclassz

```

```

1366 \let\@classiv\@arrayclassiv
1367 \let\\\@arraycr
1368 \let\@halignto\@empty
1369 \@tabarray
1370 }%
1371 \def\array@ltx{%
1372 \@ifmmode{\@badmath$}%
1373 \let\@acoll\@arrayacol
1374 \let\@acolr\@arrayacol
1375 \let\@acol\@arrayacol
1376 \let\@classz\@arrayclassz
1377 \let\@classiv\@arrayclassiv
1378 \let\\\@arraycr
1379 \let\@halignto\@empty
1380 \@tabarray
1381 }%
1382 \def\array@array{%
1383 \col@sep\arraycolsep
1384 \def\d@llarbegin{$}\let\d@llarend\d@llarbegin\gdef\@halignto{}%
1385 \@tabarray
1386 }
1387 \def\array@array@new{%
1388 \@ifmmode{\@badmath$}%
1389 \let\@acoll\@arrayacol
1390 \let\@acolr\@arrayacol
1391 \let\@acol\@arrayacol
Removed: \let\col@sep\@undefined
1392 \def\d@llarbegin{$}%
1393 \let\d@llarend\d@llarbegin
1394 \gdef\@halignto{}%
1395 \@tabarray
1396 }%

```

`\@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.

```

1397 \def\@array@LaTeX[#1]#2{%
1398 \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi\fi
1399 \bgroup
1400 \setbox\@arstrutbox\hbox{%
1401 \vrule \@height\arraystretch\ht\strutbox
1402 \@depth\arraystretch \dp\strutbox
1403 \@width\z@}%
1404 \mkpream{#2}%
1405 \edef\@preamble{%
1406 \ialign \noexpand\@halignto
1407 \bgroup \@arstrut \@preamble \tabskip\z@skip \cr}%
1408 \let\@startpbox\@startpbox \let\@endpbox\@endpbox

```

```

1409 \let\tabularnewline\\%
1410 \let\par\@empty
1411 \let\@sharp##%
1412 \set@typeset@protect
1413 \lineskip\z@skip\baselineskip\z@skip
1414 \ifhmode \@preamerr\z@ \@@par\fi
1415 \@preamble
1416 }%
1417 \def\@array@ltx[#1]#2{%
1418 \@nameuse{\@array@align@#1}%
1419 \set@arstrutbox
1420 \mkpream{#2}%
1421 \prepdef\@preamble{%
1422 \tabskip\tabmid@skip
1423 \@arstrut
1424 }%
1425 \appdef\@preamble{%
1426 \tabskip\tabright@skip
1427 \cr
1428 \array@row@pre
1429 }%
1430 % \let\@startpbox\@startpbox
1431 % \let\@endpbox\@endpbox
1432 \let\tabularnewline\\%
1433 \let\par\@empty
1434 \let\@sharp##%
1435 \set@typeset@protect
1436 \lineskip\z@skip\baselineskip\z@skip
1437 \tabskip\tableft@skip\relax
1438 \ifhmode \@preamerr\z@ \@@par\fi
1439 \everycr{}%
1440 \expandafter\halign\expandafter\@halignto\expandafter\bgroup\@preamble
1441 }%
1442 %
1443 \def\set@arstrutbox{%
1444 \setbox\@arstrutbox\hbox{%
1445 \vrule \@height\arraystretch\ht\strutbox
1446 \@depth\arraystretch \dp\strutbox
1447 \@width\z@
1448 }%
1449 }%

```

\@array@array

```

1450 \def\@array@array[#1]#2{%
1451 \@tempdima \ht \strutbox
1452 \advance \@tempdima by\extrarowheight
1453 \setbox \@arstrutbox \hbox{\vrule
1454 \@height \arraystretch \@tempdima
1455 \@depth \arraystretch \dp \strutbox
1456 \@width \z@}%
1457 \begingroup
1458 \mkpream{#2}%
1459 \xdef\@preamble{\noexpand \ialign \@halignto
1460 \bgroup \@arstrut \@preamble

```

```

1461 \tabskip \z@ \cr}%
1462 \endgroup
1463 \@arrayleft
1464 \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi \fi
1465 \bgroup
1466 \let \@sharp ##\let \protect \relax
1467 \lineskip \z@
1468 \baselineskip \z@
1469 \m@th
1470 \let\\\@arraycr \let\tabularnewline\\\let\par\@empty \@preamble
1471 }%
1472 \def\@array@array@new[#1]#2{%
1473 \@tempdima\ht\strutbox
1474 \advance\@tempdima by\extrarowheight
1475 \setbox\@arstrutbox\hbox{%
1476 \vrule \@height\arraystretch\@tempdima
1477 \@depth \arraystretch\dp\strutbox
1478 \@width \z@
1479 }%
1480 \begingroup
1481 \mkpream{#2}%
1482 \xdef\@preamble{\@preamble}%
1483 \endgroup
1484 \prepdef\@preamble{%
1485 \tabskip\tabmid@skip
1486 \@arstrut
1487 }%
1488 \appdef\@preamble{%
1489 \tabskip\tabright@skip
1490 \cr
1491 \array@row@pre
1492 }%
1493 \@arrayleft
1494 \@nameuse{\@array@align@#1}%
1495 \m@th
1496 \let\\\@arraycr
1497 \let\tabularnewline\\%
1498 \let\par\@empty
1499 \let\@sharp##%
1500 \set@typeset@protect
1501 \lineskip\z@\baselineskip\z@
1502 \tabskip\tableft@skip
1503 \everycr{}%
1504 \expandafter\halign\expandafter\@halignto\expandafter\bgroup\@preamble
1505 }%

```

`\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`.

```

1506 \def\endarray@LaTeX{%
1507 \crr\egroup\egroup
1508 }%
1509 \def\endarray@ltx{%
1510 \crr\array@row@pst\egroup\egroup

```



```

1511 }%
1512 \def\endarray@array{%
1513 \crr\egroup\egroup \@arrayright\gdef\@preamble{%
1514 }%
1515 \def\endarray@array@new{%
1516 \crr\array@row@pst\egroup\egroup % Same as \endarray@ltx
1517 \@arrayright
1518 \global\let\@preamble\@empty
1519 }%

```

\endtabular

```

1520 \def\endtabular@LaTeX{%
1521 \crr\egroup\egroup $\egroup
1522 }%
1523 \def\endtabular@ltx{%
1524 \endarray
1525 }%
1526 \def\endtabular@array{%
1527 \endarray $\egroup
1528 }%
1529 \def\endtabular@array@new{%
1530 \endarray
1531 }%

```

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

```

1532 \@namedef{endtabular*}{\endtabular}%

```

\multicolumn

```

1533 \long\def\multicolumn@LaTeX#1#2#3{%
1534 \multispan{#1}\begingroup
1535 \mkpream{#2}%
1536 \def\@sharp{#3}\set@typeset@protect
1537 \let\@startpbox\@startpbox\let\@endpbox\@endpbox
1538 \arstrut \preamble\hbox{}\endgroup\ignorespaces
1539 }%
1540 \long\def\multicolumn@ltx#1#2#3{%
1541 \multispan{#1}%
1542 \begingroup
1543 \mkpream{#2}%
1544 \def\@sharp{#3}%
1545 \set@typeset@protect
1546 %\let\@startpbox\@startpbox\let\@endpbox\@endpbox
1547 \arstrut
1548 \preamble
1549 \hbox{}%
1550 \endgroup
1551 \ignorespaces
1552 }%

```

\array@align@ Here are the various procedures for the vertical alignment options. The change from standard L^AT_EX is that we do not go into math mode in every case: only when required by `\vcenter`. Also, we use `\aftergroup` to close out the boxes

and modes we have started. It requires only that each procedure issue exactly one unmatched `\bgroup`.

We establish here the default vertical alignment.

```

1553 \def\@array@align@t{\leavevmode\vtop\bgroup}%
1554 \def\@array@align@b{\leavevmode\vbox\bgroup}%
1555 \def\@array@align@c{\leavevmode\@ifmmode{\vcenter\bgroup}{\vcenter\bgroup\aftergroup$\after
1556 \def\@array@align@v{%
1557   \@ifmmode{%
1558     \@badmath
1559     \vcenter\bgroup
1560   }{%
1561     \@ifinner{%
1562       $\vcenter\bgroup\aftergroup$
1563     }{%
1564       \@@par\bgroup
1565     }%
1566   }%
1567 }%
1568 \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.

```

1569 \def\array@row@rst{%
1570   \let\@array@align@v\@array@align@c
1571 }%
1572 \def\array@row@pre{%}%
1573 \def\array@row@pst{%}%

```

`\toprule` Default definitions for `\toprule`, `\colrule`, `\botrule`

```

\colrule 1574 \newcommand\toprule{\tab@rule{\column@font}{\column@fil}{\frstrut}}%
\botrule 1575 \newcommand\colrule{\unskip\lrstrut\\\tab@rule{\body@font}{\frstrut}}%
1576 \newcommand\botrule{\unskip\lrstrut\\\noalign{\hline@rule}{}}%

```

`\hline`

```

1577 \def\hline@LaTeX{%
1578   \noalign{\ifnum0=}\fi\hrule \@height \arrayrulewidth \futurelet
1579     \reserved@a\@xhline
1580 }%
1581 \def\hline@ltx{%
1582   \noalign{%
1583     \ifnum0=}\fi
1584     \hline@rule
1585     \futurelet\reserved@a\@xhline
1586   } \noalign ended in \@xhline
1587 }%
1588 \def\@xhline@unneeded{%
1589   \say\reserved@a
1590   \ifx\reserved@a\hline
1591     \vskip\doublerulesep

```

```

1592 \vskip-\arrayrulewidth
1593 \fi
1594 \ifnum0='{ \fi}%
1595 }%
1596 \def\tab@rule#1#2#3{%
1597 \crr
1598 \noalign{%
1599 \hline@rule
1600 \gdef\@arstrut@hook{%
1601 \global\let\@arstrut@hook\@empty
1602 #3%
1603 }%
1604 \gdef\cell@font{#1}%
1605 \gdef\cell@fil{#2}%
1606 }%
1607 }%
1608 \def\column@font{}%
1609 \def\column@fil{}%
1610 \def\body@font{}%
1611 \def\cell@font{}%
1612 \def\frstrut{}%
1613 \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@cclnc is a klookch, a magic number.

```

1614 \def\@arstrut@hline{%
1615 \relax
1616 \ifmmode{\copy}{\unhcopy}\@arstrutbox@hline
1617 \@arstrut@hook
1618 }%
1619 %
1620 \let\@arstrut@org\@arstrut
1621 \def\@arstrut@hook{%
1622 \global\let\@arstrut\@arstrut@org
1623 }%
1624 %
1625 \newbox\@arstrutbox@hline
1626 \appdef\set@arstrutbox{%
1627 \setbox\@arstrutbox@hline\hbox{%
1628 \setbox\z@\hbox{\$0~{0}_-}\$}%
1629 \dimen@ \ht\z@ \advance \dimen@ \@arstrut@hline@cclnc
1630 \ifdim \dimen@ < \arraystretch \ht\strutbox {\dimen@ = \arraystretch \ht\strutbox} {}%
1631 \vrule \@height \dimen@
1632 \depth \arraystretch \dp\strutbox
1633 \width \z@
1634 }%

```

```

1635 }%
1636 %
1637 \def\hline@rule{%
1638 \hrule \@height \arrayrulewidth
1639 \global\let\@arstrut\@arstrut@hline
1640 }%
1641 \def\@arstrut@hline@cinc{2\p@}% % Klootch: magic number

\tableleft@skip

1642 \def\tableleft@skip{\z@skip}%
1643 \def\tabmid@skip{\z@skip}%\@flushglue
1644 \def\tabright@skip{\z@skip}%
1645 \def\tableleftsep{\tabcolsep}%
1646 \def\tabmidsep{\tabcolsep}%
1647 \def\tabrightsep{\tabcolsep}%
1648 \def\cell@fil{}%
1649 \def\pbox@hook{}%

\@arstrut

1650 \appdef\@arstrut{\@arstrut@hook}%
1651 \let\@arstrut@hook\@empty
1652 \def\@addtopreamble{\appdef\@preamble}%

\@mkpream

1653 \def\@mkpream@LaTeX#1{%
1654 \@firstampttrue\@lastchclass6
1655 \let\@preamble\@empty
1656 \let\protect\@unexpandable@protect
1657 \let\@sharp\relax
1658 \let\@startpbox\relax\let\@endpbox\relax
1659 \@expast{#1}%
1660 \expandafter\@tfor \expandafter
1661 \@nextchar \expandafter:\expandafter=\reserved@a\do
1662 {\@testpach\@nextchar
1663 \ifcase \@chclass \@classz \or \@classi \or \@classii \or \@classiii
1664 \or \@classiv \or \@classv \fi\@lastchclass\@chclass}%
1665 \ifcase \@lastchclass \@acol
1666 \or \or \@preamerr \@ne\or \@preamerr \tw@\or \or \@acol \fi
1667 }%
1668 \def\@mkpream@ltx#1{%
1669 \@firstampttrue
1670 \@lastchclass6
1671 \let\@preamble\@empty
1672 \let\protect\@unexpandable@protect
1673 \let\@sharp\relax
1674 %\let\@startpbox\relax\let\@endpbox\relax
1675 \@expast{#1}%
1676 \expandafter\@tfor \expandafter\@nextchar \expandafter:\expandafter=\reserved@a
1677 \do{%
1678 \expandafter\@testpach \expandafter{\@nextchar}%
1679 \ifcase \@chclass
1680 \@classz
1681 \or
1682 \@classi

```

```

1683 \or
1684 \@classii
1685 \or
1686 \@classiii
1687 \or
1688 \@classiv
1689 \or
1690 \@classv
1691 \fi
1692 \@lastchclass\@chclass
1693 }%
1694 \ifcase\@lastchclass
1695 \@acolr % right-hand column
1696 \or
1697 \or
1698 \@preamerr\@ne
1699 \or
1700 \@preamerr\tw@
1701 \or
1702 \or
1703 \@acolr % right-hand column
1704 \fi
1705 }%

```

`\insert@column`

```

1706 \def\insert@column@array{%
1707   \the@toks \the \@tempcnta
1708   \ignorespaces \@sharp \unskip
1709   \the@toks \the \count@ \relax
1710 }%
1711 \def\insert@column@array@new{%
1712   \the@toks\the\@tempcnta
1713   \array@row@rst\cell@font
1714   \ignorespaces\@sharp\unskip
1715   \the@toks\the\count@
1716   \relax
1717 }%

```

`\@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^AT_EX, 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_EX'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`.

