

# First aid for external files and packages that need updating

Frank Mittelbach, L<sup>A</sup>T<sub>E</sub>X Project

October 13, 2024

## Abstract

This file contains some first aid for packages or classes that require updates because of internal changes to L<sup>A</sup>T<sub>E</sub>X but that aren't yet reflected in the package/class code.

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Minor kernel fixes . . . . .	2
<b>2</b>	<b>The Implementation</b>	<b>2</b>
2.1	The filehook package first aid . . . . .	3
2.2	The dinbrief class first aid . . . . .	5
2.3	The unicode-math package first aid . . . . .	5
2.4	The pgfpages and pgfmorepages first aid . . . . .	5
2.5	The babel package . . . . .	6
2.6	The songs package first aid . . . . .	6
2.7	The crop package first aid . . . . .	6
2.8	The bigfoot first aid . . . . .	7
2.9	ulem first aid . . . . .	7
2.10	varwidth first aid . . . . .	8
2.11	The german class first aid . . . . .	8
2.12	The underscore first aid . . . . .	9
2.13	The acro package first aid . . . . .	9
2.14	The chemformula package first aid . . . . .	10
2.15	The chemnum package first aid . . . . .	10
2.16	The cleveref package first aid . . . . .	10
2.17	The arydshln package first aid . . . . .	11

## 1 Introduction

Over the years package writers have hooked into various parts of internal L<sup>A</sup>T<sub>E</sub>X commands (largely because proper interfaces were missing in important places) and if we are now gradually adding such interfaces these internal commands do change and as a result patching into them stops working.

As part of making such internal changes the L<sup>A</sup>T<sub>E</sub>X Project team attempts to check for such usage in packages, alert the package maintainers and ensures that the packages get updated alongside the core L<sup>A</sup>T<sub>E</sub>X system. However it is not always possible to get packages that will fail with a new kernel updated in time and if that is the case we try to provide a temporary fix in this file for them. Once the package gets updated the fix will then be removed again.

For that reason, it is put into a separate bundle so that we can update it easily without requiring the CTAN maintainers to install a new full L<sup>A</sup>T<sub>E</sub>X system just because we take out (or add) a fix for a package here.

In the best case scenario the file documented here should be empty. In practice it will probably always contain one or the other fix while we are waiting for the package to get updated.

**Important notice:** The fixes provided here are not meant to be a permanent solution, but are