```

1718 \def\@mkpream@relax{%
1719   \let\tableftsep \relax
1720   \let\tabmidsep \relax
1721   \let\tabrightsep \relax
1722   \let\array@row@rst\relax

```

```

1723 \let\cell@font \relax
1724 \let\@startpbox \relax
1725 }%

```

\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.

```

1726 \def\mkpream@array#1{%
1727   \gdef\@preamble{}\@lastchclass 4 \@firstampttrue
1728   \let\@sharp\relax \let\@startpbox\relax \let\@endpbox\relax
1729   \@temptokena{#1}\@tempswatrue
1730   \@whilesw@if@tempswa\fi{\@tempswafalse\the\NC@list}%
1731   \count@m@ne
1732   \let\the@toks\relax
1733   \prepnext@tok
1734   \expandafter \@tfor \expandafter \@nextchar
1735   \expandafter :\expandafter =\the\@temptokena \do
1736   {\@testpach
1737    \ifcase \@chclass \@classz \or \@classi \or \@classii
1738     \or \save@decl \or \or \@classv \or \@classvi
1739     \or \@classvii \or \@classviii
1740     \or \@classx
1741     \or \@classx \fi
1742    \@lastchclass\@chclass}%
1743   \ifcase\@lastchclass
1744   \@acol \or
1745   \or
1746   \@acol \or
1747   \@preamerr \thr@@ \or
1748   \@preamerr \tw@ \@addtopreamble\@sharp \or
1749   \or
1750   \else \@preamerr \@ne \fi
1751   \def\the@toks{\the@toks}%
1752 }%
1753 \def\mkpream@array@new#1{%
1754   \gdef\@preamble{}\%
1755   \@lastchclass\@f@ur
1756   \@firstampttrue
1757   \let\@sharp\relax
1758   \mkpream@relax
1759   %\let\@startpbox\relax\let\@endpbox\relax
1760   \@temptokena{#1}\@tempswatrue
1761   \@whilesw@if@tempswa\fi{\@tempswafalse\the\NC@list}%
1762   \count@m@ne
1763   \let\the@toks\relax
1764   \prepnext@tok
1765   \expandafter \@tfor \expandafter \@nextchar \expandafter :\expandafter =\the\@temptokena
1766   \do{%
1767     \@testpach
1768     \ifcase\@chclass
1769       \@classz

```

```

1770 \or
1771 \@classi
1772 \or
1773 \@classii
1774 \or
1775 \save@decl
1776 \or
1777 \or
1778 \@classv
1779 \or
1780 \@classvi
1781 \or
1782 \@classvii
1783 \or
1784 \@classviii
1785 \or
1786 \@classx
1787 \or
1788 \@classx
1789 \fi
1790 \@lastchclass\@chclass
1791 }%
1792 \ifcase\@lastchclass
1793 \@acolr % right-hand column
1794 \or
1795 \or
1796 \@acolr % right-hand column
1797 \or
1798 \@preamerr\thr@@
1799 \or
1800 \@preamerr\tw@\@addtopreamble\@sharp
1801 \or
1802 \or
1803 \else
1804 \@preamerr\@ne
1805 \fi
1806 \def\the@toks{\the\toks}%
1807 }%

```

`\@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.

```

1808 \appdef\@mkpream@relax{%
1809 \let\CT@setup \relax
1810 \let\CT@color \relax
1811 \let\CT@do@color \relax
1812 \let\color \relax
1813 \let\CT@column@color\relax
1814 \let\CT@row@color \relax
1815 \let\CT@cell@color \relax
1816 }%

```

`\@addamp`

```

1817 \def\@addamp@LaTeX{%
1818   \if@firstamp\@firstampfalse\else\edef\@preamble{\@preamble &}\fi
1819 }%
1820 \def\@addamp@ltx{%
1821   \if@firstamp\@firstampfalse\else\@addtopreamble{&}\fi
1822 }%

\@arrayacol
1823 \def\@arrayacol@LaTeX{%
1824   \edef\@preamble{\@preamble \hskip \arraycolsep}%
1825 }%
1826 \def\@arrayacol@ltx{%
1827   \@addtopreamble{\hskip\arraycolsep}%
1828 }%

\@tabacol
1829 \def\@tabacoll{%
1830   \@addtopreamble{\hskip\tableftsep\relax}%
1831 }%
1832 \def\@tabacol@LaTeX{%
1833   \edef\@preamble{\@preamble \hskip \tabcolsep}%
1834 }%
1835 \def\@tabacol@ltx{%
1836   \@addtopreamble{\hskip\tabmidsep\relax}%
1837 }%
1838 \def\@tabacolr{%
1839   \@addtopreamble{\hskip\tabrightsep\relax}%
1840 }%

\@arrayclassz
1841 \def\@arrayclassz@LaTeX{%
1842   \ifcase \@lastchclass \@acolampacol \or \@ampacol \or
1843     \or \or \@addamp \or
1844     \@acolampacol \or \@firstampfalse \@acol \fi
1845   \edef\@preamble{\@preamble
1846     \ifcase \@chnum
1847       \hfil$\relax\@sharp$\hfil \or $\relax\@sharp$\hfil
1848       \or \hfil$\relax\@sharp$\fi}%
1849 }%
1850 \def\@arrayclassz@ltx{%
1851   \ifcase \@lastchclass
1852     \@acolampacol
1853   \or
1854     \@ampacol
1855   \or
1856   \or
1857   \or
1858     \@addamp
1859   \or
1860     \@acolampacol
1861   \or
1862     \@firstampfalse\@acoll
1863   \fi
1864   \ifcase \@chnum

```



```

1865 \@addtopreamble{%
1866   \hfil\array@row@rst$\relax\@sharp$\hfil
1867 }%
1868 \or
1869 \@addtopreamble{%
1870   \array@row@rst$\relax\@sharp$\hfil
1871 }%
1872 \or
1873 \@addtopreamble{%
1874   \hfil\array@row@rst$\relax\@sharp$%
1875 }%
1876 \fi
1877 }%

\@tabclassz
1878 \def\@tabclassz@LaTeX{%
1879   \ifcase\@lastchclass
1880     \@acolampacol
1881   \or
1882     \@ampacol
1883   \or
1884   \or
1885   \or
1886     \@addamp
1887   \or
1888     \@acolampacol
1889   \or
1890     \@firstampfalse\@acol
1891 \fi
1892 \edef\@preamble{%
1893   \@preamble{%
1894     \ifcase\@chnum
1895       \hfil\ignorespaces\@sharp\unskip\hfil
1896     \or
1897       \hskip1sp\ignorespaces\@sharp\unskip\hfil
1898     \or
1899       \hfil\hskip1sp\ignorespaces\@sharp\unskip
1900     \fi}}%
1901 }%
1902 \def\@tabclassz@ltx{%
1903   \ifcase\@lastchclass
1904     \@acolampacol
1905   \or
1906     \@ampacol
1907   \or
1908   \or
1909   \or
1910     \@addamp
1911   \or
1912     \@acolampacol
1913   \or
1914     \@firstampfalse\@acoll
1915 \fi
1916 \ifcase\@chnum

```

```

1917 \@addtopreamble{%
1918   {\hfil\array@row@rst\cell@font\ignorespaces\@sharp\unskip\hfil}%
1919 }%
1920 \or
1921 \@addtopreamble{%
1922   {\cell@fil\hskip1sp\array@row@rst\cell@font\ignorespaces\@sharp\unskip\hfil}%
1923 }%
1924 \or
1925 \@addtopreamble{%
1926   {\hfil\hskip1sp\array@row@rst\cell@font\ignorespaces\@sharp\unskip\cell@fil}%
1927 }%
1928 \fi
1929 }%

\@tabclassiv
1930 \def\@tabclassiv@LaTeX{%
1931   \@addtopreamble\@nextchar
1932 }%
1933 \def\@tabclassiv@ltx{%
1934   \expandafter\@addtopreamble\expandafter{\@nextchar}%
1935 }%

\@arrayclassiv
1936 \def\@arrayclassiv@LaTeX{%
1937   \@addtopreamble{$\@nextchar$}%
1938 }%
1939 \def\@arrayclassiv@ltx{%
1940   \expandafter\@addtopreamble\expandafter{\expandafter$\@nextchar$}%
1941 }%

\@classv
1942 \def\@classv@LaTeX{%
1943   \@addtopreamble{\@startpbox{\@nextchar}\ignorespaces
1944   \@sharp\@endpbox}%
1945 }%
1946 \def\@classv@ltx{%
1947   \expandafter\@addtopreamble
1948   \expandafter{%
1949     \expandafter \@startpbox
1950     \expandafter {\@nextchar}%
1951     \pbox@hook\array@row@rst\cell@font\ignorespaces\@sharp\@endpbox
1952   }%
1953 }%

\@classx
1954 \def\@classx@array{%
1955   \ifcase \@lastchclass
1956     \@acolampacol \or
1957     \@addamp \@acol \or
1958     \@acolampacol \or
1959     \or
1960     \@acol \@firstampfalse \or
1961     \@addamp
1962     \fi

```

```

1963 }%
1964 \def\@classx@array@new{%
1965 \ifcase \@lastchclass
1966 \acolampacol
1967 \or
1968 \@addamp \@acol
1969 \or
1970 \@acolampacol
1971 \or
1972 \or
1973 \@firstampfalse\@acoll
1974 \or
1975 \@addamp
1976 \fi
1977 }%

```

6.15 Repair other broken parts of L^AT_EX

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

```

1978 \def\bitor@LaTeX #1{\@tempcntb \count#1
1979 \ifnum \@tempcnta =\z@
1980 \else
1981 \divide\@tempcntb\@tempcnta
1982 \ifodd\@tempcntb \@testtrue\fi
1983 \fi}%
1984 \def\bitor@ltx#1{%
1985 \@tempcntb\count#1\relax
1986 \@ifnum{\@tempcnta=\z@}{\fi}%
1987 \divide\@tempcntb\@tempcnta
1988 \@ifodd\@tempcntb{\@testtrue}{\fi}%
1989 }%
1990 }%
1991 \ifx{\xbitor\bitor@LaTeX}{%
1992 \class@info{Repairing broken LaTeX \string\bitor}%
1993 }{%
1994 \class@info{Unrecognized LaTeX \string\bitor. Please update this document class! (Procee
1995 }%
1996 \let\bitor\bitor@ltx

```

6.16 Syntax

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

```

1997 \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.

```

1998 \def\@starttoc#1{%
1999   \begingroup
2000     \toc@pre
2001     \makeatletter
2002     \@input{\jobname.#1}%
2003     \if@files
2004       \expandafter\newwrite\csname tf@#1\endcsname
2005       \immediate\openout \csname tf@#1\endcsname \jobname.#1\relax
2006     \fi
2007     \@nobreakfalse
2008     \toc@post
2009   \endgroup
2010 }%
2011 \def\toc@pre{%
2012 \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.

```

2013 \def\toc@font{%
2014 \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.

```

2015 \let\tocdim@section      \leftmargini
2016 \let\tocdim@subsection   \leftmarginii
2017 \let\tocdim@subsubsection \leftmarginiii
2018 \let\tocdim@paragraph    \leftmarginiv
2019 \let\tocdim@appendix     \leftmarginv
2020 \let\tocdim@pagenum      \leftmarginvi

```

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

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

To activate this override of the standard L^AT_EX processing, the substyle does: `\let\toc@pre\toc@pre@auto` and `\let\toc@post\toc@post@auto`.

```

2021 \def\toc@pre@auto{%
2022   \toc@@font
2023   \@tempdima\z@
2024   \toc@setindent\@tempdima{section}%
2025   \toc@setindent\@tempdima{subsection}%
2026   \toc@setindent\@tempdima{subsubsection}%
2027   \toc@setindent\@tempdima{paragraph}%
2028   \toc@letdimen{appendix}%
2029   \toc@letdimen{pagenum}%
2030 }%
2031 \def\toc@post@auto{%
2032   \if@filesw
2033     \begingroup
2034       \toc@writedimen{section}%
2035       \toc@writedimen{subsection}%
2036       \toc@writedimen{subsubsection}%
2037       \toc@writedimen{paragraph}%
2038       \toc@writedimen{appendix}%
2039       \toc@writedimen{pagenum}%
2040     \endgroup
2041   \fi
2042 }%
```

`\toc@setindent`

```

2043 \def\toc@setindent#1#2{%
2044   \csname tocdim@#2\endcsname\tocdim@min\relax
2045   \@ifundefined{tocmax@#2}{\@namedef{tocmax@#2}{\z@}}{}%
2046   \advance#1\@nameuse{tocmax@#2}\relax
2047   \expandafter\edef\csname tocleft@#2\endcsname{\the#1}%
2048 }%
```

`\toc@letdimen`

```

2049 \def\toc@letdimen#1{%
2050   \csname tocdim@#1\endcsname\tocdim@min\relax
2051   \@ifundefined{tocmax@#1}{\@namedef{tocmax@#1}{\z@}}{}%
2052   \expandafter\let\csname tocleft@#1\expandafter\endcsname\csname tocmax@#1\endcsname
2053 }%
```

`\toc@writedimen`

```

2054 \def\toc@writedimen#1{%
2055   \immediate\write\@auxout{%
2056     \gdef\expandafter\string\csname tocmax@#1\endcsname{%
2057       \expandafter\the\csname tocdim@#1\endcsname
2058     }%
2059   }%
2060 }%
```

`\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 writ-

ten 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

```
2061 \def\l@sections#1#2#3#4{%
2062   \begingroup
2063   \everypar{}%
2064   \set@tocdim@pagenum\@tempboxa{#4}%
2065   \global\@tempdima\csname tocdim@#2\endcsname
2066   \leftskip\csname tocleft@#2\endcsname\relax
2067   \dimen@\csname tocleft@#1\endcsname\relax
2068   \parindent-\leftskip\advance\parindent\dimen@
2069   \rightskip\tocleft@pagenum plus 1fil\relax
2070   \skip@\parfillskip\parfillskip\z@
2071   \let\numberline\numberline@@sections
2072   \nameuse{lof@#2}%
2073   \ignorespaces#3\unskip\nobreak\hskip\skip@
2074   \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`).

```
2075   \expandafter\par
2076   \expandafter\aftergroup\csname tocdim@#2%
2077   \expandafter\endcsname
2078   \expandafter\endgroup
2079   \the\@tempdima\relax
2080 }%
```

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

```
2081 \def\set@tocdim@pagenum#1#2{%
2082   \setbox#1\hbox{\ignorespaces#2}%
2083   \@ifdim{\tocdim@pagenum<\wd#1}{\global\tocdim@pagenum\wd#1}{}%
2084 }%
```

`\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!

```
2085 \def\numberline@@sections#1{%
2086   \leavevmode\hb@xt@-\parindent{%
```

```

2087 \hfil
2088 \@ifempty{#1}{}{%
2089   \setbox\z@\hbox{#1.\kern\ltxu@dotsep}%
2090   \@ifdim{\@tempdima<\wd\z@}{\global\@tempdima\wd\z@}{}%
2091   \unhbox\z@
2092 }%
2093 }%
2094 \ignorespaces
2095 }%
2096 \def\tocdim@min{\z@}%

```

6.18 Lists

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

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

```

2097 \def\list#1#2{%
2098   \ifnum \@listdepth >5\relax
2099     \@toodeep
2100   \else
2101     \global\advance\@listdepth\@ne
2102   \fi
2103   \rightmargin\z@
2104   \listparindent\z@
2105   \itemindent\z@
2106   \csname @list\romannumeral\the\@listdepth\endcsname
2107   \def\@itemlabel{#1}%
2108   \let\makelabel\@mklab
2109   \@nmblistfalse
2110   #2\relax
2111   \@trivlist
2112   \parskip\parsep
2113   \set@listindent
2114   \ignorespaces
2115 }%
2116 \def\set@listindent@parshape{%
2117   \parindent\listparindent
2118   \advance\@totalleftmargin\leftmargin
2119   \advance\linewidth-\rightmargin
2120   \advance\linewidth-\leftmargin
2121   \parshape\@ne\@totalleftmargin\linewidth
2122 }%
2123 \def\set@listindent@{%
2124   \parindent\listparindent
2125   \advance\@totalleftmargin\leftmargin
2126   \advance\rightskip\rightmargin
2127   \advance\leftskip\@totalleftmargin
2128 }%
2129 \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 2130 \providecommand\href[0]{\begingroup\@sanitize@url\@href}%
      \doi 2131 \def\@href#1{\@startlink{#1}\endgroup\@@href}%
\doibase 2132 \def\@@href#1{#1\@@endlink}%
      2133 \providecommand \url [0]{\begingroup\@sanitize@url \@url }%
      2134 \def \@url #1{\endgroup\@href {#1}{\URL@prefix#1}}%
      2135 \providecommand \URL@prefix [0]{URL }%
      2136 \providecommand\doi[0]{\begingroup\@sanitize@url\@doi}%
      2137 \def\@doi#1{\endgroup\@@startlink{\doibase#1}doi:\discretionary {}{}#1\@@endlink }%
      2138 %changes{4.2a}{2017/11/21}{(MD) Use updated best practice to use https and doi.org}%
      2139 \providecommand \doibase [0]{https://doi.org}/%
      2140 \providecommand \@sanitize@url[0]{\chardef\cat@space\the\catcode`\@ \@sanitize\catcode`\cat

```

`\@startlink` How we define `\@startlink` and `\@endlink` will depend on whether we are `\@endlink` running under PDFLATEX. If so, and if PDF output is requested, then we `\pdfstartlink@attr` use its primitives to implement hypertext, breaking out the link attributes in `\hypertext@enable@ltx` `\pdfstartlink@attr` and using the `hyperref` defaults; `\pdfstartlink@attr` can be redefined by a client package. Otherwise we fall back the HyperTeX standard and leave things to the DVI translator.

A class or package that wishes to employ hypertext capabilities should execute the `\hypertext@enable@ltx` procedure.

```

2141 \def\@@startlink#1{}%
2142 \def\@@endlink{}%
2143 \@ifxundefined \pdfoutput {\true@sw}{\@ifnum{\z@=\pdfoutput}{\true@sw}{\false@sw}}}%
2144 {%
2145 \def\@@startlink@hypertext#1{\leavevmode\special{html:<a href="#1">}}}%
2146 \def\@@endlink@hypertext{\special{html:</a>}}}%
2147 }{%
2148 \def\@@startlink@hypertext#1{%
2149 \leavevmode
2150 \pdfstartlink\pdfstartlink@attr
2151   user{/Subtype/Link/A<</Type/Action/S/URI/URI(#1)>>}%
2152 \relax
2153 }%
2154 \def\@@endlink@hypertext{\pdfendlink}%
2155 \def\pdfstartlink@attr{attr{/Border[0 0 1 ]/H/I/C[0 1 1]}}}%
2156 }%
2157 \def\hypertext@enable<ltx{%
2158 \let\@@startlink\@@startlink@hypertext
2159 \let\@@endlink\@@endlink@hypertext
2160 }%

```

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

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

```

2161 \def\href@Hy{\hyper@normalise \href@}%
2162 \def\href@Hy{lt{x}\@ifnextchar\bgroup\Hy@href{\hyper@normalise\href@}}%
2163 \def\Hy@href#{\hyper@normalise\href@}%

```



```

2164 \begingroup
2165 \endlinechar=-1 %
2166 \catcode'\^^A=14 %
2167 \catcode'\^^M\active
2168 \catcode'\%\active
2169 \catcode'\#\active
2170 \catcode'\_\active
2171 \catcode'\$\active
2172 \catcode'\&\active
2173 \gdef\hyper@normalise@ltx{^^A
2174   \begingroup
2175   \catcode'\^^M\active
2176   \def^^M{ }^^A
2177   \catcode'\%\active
2178   \let%\@percentchar
2179   \let%\@percentchar
2180   \catcode'\#\active
2181   \def#{\hyper@hash}^^A
2182   \def\#{\hyper@hash}^^A
2183   \@makeother\&^^A
2184   \edef&{\string&}^^A
2185   \edef&{\string&}^^A
2186   \edef\textunderscore{\string_}^^A
2187   \let_\textunderscore
2188   \catcode'\_\active
2189   \let_\textunderscore
2190   \let~\hyper@tilde
2191   \let~\hyper@tilde
2192   \let\textasciitilde\hyper@tilde
2193   \let\\\@backslashchar
2194   \edef${\string$}^^A
2195   \Hy@safe@activestrue
2196   \hyper@n@rmalise
2197 }^^A
2198 \catcode'\#=6 ^^A
2199 \gdef\Hy@ActiveCarriageReturn@ltx{^^M}^^A
2200 \gdef\hyper@n@rmalise@ltx#1#2{^^A
2201   \def\Hy@tempa{#2}^^A
2202   \ifx\Hy@tempa\Hy@ActiveCarriageReturn
2203     \Hy@ReturnAfterElseFi{^^A
2204       \hyper@@normalise{#1}^^A
2205     }^^A
2206   \else
2207     \Hy@ReturnAfterFi{^^A
2208       \hyper@@normalise{#1}{#2}^^A
2209     }^^A
2210   \fi
2211 }^^A
2212 \gdef\hyper@@normalise@ltx#1#2{^^A
2213   \edef\Hy@tempa{^^A
2214     \endgroup
2215     \noexpand#1{\Hy@RemovePercentCr#2%\^^M\@nil}^^A
2216   }^^A
2217   \Hy@tempa

```

```

2218 }^^A
2219 \gdef\Hy@RemovePercentCr@ltx#1%^M#2\@nil{^^A
2220   #1^^A
2221   \ifx\limits#2\limits
2222   \else
2223     \Hy@ReturnAfterFi{^^A
2224     \Hy@RemovePercentCr #2\@nil
2225   }^^A
2226   \fi
2227 }^^A
2228 \endgroup
2229 \def\switch@hyperref@href{%
2230 \expandafter\@ifx\expandafter{\csname href \endcsname\href@Hy}{
2231 \class@info{Repairing hyperref 6.75r \string\href}%
2232 \let\hyper@normalise\hyper@normalise@ltx
2233 \let\hyper@@normalise\hyper@@normalise@ltx
2234 \let\hyper@n@rmalise\hyper@n@rmalise@ltx
2235 \let\Hy@ActiveCarriageReturn\Hy@ActiveCarriageReturn@ltx
2236 \let\Hy@RemovePercentCr\Hy@RemovePercentCr@ltx
2237 \let\href\href@Hy@ltx
2238 }{}}%
2239 }%
2240 \appdef\document@inithook{\switch@hyperref@href}%

```

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

```

2241 \def\typeout@org#1{%
2242 \begingroup
2243 \set@display@protect
2244 \immediate\write\@unused{#1}%
2245 \endgroup
2246 }%
2247 \long\def\typeout@ltx#1{%
2248 \begingroup
2249 \set@display@protect
2250 \immediate\write\@unused{#1}%
2251 \endgroup
2252 }%
2253 \@ifx{\typeout\typeout@org}{%
2254 \let\typeout\typeout@ltx
2255 \true@sw
2256 }{%
2257 \rvtx@ifformat@geq{2020/10/01}%
2258 {\true@sw}{\false@sw}%
2259 }%
2260 {\class@info{Making \string\typeout\space \string\long}}%
2261 {}%

```

6.20 End of the kernel DOCSTRIP module

Here ends the module.

```
2262 %</kernel>
```

Index

Symbols

\#	2169, 2180, 2182, 2198	
\\$	2171	
\$TEXMF/	3	
\%	912, 2168, 2177, 2179	
\&	2172, 2183, 2185	
\(677	
\)	678	
.aux	38	
.cls	19	
.dtx	4	
.rtx	18	
\@	479, 487	
\@@array	1241, 1307, 1310	
\@@end	374, 467	
\@@endlink	64	
\@@endlink	2132, 2137, 2141	
\@@endlink@hypertext	2146, 2154, 2159	
\@@endpbox	1408, 1431, 1537, 1546	
\@chref	2131, 2132	
\@cpar	807, 983, 1023, 1414, 1438, 1564	
\@startlink	64	
\@startlink	2131, 2137, 2141	
\@startlink@hypertext	2145, 2148, 2158	
\@startpbox	1408, 1430, 1537, 1546	
\@Esphack	900, 902	
\@M	44	
\@MM	26	
\@acol	54	
\@acol	1269, 1278, 1295, 1364, 1375, 1391, 1665, 1666, 1744, 1746, 1844, 1890, 1957, 1960, 1968	
\@acolampacol	1842, 1844, 1852, 1860, 1880, 1888, 1904, 1912, 1956, 1958, 1966, 1970	
\@acoll	1276, 1293, 1373, 1389, 1862, 1914, 1973	
\@acolr	54	
\@acolr	1277, 1294, 1374, 1390, 1695, 1703, 1793, 1796	
\@addamp	1076, 1166, 1817, 1843, 1858, 1886, 1910, 1957, 1961, 1968, 1975	
\@addamp@LaTeX	1076, 1817	
\@addamp@ltx	1166, 1820	
\@addtopreamble	1652, 1748, 1800, 1821, 1827, 1830, 1836, 1839, 1865, 1869, 1873, 1917, 1921, 1925, 1931, 1934, 1937, 1940, 1943, 1947	
\@afterindentfalse	944	
\@afterindenttrue	942	
\@ampacol	1842, 1854, 1882, 1906	
\@argarraycr	1336, 1340	
\@argswap	12	
\@argswap	298	
\@argswap@val	12	
\@argswap@val	298, 586	
\@argtabularcr	1318, 1319	
\@array	46	
\@array	1068, 1158, 1183, 1240, 1241, 1301, 1304, 1397	
\@array@LaTeX	1068, 1397	
\@array@align@	1553	
\@array@align@b	1554	
\@array@align@c	50	
\@array@align@c	1555, 1570	
\@array@align@t	1553	
\@array@align@v	50	
\@array@align@v	1556, 1570	
\@array@array	1183, 1450	
\@array@array@new	1240, 1472	
\@array@ltx	1158, 1417	
\@array@sw	43	
\@array@sw	1067, 1182, 1264	
\@array@sw@LaTeX	1182, 1264	
\@array@sw@array	1067, 1265	
\@arrayacol	1077, 1167, 1249, 1364, 1373, 1374, 1375, 1389, 1390, 1391, 1823	
\@arrayacol@LaTeX	1077, 1823	
\@arrayacol@ltx	1167, 1249, 1826	
\@arrayclassiv	1081, 1171, 1366, 1377, 1936	
\@arrayclassiv@LaTeX	1081, 1936	
\@arrayclassiv@ltx	1171, 1939	

<code>\@arrayclassz</code> .	1079, 1169, 1365, 1376, 1841	<code>\@chclass</code>	1663, 1664, 1679, 1692, 1737, 1742, 1768, 1790
<code>\@arrayclassz@LaTeX</code>	1079, 1841	<code>\@checkend</code> . . .	370, 412, 424, 428
<code>\@arrayclassz@ltx</code> . .	1169, 1850	<code>\@chnum</code> . .	1846, 1864, 1894, 1916
<code>\@arraycr</code>	45	<code>\@classi</code> .	1663, 1682, 1737, 1771
<code>\@arraycr</code>	1192, 1253, 1312, 1367, 1378, 1470, 1496	<code>\@classii</code>	1663, 1684, 1737, 1773
<code>\@arraycr@array</code>	1192, 1324	<code>\@classiii</code>	1663, 1686
<code>\@arraycr@new</code>	1253, 1329	<code>\@classiv</code>	1271, 1280, 1366, 1377, 1664, 1688
<code>\@arrayleft</code>	1463, 1493	<code>\@classv</code> .	1083, 1173, 1664, 1690, 1738, 1778, 1942
<code>\@arrayright</code>	1513, 1517	<code>\@classv@LaTeX</code>	1083, 1942
<code>\@arstrut</code>	51	<code>\@classv@ltx</code>	1173, 1946
<code>\@arstrut</code>	1407, 1423, 1460, 1486, 1538, 1547, 1620, 1622, 1639, 1650	<code>\@classvi</code>	1738, 1780
<code>\@arstrut@hline</code>	51	<code>\@classvii</code>	1739, 1782
<code>\@arstrut@hline</code>	1614	<code>\@classviii</code>	1739, 1784
<code>\@arstrut@hline@cinc</code>	51	<code>\@classx</code> .	1190, 1248, 1740, 1741, 1786, 1788, 1954
<code>\@arstrut@hline@cinc</code>	1629, 1641	<code>\@classx@array</code>	1190, 1954
<code>\@arstrut@hook</code>	51	<code>\@classx@array@new</code> .	1248, 1964
<code>\@arstrut@hook</code>	1600, 1601, 1614, 1650, 1651	<code>\@classz</code> .	1270, 1279, 1365, 1376, 1663, 1680, 1737, 1769
<code>\@arstrut@org</code>	1614	<code>\@clsextension</code>	18, 19
<code>\@arstrutbox</code>	46, 51	<code>\@clsextension</code> . . .	461, 478, 486
<code>\@arstrutbox</code> 1320, 1321, 1344, 1350, 1400, 1444, 1453, 1475	<code>\@ctrerr</code>	936
<code>\@arstrutbox@hline</code>	51	<code>\@currbox</code>	33
<code>\@arstrutbox@hline</code>	1614	<code>\@currbox</code>	852, 855, 857, 858, 864, 865, 874
<code>\@auxout</code>	682, 691, 2055	<code>\@currentHref</code>	627, 1049
<code>\@backslashchar</code>	2193	<code>\@currentcounter</code> .	503, 520, 548
<code>\@boole@def</code>	13	<code>\@currentlabel</code>	502, 519, 654, 692
<code>\@boole@def</code> .	308, 320, 321, 322, 323, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335	<code>\@currentlabelname</code>	37
<code>\@boolean</code>	13	<code>\@currentxt</code>	18
<code>\@boolean</code>	308	<code>\@currentxt</code>	443, 460, 461, 477, 485
<code>\@booleanfalse</code>	13	<code>\@currname</code>	18
<code>\@booleanfalse</code> . . .	318, 815, 860	<code>\@currname</code> . . .	443, 459, 476, 484
<code>\@booleantrue</code>	13	<code>\@currnamestack@ltx</code> .	471, 475, 480, 488, 490
<code>\@booleantrue</code>	318, 745, 858	<code>\@dblarg</code>	952, 953
<code>\@bsphack</code>	690, 905	<code>\@dblfloat</code>	796, 798, 820
<code>\@caption</code>	29	<code>\@dblfloat@LaTeX</code>	796, 820
<code>\@capttype</code>	33	<code>\@defaultsubs</code>	393
<code>\@capttype</code>	761	<code>\@depth</code> . .	1320, 1321, 1345, 1351, 1402, 1446, 1455, 1477, 1632
<code>\@centercr</code>	19, 20	<code>\@dofilelist</code>	386
<code>\@centercr</code>	498	<code>\@doi</code>	2136, 2137
<code>\@centering</code> .	507, 510, 513, 533, 539, 560	<code>\@eha</code>	203
		<code>\@enddocumenthook</code>	369
		<code>\@endpbox</code>	54

<code>\@endpbox</code>	1408, 1431, 1537, 1546, 1658, 1674, 1728, 1759, 1944, 1951	<code>\@ifinner</code>	320, 724, 1561
<code>\@eqcnt</code>	506, 511, 512, 514, 521, 534, 537, 540	<code>\@ifl@t@r</code>	304
<code>\@eqncr</code>	44	<code>\@ifl@ter</code>	463
<code>\@eqnncr</code>	508, 523	<code>\@ifmmode</code>	320, 1372, 1388, 1555, 1557, 1616
<code>\@eqnswtrue</code>	510, 533	<code>\@ifnextchar</code>	569, 578, 593, 1301, 1304, 1307, 1310, 1318, 1319, 1335, 1339, 2162
<code>\@eqnswtrue</code>	504, 521	<code>\@ifnotempty</code>	13
<code>\@eqpen</code>	44	<code>\@ifnotrelax</code>	12
<code>\@esphack</code>	694	<code>\@ifnotrelax</code>	298, 697, 872, 891
<code>\@expast</code>	1659, 1675	<code>\@ifnum</code>	320, 857, 958, 987, 1001, 1986, 2143
<code>\@extension</code>	18	<code>\@ifodd</code>	320, 1988
<code>\@finalstrut</code>	660	<code>\@ifpackageloaded</code>	222, 618, 1057, 1181
<code>\@firstampfalse</code>	1818, 1821, 1844, 1862, 1890, 1914, 1960, 1973	<code>\@ifstar</code>	951, 1316, 1317, 1327, 1332
<code>\@firstamptrue</code>	1654, 1669, 1727, 1756	<code>\@ifundefined</code>	967, 980, 985, 996, 999, 1020, 1025, 1030, 1033, 2045, 2051
<code>\@firstofone</code>	368, 546, 549	<code>\@ifvbox</code>	320
<code>\@float</code>	27, 32, 33	<code>\@ifvmode</code>	320, 563
<code>\@float@LaTeX</code>	794, 795, 819	<code>\@ifvoid</code>	320, 721, 732
<code>\@floatboxreset</code>	767	<code>\@ifx</code>	12, 13
<code>\@flushglue</code>	496, 1643	<code>\@ifx</code>	298, 299, 302, 320, 341, 396, 546, 549, 631, 919, 1068, 1069, 1070, 1071, 1072, 1073, 1074, 1075, 1076, 1077, 1078, 1079, 1080, 1081, 1082, 1083, 1084, 1085, 1086, 1087, 1088, 1183, 1184, 1185, 1186, 1187, 1188, 1189, 1190, 1191, 1192, 1193, 1194, 1195, 1264, 1265, 1991, 2230, 2253
<code>\@font@warning</code>	388	<code>\@ifx@empty</code>	320
<code>\@footnotemark</code>	25	<code>\@ifxundefined</code>	12
<code>\@footnotetext</code>	22, 25	<code>\@ifxundefined</code>	270, 275, 280, 298, 794, 2143
<code>\@footnotetext</code>	599, 619, 620, 635	<code>\@ifxundefined@cs</code>	302, 802, 813, 833
<code>\@gobble@opt@one</code>	59	<code>\@iiparbox</code>	743
<code>\@gobble@opt@one</code>	1997	<code>\@itemlabel</code>	2107
<code>\@halignto</code>	1368, 1379, 1384, 1394, 1406, 1440, 1459, 1504	<code>\@lastchclass</code>	1654, 1664, 1665, 1670, 1692, 1694, 1727, 1742, 1743, 1755, 1790, 1792, 1842, 1851, 1879, 1903, 1955, 1965
<code>\@hang@from</code>	980, 1011	<code>\@latex@warning@no@line</code>	398
<code>\@hang@froms</code>	1020, 1039	<code>\@listdepth</code>	2098, 2101, 2106
<code>\@hangfrom</code>	35–37		
<code>\@hangfrom</code>	1011		
<code>\@hangfrom@section</code>	36		
<code>\@height</code>	1320, 1321, 1401, 1445, 1454, 1476, 1578, 1631, 1638		
<code>\@href</code>	2130, 2131, 2134		
<code>\@ialph</code>	931, 932		
<code>\@if@empty</code>	320, 2088		
<code>\@if@sw</code>	323, 324, 378, 395, 397		
<code>\@ifcat</code>	320		
<code>\@ifdim</code>	320, 387, 877, 976, 1016, 1630, 2083, 2090		
<code>\@ifempty</code>	13		
<code>\@ifeof</code>	320		
<code>\@ifhbox</code>	320, 873, 892		
<code>\@ifhmode</code>	320		

<code>\@mainaux</code>	15	<code>\@percentchar</code>	2178, 2179
<code>\@mainaux</code>	379	<code>\@preamble</code> 1405, 1407, 1415, 1421,	
<code>\@makecol</code>	9	1425, 1440, 1459, 1460, 1470,	
<code>\@makefnmark</code>	22, 26	1482, 1484, 1488, 1504, 1513,	
<code>\@makefnmark</code>	587, 635, 781	1518, 1538, 1548, 1652, 1655,	
<code>\@makefntext</code>	26	1671, 1727, 1754, 1818, 1824,	
<code>\@makefntext</code>	658	1833, 1845, 1892, 1893	
<code>\@makeoother</code>	912, 916, 2183	<code>\@preamerr</code> 1414, 1438, 1666, 1698,	
<code>\@minipagefalse</code>	740	1700, 1747, 1748, 1750, 1798,	
<code>\@mklab</code>	2108	1800, 1804	
<code>\@mkpream</code>	53–55	<code>\@provide</code> . 340, 1059, 1061, 1064	
<code>\@mkpream</code> 1075, 1165, 1189, 1247,		<code>\@pushfilename</code>	474
1404, 1420, 1458, 1481, 1535,		<code>\@pushfilename@ltx</code> ...	454, 474
1543, 1653, 1726		<code>\@refundundefined</code>	394
<code>\@mkpream@LaTeX</code>	1075, 1653	<code>\@reset@options</code>	19
<code>\@mkpream@array</code>	1189, 1726	<code>\@reset@options</code>	457, 472
<code>\@mkpream@array@new</code>	1247, 1753	<code>\@resetactivechars</code>	262
<code>\@mkpream@ltx</code>	1165, 1668	<code>\@rightskip</code>	495
<code>\@mkpream@relax</code>	53, 54	<code>\@runin@to</code>	36
<code>\@mkpream@relax</code> 1718, 1758, 1808		<code>\@runin@to</code>	996, 1012
<code>\@mpargs</code>	743	<code>\@runin@tos</code>	1030, 1040
<code>\@mpfn</code>	23, 26	<code>\@sanitize</code>	2140
<code>\@mpfn</code> ... 572, 576, 585, 598, 604,		<code>\@sanitize@url</code> 2130, 2133, 2136,	
608, 611, 614, 655, 754		2140	
<code>\@mpfootins</code>	29	<code>\@secCNTformat</code>	35, 36
<code>\@mpfootins</code> . 710, 713, 714, 721,		<code>\@secCNTformat</code>	968
722, 725, 728, 732, 733		<code>\@secpenalty</code>	949
<code>\@mpfootnotetext</code>	25	<code>\@sect</code>	35
<code>\@mpfootnotetext</code> . 621, 622, 635		<code>\@sect</code>	956
<code>\@mpmakefntext</code>	746	<code>\@sect@ltx</code>	953, 957
<code>\@multiplelabels</code>	396, 404	<code>\@setckpt</code>	380
<code>\@namedef</code> . 826, 1532, 2045, 2051		<code>\@sharp</code> .. 1411, 1434, 1466, 1499,	
<code>\@nameuse</code> . 219, 749, 1418, 1494,		1536, 1544, 1657, 1673, 1708,	
2046, 2072		1714, 1728, 1748, 1757, 1800,	
<code>\@ne</code>	11, 63	1847, 1848, 1866, 1870, 1874,	
<code>\@newl@bel</code>	381	1895, 1897, 1899, 1918, 1922,	
<code>\@nextchar</code> 1661, 1662, 1676, 1678,		1926, 1944, 1951	
1734, 1765, 1931, 1934, 1937,		<code>\@sssect</code>	35, 37
1940, 1943, 1950		<code>\@sssect</code>	1013
<code>\@nmbrlistfalse</code>	2109	<code>\@sssect@ltx</code>	952, 1013
<code>\@nobreakfalse</code>	2007	<code>\@startcolumn</code>	9
<code>\@normalcr</code>	19	<code>\@startpbox</code>	54
<code>\@normalcr</code>	492	<code>\@startpbox</code>	
<code>\@nul</code>	917, 928	... 1408, 1430, 1537, 1546,	
<code>\@onefilewithoptions</code>	18	1658, 1674, 1724, 1728, 1759,	
<code>\@outputdblcol</code>	9	1943, 1949	
<code>\@p@pfilename</code>	474	<code>\@startsection</code>	35, 37
<code>\@p@pfilename@ltx</code>	471, 483	<code>\@startsection</code>	937
<code>\@parboxrestore</code>	653, 766	<code>\@startsection@hook</code> ..	938, 955
<code>\@pass@options</code>	449	<code>\@starttoc</code>	60

<code>\@starttoc</code>	1998	<code>\@tempcntb</code>	894, 1978, 1981, 1982, 1985, 1987, 1988
<code>\@svsec</code>	36	<code>\@tempdima</code>	62
<code>\@svsec</code>	960, 966, 968, 971, 981, 997	<code>\@tempskipa</code>	941, 943, 944, 949, 975, 976, 1015, 1016
<code>\@svsechd</code>	994, 1028	<code>\@temptokena</code>	1729, 1735, 1760, 1765
<code>\@tabacol</code>	1078, 1168, 1250, 1269, 1278, 1295, 1829	<code>\@testdef</code>	381
<code>\@tabacol@LaTeX</code>	1078, 1832	<code>\@testpach</code>	1662, 1678, 1736, 1767
<code>\@tabacol@ltx</code>	1168, 1250, 1835	<code>\@testtrue</code>	1982, 1988
<code>\@tabacoll</code>	1276, 1293, 1829	<code>\@textsuperscript</code>	637
<code>\@tabacolr</code>	1277, 1294, 1838	<code>\@tfor</code>	1660, 1676, 1734, 1765
<code>\@tabarray</code>	44	<code>\@thefnmark</code>	605, 612, 638, 655
<code>\@tabarray</code>	1071, 1161, 1185, 1243, 1273, 1282, 1290, 1298, 1300, 1369, 1380, 1385, 1395	<code>\@toodeep</code>	2099
<code>\@tabarray@LaTeX</code>	1071, 1300	<code>\@totalleftmargin</code>	2118, 2121, 2125, 2127
<code>\@tabarray@array</code>	1185, 1306	<code>\@tpfootnotetext</code>	635
<code>\@tabarray@array@new</code>	1243, 1309	<code>\@trivlist</code>	2111
<code>\@tabarray@ltx</code>	1161, 1303	<code>\@twopowerfourteen</code>	33
<code>\@tabclassiv</code>	1080, 1170, 1271, 1280, 1930	<code>\@twopowerfourteen</code>	848, 876
<code>\@tabclassiv@LaTeX</code>	1080, 1930	<code>\@twopowertwo</code>	33
<code>\@tabclassiv@ltx</code>	1170, 1933	<code>\@twopowertwo</code>	849, 893, 894
<code>\@tabclassz</code>	1082, 1172, 1270, 1279, 1878	<code>\@undefined</code>	43
<code>\@tabclassz@LaTeX</code>	1082, 1878	<code>\@undefined</code>	32, 470
<code>\@tabclassz@ltx</code>	1172, 1902	<code>\@unexpandable@protect</code>	1656, 1672
<code>\@tabular</code>	43	<code>\@unused</code>	2244, 2250
<code>\@tabular</code>	1063, 1070, 1160, 1184, 1242, 1266	<code>\@unusedoptionlist</code>	449, 450
<code>\@tabular@LaTeX</code>	1070, 1266	<code>\@url</code>	2133, 2134
<code>\@tabular@array</code>	1184, 1284	<code>\@whiles</code>	1730, 1761
<code>\@tabular@array@new</code>	1242, 1292	<code>\@width</code>	1320, 1321, 1345, 1351, 1403, 1447, 1456, 1478, 1633
<code>\@tabular@ltx</code>	1160, 1275	<code>\@write@floatline</code>	34
<code>\@tabularcr</code>	44	<code>\@write@floatline</code>	915
<code>\@tabularcr</code>	1085, 1175, 1264, 1272, 1281, 1312, 1312	<code>\@writefile</code>	682
<code>\@tabularcr@LaTeX</code>	44	<code>\@x@sf</code>	583, 590
<code>\@tabularcr@LaTeX</code>	1085, 1316	<code>\@xargarraycr</code>	44, 45
<code>\@tabularcr@ltx</code>	1175, 1317	<code>\@xargarraycr</code>	1087, 1177, 1194, 1255, 1312
<code>\@tbpen</code>	44	<code>\@xargarraycr@LaTeX</code>	1087, 1320
<code>\@tbpen</code>	1312	<code>\@xargarraycr@array</code>	1194, 1342
<code>\@temp@sw</code>	858, 860, 864	<code>\@xargarraycr@ltx</code>	1177, 1321
<code>\@tempa</code>	223, 225, 870, 872, 873, 878, 879, 881, 882, 889, 891, 892, 893, 894, 918, 919	<code>\@xargarraycr@new</code>	1255, 1348
<code>\@tempb</code>	909, 919	<code>\@xarraycr</code>	45
<code>\@tempboxa</code>	62	<code>\@xarraycr</code>	1193, 1254, 1312
<code>\@tempc</code>	907, 925	<code>\@xarraycr@array</code>	1193, 1334
<code>\@tempcnta</code>	33	<code>\@xarraycr@new</code>	1254, 1338
		<code>\@xbitor</code>	1978
		<code>\@xbitor@LaTeX</code>	1978, 1991
		<code>\@xbitor@ltx</code>	1984, 1996

<code>\xfloat</code>	746
<code>\xfloat@LaTeX</code>	746, 750
<code>\xfloat@anchored</code>	750, 760
<code>\xfloat@prep</code>	748, 752
<code>\xfootnotemark</code>	22
<code>\xfootnotenext</code>	25
<code>\xhline</code>	1579, 1585, 1586
<code>\xhline@unneeded</code>	1588
<code>\xifempty</code>	13
<code>\xssect</code>	1009, 1037
<code>\xtabularcr</code>	44
<code>\xtabularcr</code>	1086, 1176, 1312
<code>\xtabularcr@LaTeX</code>	1086, 1318
<code>\xtabularcr@ltx</code>	1176, 1319
<code>\xvi</code>	847, 857, 858
<code>\xxxii</code>	853, 854, 856
<code>\yargarraycr</code>	44, 45
<code>\yargarraycr</code>	1088, 1178, 1195, 1256, 1312
<code>\yargarraycr@LaTeX</code>	1088, 1322
<code>\yargarraycr@array</code>	1195, 1355
<code>\yargarraycr@ltx</code>	1178, 1323
<code>\yargarraycr@new</code>	1256, 1359
<code>\[</code>	22
<code>\[</code>	916
<code>\]</code>	19, 20, 44
<code>\{</code>	133, 912, 916
<code>\}</code>	133, 912, 916
<code>_</code>	41, 47, 67, 70, 71, 75, 92, 94, 100, 105, 112, 223, 2140
<code>\]</code>	22
<code>\]</code>	916
<code>\^</code>	911, 924, 2166, 2167, 2175
<code>_</code>	52, 2170, 2187, 2188
<code>\~</code>	2191
A	
<code>\abovedisplayshortskip</code>	529
<code>\abovedisplayskip</code>	524, 525, 526, 527, 528, 529
<code>\active</code>	911, 2167, 2168, 2169, 2170, 2171, 2172, 2175, 2177, 2180, 2188
<code>\addcontentsline</code>	37
<code>\addcontentsline</code>	684, 986, 1000, 1026, 1034
<code>\addtocontents</code>	672, 685
<code>\AddToHook</code>	346, 410
<code>\aftergroup</code>	49
<code>\aftergroup</code>	1555, 1562, 2076
<code>\AfterLastShipout</code>	15, 17
<code>\appdef</code>	12, 15, 17, 28
<code>\appdef</code>	262, 269, 355, 438, 616, 702, 706, 1052, 1056, 1315, 1425, 1488, 1626, 1650, 1652, 1808, 2240
<code>\appdef@e</code>	287
<code>\appdef@eval</code>	292
<code>\appdef@val</code>	284, 293
<code>aps.rtx</code>	18
argument	
class	18
float	34
text	23, 38
your_tokens_here	15
argument, optional	
number	23
<code>\array</code>	45
<code>\array</code>	1072, 1162, 1186, 1244, 1363
array document class	9, 39, 43, 45, 53, 55
array environment	2, 39, 44
<code>\array@array</code>	1186, 1382
<code>\array@array@new</code>	1244, 1387
<code>\array@default</code>	818, 1304, 1310, 1553
<code>\array@hook</code>	1062
<code>\array@LaTeX</code>	1072, 1363
<code>\array@ltx</code>	1162, 1371
<code>\array@message</code>	1181, 1258
<code>\array@row@pre</code>	1428, 1491, 1569
<code>\array@row@pst</code>	1510, 1516, 1569
<code>\array@row@rst</code>	50
<code>\array@row@rst</code>	1569, 1713, 1722, 1866, 1870, 1874, 1918, 1922, 1926, 1951
<code>\arraycolsep</code>	511, 512, 1383, 1824, 1827
<code>\arrayrulewidth</code>	1578, 1592, 1638
<code>\arraystretch</code>	1401, 1402, 1445, 1446, 1454, 1455, 1476, 1477, 1630, 1632
<code>\AtBeginDocument</code>	14, 15, 27, 28, 44
<code>\AtBeginDocument</code>	356
<code>\AtEndDocument</code>	15

`\AtEndDocument` 358, 368
`\AtEndOfClass` 18
`atveryend` document class 17, 85
`\author` 50

B

`\baselineskip` 11, 21
`\baselineskip` . 1413, 1436, 1468, 1501
`\baselineskip` 11
`\begin` 14
`\begin@float@pagebreak` ... 762, 776, 808, 834
`\belowdisplayshortskip` ... 528
`\belowdisplayskip` 527
`\bgroup` 25, 50
`\body@font` 1575, 1610
`\botrule` 50
`\botrule` 1574
`\box` 62

C

`\c@mpfootnote` 629, 630, 756
`\c@secnumdepth` .. 958, 987, 1001
`\Call@AfterLastShipout` 17
`\Call@AfterLastShipout` 435, 439
`\caption` 29
`\caption` 706
`\cat@letter` 241
`\cat@space` 2140
`\catcode` 18
`\catcode` 479, 487, 911, 916, 2140, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2175, 2177, 2180, 2188, 2198
`\cell@fil` 1605, 1648, 1922, 1926
`\cell@font` 1604, 1611, 1713, 1723, 1918, 1922, 1926, 1951
`\centering` 19
`\changes` . 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188
`\chardef` . 240, 241, 242, 847, 907, 2140

`\check@aux` 17
`\check@aux` 363
`<class>` placeholder 18
`class`, argument 18
`\class@documenthook` ... 15, 31
`\class@documenthook` . 355, 438, 616, 706
`\class@enddocumenthook` . 15, 28
`\class@enddocumenthook` 231, 355
`\class@err` 203
`\class@ext@hook` 441
`\class@extension` 441
`\class@extensionfile` 18
`\class@extensionfile` 441
`\class@info` 203, 464, 466, 553, 632, 863, 895, 956, 1154, 1156, 1236, 1259, 1262, 1992, 1994, 2231, 2260
`\class@name` 10
`\class@name` . 203, 204, 205, 224, 233, 234
`\class@warn` 203, 235
`\class@warn@end` .. 207, 212, 217, 230
`<class customization commands>` placeholder 9
`\ClassError` 203, 224
`classes.dtx` 27
`classes.dtx` document class .. 32
`\ClassInfo` 205
`\classname` 61, 69, 121, 123, 125, 140, 164, 165, 177, 178, 180, 181, 185
`\classoption` 167
`\ClassWarningNoLine` 204
`\clear@document` 15–17
`\clear@document` .. 371, 429, 432
`\clearpage` 16, 17
`\clearpage` 425, 433
`\closeout` 379, 828
`\cmd` 133
`\col@sep` 43
`\col@sep` 1287, 1383
`\color` 1812
`\color@begingroup` 657
`\color@endgroup` 662, 741
`colortbl` document class . 43, 44, 54, 55, 84, 85
`\colortbl@message` .. 1181, 1261
`\colrule` 50

<code>\colrule</code>	1574	<code>\deadcycles</code>	373
<code>\column@fil</code>	1574, 1609	<code>\DeclareRobustCommand</code>	19
<code>\column@font</code>	1574, 1608	<code>\DeclareRobustCommand</code>	498
<code>\columnwidth</code>	25	<code>\detokenize</code>	414, 415
<code>\columnwidth</code>	765	<code>\dimen</code>	60
<code>\contentsline</code>	28, 59	<code>\dimen@</code>	875, 876, 878, 879, 881, 882, 1629, 1630, 1631, 2067, 2068
<code>\contentsline</code>	686, 703, 704	<code>\dimen@iii</code>	238
<code>\contentsline@latex</code>	700, 703	<code>\dimendef</code>	238
<code>\copy</code>	1616	<code>\discretionary</code>	2137
<code>\copyright</code>	47	<code>displaymath</code> environment	22
<code>\count</code>	33	<code>\displaystyle</code>	510, 513, 533, 539
<code>\count</code>	852, 855, 857, 858, 864, 1978, 1985	<code>\displaywidth</code>	509
<code>\count@</code>	1709, 1715, 1731, 1762	<code>\do</code>	1661, 1677, 1735, 1766
<code>\count@i</code>	267	<code>\do@check@aux</code>	16, 17
<code>\count@ii</code>	267	<code>\do@check@aux</code>	363
<code>\countdef</code>	267, 268	<code>\do@if@floats</code>	27, 31
<code>\crcr</code>	1507, 1510, 1513, 1516, 1521, 1597	<code>\do@if@floats</code>	784
<code>\cs</code>	137, 138, 140, 151, 152, 153, 155, 156, 158, 159, 160, 161, 162, 163, 166, 168, 175, 177, 180, 181, 182, 188	<code>\do@output@cclv</code>	434
<code>\csname</code>	12, 46	<code>doc</code>	4
<code>\csname</code>	32, 33, 245, 247, 251, 253, 302, 323, 443, 445, 451, 572, 576, 604, 608, 614, 626, 627, 655, 697, 698, 787, 789, 792, 793, 824, 825, 829, 838, 842, 844, 871, 890, 907, 910, 972, 980, 985, 990, 996, 999, 1004, 1020, 1025, 1030, 1033, 1252, 2004, 2005, 2044, 2047, 2050, 2052, 2056, 2057, 2065, 2066, 2067, 2076, 2106, 2230	<code>doc/</code>	3
<code>\CT@cell@color</code>	1815	<code>\DocInput</code>	8
<code>\CT@color</code>	1810	<code>\document</code>	14
<code>\CT@column@color</code>	1813	<code>\document</code>	344
<code>\CT@do@color</code>	1811	<code>document</code> class	
<code>\CT@end</code>	44	<code>array</code>	9, 39, 43, 45, 53, 55
<code>\CT@row@color</code>	1814	<code>atveryend</code>	17, 85
<code>\CT@setup</code>	1809	<code>classes.dtx</code>	32
<code>\CT@start</code>	44	<code>colortbl</code>	43, 44, 54, 55, 84, 85
<code>\CurrentOption</code>	456	<code>ftnright</code>	9
		<code>hyperref</code>	20, 22, 24, 25, 28, 29, 37, 38, 59, 62, 64, 83–85
		<code>hyperref_Lpackage</code>	2, 28
		<code>hyperref.sty</code>	86
		<code>lineno.sty</code>	62, 85, 86
		<code>longtable</code>	9, 53
		<code>ltxdoc</code>	5, 8
		<code>ltxfront</code>	26
		<code>ltxgrid</code>	9, 15, 17, 25
		<code>ltxutil</code>	1, 9, 17
		<code>ltxutil.dtx</code>	3
		<code>ltxutil.pdf</code>	3
		<code>ltxutil.sty</code>	3
		<code>revtex4</code>	17
		<code>textcase</code>	19, 85
		<code>document</code> class option	
		<code>hypertext</code>	85
		<code>lengthcheck</code>	84, 85
		<code>document</code> environment	5, 15

`\document@inithook` 15
`\document@inithook` 346, 350, [354](#),
 355, 702, 1052, 1056, 2240
`\doi` 64
`\doi` [2130](#)
`\doibase` [2130](#)
`\doublerulesep` 1591
`\dp` 669, 865, 881, 882,
 894, 1320, 1321, 1344, 1350,
 1402, 1446, 1455, 1477, 1632

E

`\edef` 53
`\edef` 272, 277, 282, 414, 415,
 583, 1405, 1818, 1824, 1833,
 1845, 1892, 2047, 2184, 2185,
 2186, 2194, 2213
`\emph` 92
`\end` 34
`\end@float@anchored` . 763, 764,
 770
`\end@float@pagebreak` 774, 777,
 808
`\endarray` 44, 48
`\endarray` [1056](#), [1073](#), [1163](#), [1187](#),
 1245, [1506](#), 1524, 1527, 1530
`\endarray@array` 1187, 1512
`\endarray@array@new` 1245, 1515
`\endarray@hook` 1060, 1061
`\endarray@LaTeX` 1073, 1506
`\endarray@ltx` 48
`\endarray@ltx` .. 1163, 1509, 1516
`\endcsname` 32, 33, 245, 247,
 251, 253, 302, 323, 443, 445,
 451, 572, 576, 604, 608, 614,
 626, 627, 655, 697, 698, 787,
 789, 792, 793, 824, 825, 829,
 838, 842, 844, 871, 890, 907,
 910, 972, 980, 985, 990, 996,
 999, 1004, 1020, 1025, 1030,
 1033, 1252, 2004, 2005, 2044,
 2047, 2050, 2052, 2056, 2057,
 2065, 2066, 2067, 2077, 2106,
 2230
`\enddocument` 16, 17
`\enddocument` [363](#)
`\endgroup` 14, 17
`\endlinechar` 2165
`\endminipage` 736

`\endtabular`
 .. [1056](#), 1074, 1164, 1188,
 1246, 1252, [1520](#), 1532
`endtabular environment` 49
`\endtabular*` [1532](#)
`\endtabular@array` .. 1188, 1526
`\endtabular@array@new` ... 1246,
 1529
`\endtabular@hook` ... 1058, 1059
`\endtabular@LaTeX` .. 1074, 1520
`\endtabular@ltx` 1164, 1523
`\endwrite@float` 900, 910
`\endwrite@floats` 902
`environment`
 `array` 2, 39, 44
 `displaymath` 22
 `document` 5, 15
 `endtabular` 49
 `eqnarray` 20, 21, 44
 `equation` 21
 `figure` 27
 `figure@write` 27, 31
 `longtable` 27, 34
 `minipage` 22, 27
 `tabular` 2, 39, 44
`\eqnarray` 20
`\eqnarray` 546, 549, 554, 567
`eqnarray environment` . 20, 21, 44
`\eqnarray@fleqn@fixed` [499](#)
`\eqnarray@LaTeX` [499](#)
`\eqncolsep` ... 535, 538, 555, 556
`\equation` 22
`\equation` 566
`equation environment` 21
`\errorstopmode` 259
`\extrarowheight` 1452, 1474

F

`\f@ur` [240](#), 1755
`\false@sw` ... 313, 319, 324, [336](#),
 341, 1091, 1094, 1097, 1100,
 1103, 1106, 1109, 1112, 1115,
 1118, 1121, 1124, 1127, 1130,
 1133, 1136, 1139, 1142, 1145,
 1148, 1151, 1198, 1201, 1204,
 1207, 1210, 1213, 1216, 1219,
 1222, 1225, 1228, 1231, 1234,
 2143, 2258
`figure environment` 27
`figure@write environment` 27, 31

<code>\Hy@ActiveCarriageReturn@ltx</code>	310, 322, 1398, 1464
.....	2199, 2235
<code>\Hy@footnote@currentHref</code>	586, 627
<code>\Hy@href</code>	2162, 2163
<code>\Hy@raisedlink</code>	1044
<code>\Hy@RemovePercentCr</code>	2215, 2224, 2236
<code>\Hy@RemovePercentCr@ltx</code>	2219, 2236
<code>\Hy@ReturnAfterElseFi</code>	2203
<code>\Hy@ReturnAfterFi</code>	2207, 2223
<code>\Hy@safe@activetrue</code>	2195
<code>\Hy@tempa</code>	2201, 2202, 2213, 2217
<code>\hyper@normalise</code>	2204, 2208, 2233
<code>\hyper@normalise@ltx</code>	2212, 2233
<code>\hyper@anchorend</code>	38
<code>\hyper@anchorend</code>	1046
<code>\hyper@anchorstart</code>	38
<code>\hyper@anchorstart</code>	1045
<code>\hyper@hash</code>	2181, 2182
<code>\hyper@last</code>	38
<code>\hyper@linkend</code>	38
<code>\hyper@linkend</code>	588, 1048
<code>\hyper@linkstart</code>	38
<code>\hyper@linkstart</code>	586, 1047
<code>\hyper@makecurrent</code>	1043
<code>\hyper@normalise</code>	2196, 2234
<code>\hyper@normalise@ltx</code>	2200, 2234
<code>\hyper@normalise</code>	2161, 2162, 2163, 2232
<code>\hyper@normalise@ltx</code>	2173, 2232
<code>\hyper@tilde</code>	2190, 2191, 2192
<code>\hyperanchor</code>	38
<code>hyperref</code> document class	20, 22, 24, 25, 28, 29, 37, 38, 59, 62, 64, 83–85
<code>hyperref</code> _L package document class 2, 28
<code>hyperref.sty</code> document class	86
<code>hypertext</code> document class option	85
<code>\hypertext@enable@ltx</code>	64
<code>\hypertext@enable@ltx</code>	2141
I	
<code>\ialign</code>	46
<code>\ialign</code>	1406, 1459
<code>\if</code>	46
<code>\if</code>	310, 322, 1398, 1464
<code>\if@filesw</code>	378, 395, 2003, 2032
<code>\if@firstamp</code>	1818, 1821
<code>\if@nobreak</code>	946
<code>\if@noskipsec</code>	939
<code>\ifeof</code>	326
<code>\iffalse</code>	54, 1326, 1331
<code>\IfFileExists</code>	442
<code>\IfFormatAtLeastTF</code>	303, 306
<code>\ifhbox</code>	327
<code>\ifinner</code>	329
<code>\ifmmode</code>	330
<code>\ifodd</code>	332, 1982
<code>\ifvbox</code>	333
<code>\ifvoid</code>	335
<code>\ignorespaces</code> 659, 1538, 1551, 1708, 1714, 1895, 1897, 1899, 1918, 1922, 1926, 1943, 1951, 2073, 2082, 2094, 2114
<code>\immediate</code>	379, 790, 827, 925, 2005, 2055, 2244, 2250
<code>\incompatible@package</code>	221
<code>\index</code>	674
<code>\init@hyperref</code>	1041
<code>\insert</code>	26
<code>\insert</code>	643, 733
<code>\insert@column</code>	1191, 1251, 1706
<code>\insert@column@array</code>	1191, 1706
<code>\insert@column@array@new</code>	1251, 1711
<code>\interdisplaylinepenalty</code>	44
<code>\interfootnotelinepenalty</code>	667
<code>\interlinepenalty</code>	667, 978, 1018
<code>\interlineskip</code>	26
<code>\intertabularlinepenalty</code>	44
<code>\intertabularlinepenalty</code>	1312, 1313, 1315, 1317, 1332
<code>\intertextsep</code>	776, 777
<code>\item</code>	103, 113, 116, 120, 122, 124, 129, 131
<code>\itemindent</code>	2105
<code>\itshape</code>	629, 638
J	
<code>\j@nk</code>	342
K	
<code>kernel</code>	2, 10, 66

L

`\l@` 60
`\l@sections` 60
`\l@sections` 2061
`\l@section` 2015
`\label` 673, 684
`\lastbox` 563, 726, 832
`\LaTeX` . 41, 70, 71, 81, 91, 92, 94, 105, 109, 112, 114, 188
`latex/` 3
`\LaTeXe` 75, 100, 198
`\leftmargin` ... 2118, 2120, 2125
`\leftmargini` 2015
`\leftmarginii` 2016
`\leftmarginiii` 2017
`\leftmarginiv` 2018
`\leftmarginv` 2019
`\leftmarginvi` 2020
`\leftskip` 63
`\leftskip` . 493, 2066, 2068, 2127
`lengthcheck` document class option 84, 85
`\lengthcheck@sw` 810
`\let` 43, 61
`\let@environment` 11
`\let@environment` . 243, 800, 801, 803, 822, 823
`\limits` 2221
`lineno.sty` document class 62, 85, 86
`\lineskip` 1413, 1436, 1467, 1501
`\linewidth` 532, 2119, 2120, 2121
`\list` 63
`\list` 2097
`\listparindent` . 2104, 2117, 2124
`\LoadClass` 18
`\long` 66
`longtable` document class . 9, 53
`longtable` environment .. 27, 34
`\loop` 14
`\loopuntil` 338
`\loopwhile` 338
`\lrstrut` 1575, 1576, 1613
`\LT@array` 9
`ltxfloat.dtx` 22
`\ltx@contentsline` 684
`\ltx@def@footproc` 23, 24
`\ltx@def@footproc` 571, 579, 594, 602
`\ltx@footmark` 23

`\ltx@footmark` 569
`\ltx@footnote@pop` 753, 759
`\ltx@foottext` 23
`\ltx@foottext` 569
`\ltx@make@current@footnote` 569
`\ltx@mathindent` 21
`\ltx@mathindent` 522, 560
`\ltx@stp@footproc` 23, 24
`\ltx@stp@footproc` 575, 580, 595, 610
`\ltx@thempfootnote` ... 630, 633
`\ltx@xfootmark` 578, 579
`\ltx@xfootnote` 23
`\ltx@xfootnote` 569
`\ltx@xfoottext` 593, 594
`\ltx@yfootmark` 578, 580
`\ltx@yfootnote` 569, 574
`\ltx@yfoottext` 593, 595
`ltxdoc` document class 5, 8
`ltxfront` document class 26
`ltxgrid` document class 9, 15, 17, 25
`\ltxu@dotsep` 2014, 2089
`ltxutil` document class . 1, 9, 17
`ltxutil.dtx` 2, 3
`ltxutil.dtx` document class ... 3
`ltxutil.pdf` 1
`ltxutil.pdf` document class ... 3
`ltxutil.sty` 1–3
`ltxutil.sty` document class ... 3

M

`\make@footnotetext` 25, 26
`\make@footnotetext` 635
`\makeatletter` ... 383, 455, 2001
`makeindex` 3
`\makelabel` 2108
`\maketitle` 25
`\maketitle` 59
`\mark` 28
`\mathchardef` 848, 849
`\mathindent` 21
`\meaning` 264
`\MessageBreak` 389
`minipage` environment ... 22, 27
`\minipagefootnote@drop` 25
`\minipagefootnote@drop` 663, 709
`\minipagefootnote@foot` ... 709
`\minipagefootnote@here` 29

<code>\minipagefootnote@here</code> ...	707, 709 , 771
<code>\minipagefootnote@init</code>	709 , 768
<code>\minipagefootnote@pick</code>	647, 709
<code>\mpfootnote</code>	632
<code>\multicolumn</code> ..	1069 , 1159 , 1533
<code>\multicolumn@LaTeX</code> ..	1069 , 1533
<code>\multicolumn@ltx</code> ...	1159 , 1540
<code>\multiply</code>	854
<code>\multispan</code>	1534 , 1541
N	
<code>nameref.sty</code>	37
<code>\NC@list</code>	1730 , 1761
<code>\newbox</code>	1625
<code>\newcount</code>	1312 , 1314
<code>\newenvironment</code>	34
<code>\newlabel</code>	28
<code>\newlabel</code>	692
<code>\newlinechar</code>	924
<code>\newwrite</code>	786, 2004
<code>\noalign</code> ..	1319 , 1321 , 1322 , 1323 , 1340 , 1353 , 1357 , 1361 , 1576 , 1578 , 1582 , 1586 , 1598
<code>\noexpand</code> ..	264, 265, 1406, 1459, 2215
<code>\normalfont</code>	638
<code>number, optional argument</code> ...	23
<code>\numberline</code>	62
<code>\numberline@@sections</code> ...	2071 , 2085
O	
<code>\obeylines</code>	921
<code>\obsolete@command</code>	206
<code>\oneapage</code>	808
<code>\openout</code>	791, 2005
<code>\OptionNotUsed</code>	19
<code>\other</code>	242
P	
<code>\p@equation</code>	502, 519
<code>package</code>	2, 10
<code>\package@name</code>	196 , 197
<code>\PackageInfo</code>	197
<code>\par</code>	62, 66
<code>\parindent</code>	21, 63
<code>\parindent</code>	779 , 2068 , 2086 , 2117 , 2124
<code>\parsep</code>	2112
<code>\parshape</code>	63
<code>\parshape</code>	2121
<code>\parskip</code>	21
<code>\parskip</code>	526, 2112
<code>\parskipfillskip</code>	21
<code>\partopsep</code>	525
<code>\pbox@hook</code>	1649, 1951
<code>\pdfendlink</code>	2154
<code>pdfmark.def</code>	37
<code>\pdfoutput</code>	2143
<code>\pdfstartlink</code>	2150
<code>\pdfstartlink@attr</code>	64
<code>\pdfstartlink@attr</code>	2141
<code>\pdfstringdefDisableCommands</code>	28
<code>\phantomsection</code>	37
<code>\phantomsection</code> ..	959 , 964 , 1014 , 1042
<code>placeholder</code> <code><class></code>	18
<code><class customization commands></code>	9
<code><hooks and bookkeeping></code> ..	16
<code><read main .aux and final checks></code>	16
<code><your document here></code>	9
<code>\prep@math</code>	21
<code>\prep@math</code>	562
<code>\prep@math@patch</code>	21, 22
<code>\prep@math@patch</code>	562
<code>\prepdef</code>	12
<code>\prepdef</code>	269 , 348 , 566 , 567 , 707 , 824 , 825 , 968 , 970 , 1058 , 1060 , 1063 , 1421 , 1484
<code>\prepnext@tok</code>	1733 , 1764
<code>\prevdepth</code>	11
<code>\prevdepth</code>	266
<code>\print@float</code>	32
<code>\print@float</code>	807
<code>\printfigures</code>	28
<code>\ProcessOptions</code>	19
<code>\protected</code>	412
<code>\protected@edef</code>	24
<code>\protected@edef</code> ..	627 , 654 , 966
<code>\protected@write</code>	682 , 691
<code>\protected@xdef</code>	612
<code>\providecommand</code>
...	439 , 1042 , 1043 , 1044 , 1045 , 1046 , 1047 , 1048 , 1049 ,

1062, 2130, 2133, 2135, 2136, 2139, 2140	
\ProvidesFile	6
\ProvidesPackage	19
R	
\raggedleft	19
\raggedright	19
<i><read main .aux and final checks></i> place- holder	16
README-LTXUTIL	3, 4
\realfootnote	25
\RecordChanges	36
\refstepcounter	36
\relax	36, 38
\repeat	14
\replace@command	211
\replace@environment	216
\RequirePackage	25, 27, 33
\reserved@a .. 1579, 1585, 1589, 1590, 1661, 1676	
\reset@font	666
revtex/	3
revtex4 document class	17
\robustify@contents	672
\romannumeral	2106
\rule	659
\rvtx@enddocument@patch ..	409
\rvtx@enddocument@patch@end	418, 425
\rvtx@enddocument@patch@more	420, 423
\rvtx@ifformat@geq	13
\rvtx@ifformat@geq .. 303, 344, 363, 409, 2257	
\rvtx@tmpa	499, 545, 548
S	
\samepage	44
\samepage	1315
\save@decl	1738, 1775
\say	11
\say	264, 1589
\saythe	11
\saythe	264
\sc	30, 91
\sec@upcase	38
\sec@upcase	1055
\section	87, 835
\Sectionformat	37
\set@arstrutbox	51
\set@arstrutbox 1419, 1443, 1614	
\set@display@protect 2243, 2249	
\set@eqnarray@skips	561
\set@footnotefont	27
\set@footnotefont 651, 665	
\set@footnotewidth 25, 27	
\set@footnotewidth	635
\set@footnotewidth@one ...	671
\set@listindent 2113, 2129	
\set@listindent@	2123
\set@listindent@parshape 2116, 2129	
\set@tocdim@pagenum	62
\set@tocdim@pagenum 2064, 2081	
\set@typeset@protect ... 1412, 1435, 1500, 1536, 1545	
\shipout	11
\show@box@size@sw	862
\showboxbreadth	259
\showboxdepth	259
\showoutput	262
\skip@	2070, 2073
source/	3
\spacefactor	583, 590
\special	2145, 2146
\splitmaxdepth	669
\splittopskip	668
\stepcounter 501, 519, 611	
\stop	225
\StopEventually	5
\string	207, 212, 223, 632, 677, 678, 679, 682, 692, 956, 1992, 1994, 2056, 2184, 2185, 2186, 2194, 2231, 2260
\strutbox	660, 669, 1401, 1402, 1445, 1446, 1451, 1455, 1473, 1477, 1630, 1632
\subsection	97, 136
\substyle@ext	18
\switch@array	39, 45
\switch@array	1057, 1066
\switch@array@info . 1181, 1238	
\switch@hyperref@href ... 2229, 2240	
\switch@tabular	39, 44
\switch@tabular 1057, 1066	

T

<code>\t@</code>	237 , 239
<code>\tab@rule</code>	51
<code>\tab@rule</code>	1574 , 1575 , 1596
<code>\tabcolsep</code>	1287 , 1645 , 1646 , 1647 , 1833
<code>\tableft@skip</code> ..	1437 , 1502 , 1642
<code>\tableftsep</code> ...	1645 , 1719 , 1830
<code>\tableofcontents</code>	85
<code>\tabmid@skip</code> ..	1422 , 1485 , 1643
<code>\tabmidsep</code>	1646 , 1720 , 1836
<code>\tabright@skip</code> .	1426 , 1489 , 1644
<code>\tabrightsep</code> ..	1647 , 1721 , 1839
<code>\tabskip</code>	21 , 46
<code>\tabskip</code>	507 , 510 , 513 , 515 , 522 , 533 , 539 , 541 , 1407 , 1422 , 1426 , 1437 , 1461 , 1485 , 1489 , 1502
<code>\tabular</code>	39
<code>\tabular environment</code> ..	2 , 39 , 44
<code>\tabular@hook</code>	1063 , 1064
<code>\tabularnewline</code>	1409 , 1432 , 1470 , 1497
<code>\tally@float</code>	33
<code>\tally@float</code>	847
<code>\TeX</code>	67 , 75 , 114 , 185
<code>tex/</code>	3
<code>texmf-local/</code>	3
<code>text, argument</code>	23 , 38
<code>\textasciitilde</code>	2192
<code>textcase document class</code> .	19 , 85
<code>\texttt</code> ..	52 , 110 , 111 , 118 , 127 , 130 , 133 , 137
<code>\textunderscore</code>	2186 , 2187 , 2189
<code>\textwidth</code>	877
<code>\thanks</code>	42 , 46 , 52
<code>\the@toks</code>	1707 , 1709 , 1712 , 1715 , 1732 , 1751 , 1763 , 1806
<code>\theequation</code>	502 , 519
<code>\thefootnote</code>	23
<code>\thempfn</code>	23
<code>\thempfn</code>	605 , 612 , 755
<code>\thempfootnote</code>	24
<code>\thempfootnote</code> ...	631 , 633 , 755
<code>\thempfootnote@latex</code> .	629 , 631
<code>\thepage</code>	686 , 692
<code>\thr@</code>	11
<code>\thr@@</code> .	238 , 514 , 540 , 1747 , 1798
<code>\title</code>	40
<code>\toc@@font</code>	60
<code>\toc@@font</code>	2013 , 2022
<code>\toc@letdimen</code> ..	2028 , 2029 , 2049
<code>\toc@post</code>	61
<code>\toc@post</code>	2008 , 2012
<code>\toc@post@auto</code>	61
<code>\toc@post@auto</code>	2021
<code>\toc@pre</code>	61
<code>\toc@pre</code>	2000 , 2011
<code>\toc@pre@auto</code>	61
<code>\toc@pre@auto</code>	2021
<code>\toc@setindent</code>	2024 , 2025 , 2026 , 2027 , 2043
<code>\toc@writedimen</code>	2034 , 2035 , 2036 , 2037 , 2038 , 2039 , 2054
<code>\tocdim@appendix</code>	2019
<code>\tocdim@min</code> ...	2044 , 2050 , 2096
<code>\tocdim@pagenum</code>	2020 , 2083
<code>\tocdim@paragraph</code>	2018
<code>\tocdim@section</code>	62
<code>\tocdim@section</code>	2015
<code>\tocdim@subsection</code>	2016
<code>\tocdim@subsubsection</code> ...	2017
<code>\tocleft@pagenum</code>	2069
<code>\tocleft@section</code>	61
<code>\tocmax@section</code>	61
<code>\toks</code>	1751 , 1806
<code>\toks@</code> ...	270 , 272 , 275 , 277 , 280 , 282 , 908 , 925 , 927
<code>\toks@ii</code>	12
<code>\toks@ii</code> .	271 , 272 , 276 , 277 , 281 , 282 , 297
<code>\toksdef</code>	297
<code>\toprule</code>	50
<code>\toprule</code>	1574
<code>\topsep</code>	524
<code>\total@float</code>	33
<code>\total@float</code>	811 , 847
<code>\traceoutput</code>	11
<code>\traceoutput</code>	261
<code>\tracingall</code>	11
<code>\tracingcommands</code>	256
<code>\tracinglostchars</code>	257
<code>\tracingmacros</code>	258
<code>\tracingonline</code>	256
<code>\tracingoutput</code>	11
<code>\tracingoutput</code>	257
<code>\tracingpages</code>	257
<code>\tracingparagraphs</code>	258
<code>\tracingplain</code>	11
<code>\tracingplain</code>	255

[\tracingrestores](#) 258
[\tracingstats](#) 256
[\trigger@float@par](#) .. 821, 824,
 825, 831
[\triggerpar](#) 807, 821
[\true@sw](#) 311, 318, 324, [336](#),
 341, 351, 1089, 1196, 2143,
 2255, 2258
[\tw@](#) 11
[\typeout](#) 66
[\typeout](#) [2241](#)
[\typeout@ltx](#) 2247, 2254
[\typeout@org](#) 2241, 2253

U

[\unhbox](#) 2074, 2091
[\unhcopy](#) 1616
[\unrestored@protected@xdef](#) 605
[\unskip](#) [738](#), [1343](#), [1349](#), [1575](#), [1576](#),
 [1708](#), [1714](#), [1895](#), [1897](#), [1899](#),
 [1918](#), [1922](#), [1926](#), [2073](#)
[\unvbox](#) 714, 728, 733, 743
[\unvcopy](#) 725, 874
[\uppercase](#) 38
[\url](#) 64
[\url](#) 26, 65, 72, [2130](#)
[\URL@prefix](#) [2130](#)
[\usepackage](#) 3

V

[\vadjust](#) 660
[\vbox](#) .. 713, 874, 1398, 1464, 1554
[\vcenter](#) 49
[\vcenter](#) . 1398, 1464, 1555, 1559,
 1562
[\voidb@x](#) 710
[\vrule](#) [1320](#), [1321](#), [1345](#), [1351](#), [1401](#),
 [1445](#), [1453](#), [1476](#), [1631](#)
[\vsize](#) 14
[\vsplit](#) 26
[\vtop](#) 725, 1398, 1464, 1553

W

[\write](#) 925, 2055, 2244, 2250
[\write@@float](#) [899](#), [903](#)
[\write@float](#) 27, 32, 34
[\write@float](#) 797, [899](#)
[\write@floatline](#) 34
[\write@floatline](#) [913](#), [915](#)
[\write@floats](#) 798, [899](#)

X

[\x](#) 414, 417
[\xdef](#) 459, 475, 1459, 1482

Y

[\y](#) 415, 417
[your_tokens_here](#), argument . 15
\your document here placeholder 9

Z

[\z@skip](#) 493,
 494, 495, 510, 515, 533, 541,
 660, 772, 1407, 1413, 1436,
 1642, 1643, 1644, 2074

Change History

4.0b	General: AO: Fixed spurious CR and (return) characters in output file. Also, if the document did not have the <code>\endfigure</code> on a line of its own, the macro wouldn't work. Fixed.	3		
	AO: Removed superfluous <code>\defs</code> , changed to using <code>\floats@sw</code> as the flag. Also stopped using DPC's <code>\if@twocolumn</code> flag: using <code>\floats@sw</code> instead. Also added <code>\par\vskip\z@skip</code> after the <code>\minipagefootnotes</code> so that the float box would have zero depth like the kernel one.	3		
	only execute if there really were floats of the given type	3		
	Support the hack with <code>\prepdef</code> , and delay until <code>\AtBeginDocument</code> time, since <code>hyperref</code> clobbers <code>\caption</code> . . .	3		
	<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.	30		
	<code>\caption</code> : Support the hack with <code>\prepdef</code> , and delay until <code>\AtBeginDocument</code> time, since <code>hyperref</code> clobbers <code>\caption</code> . .	29		
	<code>\print@float</code> : only execute if there really were floats of the given type	32		
	<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.	34		
4.0c	General: (AO, 110) Install hooks for endfloats processing	3		
	(AO, 116) <code>Hyperref</code> compatibility	3		
	(AO, 130) Interference from array package	3		
	*-form mandates pagebreak at each float; only print section head if there is something there.	3		
	<code>\@mpmakefntext</code> : (AO, 110) Install hooks for endfloats processing	30		
	<code>\@ssect</code> : (AO, 116) <code>Hyperref</code> compatibility	37		
	<code>\endarray</code> : (AO, 130) Interference from array package	39		
	<code>\print@float</code> : *-form mandates pagebreak at each float; only print section head if there is something there.	32		
4.0d	General: (AO, 127) Floats placed [h] to allow page breaks	3		
	(AO, 174) kernel fix	3, 22		
	(AO, 224) <code>Hyperref</code> compatibility.	3		
	Allow things to break over pages by setting <code>array@default</code>	3		
	<code>\@mpmakefntext</code> : (AO, 127) Floats placed [h] to allow page breaks	30		
	(AO, 224) <code>Hyperref</code> compatibility.	30		
	<code>\print@float</code> : Allow things to break over pages by setting <code>array@default</code>	32		
4.0e	General: (AO, 221) Remove <code>samepage</code> command from <code>@xfloat@prep</code> : If the float can break over pages, we want better control.	3		
	<code>\@mpmakefntext</code> : (AO, 221) Remove <code>samepage</code> command from <code>@xfloat@prep</code> : If the float can break over pages, we want better control.	30		
4.0f	General: (AO, 404) <code>Hyperref</code> compatibility	3		
	<code>\@ssect</code> : (AO, 404) <code>Hyperref</code> compatibility	37		

4.1a	
General: (AO) Make	
<code>\addtocontents a \long \def;</code>	
gobble up <code>\footnote</code>	3
(AO) Remove code that avoided	
changes to <code>\xfootnotemark</code>	3
(AO, 438) Complete rewrite of	
footnote macros.	3
(AO, 459) do not assume	
<code>\class@name</code> is defined	3, 10
(AO, 461) Change the csname	
from <code>\dotsep</code> to	
<code>\ltxu@dotsep</code> . The former is	
understood in mu. (What we	
wanted was a dimension.)	3
(AO, 475) I had not properly	
reproduced the LaTeX macro	
<code>\eqnarray</code>	3
<code>\xfloat@prep</code> calls	
<code>\ltx@footnote@pop</code> to restore	
the original <code>\ltx@footmark</code>	
and <code>\ltx@foottext</code>	
procedures, in case footnote	
processing has switched.	3
<code>\class@documenthook</code> is the last	
<code>\AtBeginDocument</code> token now	3
Class extension mechanism	
<code>\pushfilename@ltx</code> and	
<code>\p@pfilename@ltx</code>	3
Class extension mechanism	
<code>\class@extension</code> ,	
<code>\class@extensionfile</code> , and	
<code>\class@ext@hook</code>	3, 18
Get rid of <code>\set@typesize@hook</code>	
<code>\set@pica@hook</code> and the	
<code>\normalsize</code> directive	3
<code>\@mpmakefntext: \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.	30
<code>\p@pfilename: Class extension</code>	
mechanism	
<code>\pushfilename@ltx</code> and	
<code>\p@pfilename@ltx</code>	19
<code>\class@enddocumenthook:</code>	
<code>\class@documenthook</code> is the	
last <code>\AtBeginDocument</code> token	
now	15
<code>\document: Get rid of</code>	
<code>\set@typesize@hook</code>	
	<code>\set@pica@hook</code> and the
	<code>\normalsize</code> directive
	14
	<code>\eqnarray@fleqn@fixed: (AO,</code>
	475) I had not properly
	reproduced the LaTeX macro
	<code>\eqnarray</code>
	20
	<code>\footnote: (AO) Remove code</code>
	that avoided changes to
	<code>\xfootnotemark</code>
	25
	<code>\ltx@make@current@footnote:</code>
	(AO, 438) Complete rewrite of
	footnote macros.
	23
	<code>\numberline@@sections: (AO,</code>
	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.)
	62
	<code>\robustify@contents: (AO) Make</code>
	<code>\addtocontents a \long \def;</code>
	gobble up <code>\footnote</code>
	28
	<code>\toc@@font: (AO, 461) Change the</code>
	csname from <code>\dotsep</code> to
	<code>\ltxu@dotsep</code> . The former is
	understood in mu. (What we
	wanted was a dimension.)
	60
4.1b	
General: (AO, 487) Support for	
video figures and the	
<code>\setfloatlink</code> command	3
(AO, 505) try to accommodate	
<code>colortbl</code>	3
Acquire <code>hyperref</code> savoir	3
Default assignment of	
<code>\float@sw</code> now, not at	
<code>\AtBeginDocument</code> time.	3
If class option <code>lengthcheck</code> is in	
effect, log the height of this	
float class.	3
No need to protect against	
undefined <code>\float@sw</code>	3
Patch the array package even	
later: after all package patches	
go in.	3
Refine toc processing: provide	
default.	3
Tally and log the height of a	
float class	3
<code>\mkpream: (AO, 505) try to</code>	
accommodate <code>colortbl</code>	54
<code>\mkpream@relax: (AO, 505) try</code>	
to accommodate <code>colortbl</code>	55
<code>\@mpmakefntext: No need to</code>	
protect against undefined	

<code>\float@sw</code>	30	(AO, 532) Both arguments of <code>\href</code> get sanitized	3
<code>\@tabarray:</code> (AO, 505) try to accommodate <code>colortbl</code>	44	(AO, 539) Use of double-backslash in argument of <code>\section</code> gives error. The <code>textcase</code> package is involved.	3, 19
<code>\@tabular:</code> (AO, 505) try to accommodate <code>colortbl</code>	43	<code>\doibase:</code> (AO, 532) Both arguments of <code>\href</code> get sanitized	64
<code>\array:</code> (AO, 505) try to accommodate <code>colortbl</code>	46	4.1j <code>\doibase:</code> (AO, 545) Provide definition for <code>\doi</code> that does hypertext	64
<code>\do@if@floats:</code> No need to protect against undefined <code>\float@sw</code>	31	<code>\hypertext@enable@ltx:</code> (AO, 545) hypertext capabilities off by default; enable with <code>hypertext</code>	64
<code>\doibase:</code> (AO, 487) Support for video figures and the <code>\setfloatlink</code> command	64	4.1n General: (AO) Incorporate change to <code>ltxmiscn.dtx</code> v1.1i 2000/05/19	3
<code>\endarray:</code> Patch the array package even later: after all package patches go in.	39	(AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page	3
<code>\floats@sw:</code> Default assignment of <code>\float@sw</code> now, not at <code>\AtBeginDocument</code> time.	30	(AO, 569) execute <code>atveryend</code> 's <code>\Call@AfterLastShipout</code> at the proper time	3
<code>\init@hyperref:</code> Acquire <code>hyperref</code> savoir	38	(AO, 571) Interface <code>\set@footnotewidth</code> for determining the set width of footnotes	3
<code>\ltx@contentsline:</code> Refine toc processing: provide default.	28	(AO, 571) allow split after last line of footnote	3
<code>\print@float:</code> If class option <code>lengthcheck</code> is in effect, log the height of this float class.	32	(AO, 572) title block footnotes numbered independently from body footnotes	3
<code>\switch@array:</code> (AO, 505) Try to accommodate <code>colortbl</code>	41	(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>	3
<code>\total@float:</code> Tally and log the height of a float class	33	<code>\clear@document:</code> (AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page	17
4.1d General: (AO, 511) Compatability with <code>lineno.sty</code> 's erroneous way of detecting <code>fleqn.clo</code>	3	(AO, 569) execute <code>atveryend</code> 's <code>\Call@AfterLastShipout</code> at the proper time	17
<code>\eqnarray@fleqn@fixed:</code> (AO, 511) Compatability with <code>lineno.sty</code> 's erroneous way of detecting <code>fleqn.clo</code>	21	<code>\do@check@aux:</code> (AO) Incorporate change to <code>ltxmiscn.dtx</code> v1.1i 2000/05/19	15
4.1f General: (AO, 515) Hook for setting the font of a footnote	3	(AO, 569) Use of <code>hyperref</code> interferes with column balancing of last page	15
(AO, 518) Tally register overflow when document is long	3		
<code>\set@footnotewidth:</code> (AO, 515) Hook for setting the font of a footnote	27		
<code>\total@float:</code> (AO, 518) Tally register overflow when document is long	33		
4.1g General: (AO, 525) Remove phantom paragraph above display math that is given in vertical mode	3, 21		

<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>	62	4.2a	General: (MD) Updated name of README file and use standard fonts when typesetting	3
<code>\set@footnotewidth:</code> (AO, 571) Interface <code>\set@footnotewidth</code> for determining the set width of footnotes	27		(MD) Use updated best practice to use https and doi.org	3
(AO, 571) allow split after last line of footnote	26	4.2d	General: (PHO) Adapt <code>\document</code> and <code>\enddocument</code> hooks to the 2020-10-01 L ^A T _E X release.	3
(AO, 572) title block footnotes numbered independently from body footnotes	26		<code>\do@check@aux:</code> (PHO) Only redefine <code>\enddocument</code> in older versions.	15
4.1p			(PHO) Patch <code>\enddocument</code> at runtime in newer versions.	16
General: (AO, 582) A patch of <code>hyperref.sty</code> to provide backward compatibility to T _E XLive 2007's version 6.75r . . .	3		<code>\document:</code> (PHO) Use L ^A T _E X's hook management system, if possible.	14
<code>\href:</code> (AO, 582) A patch of <code>hyperref.sty</code> to provide backward compatibility to T _E XLive 2007's version 6.75r . . .	64	4.2f	<code>\rvtx@ifformat@geq:</code> (PHO) Add <code>\rvtx@ifformat@geq</code>	13
			<code>\eqnarray@fleqn@fixed:</code> (PHO) Fix detection of <code>\eqnarray</code> in newer L ^A T _E X.	20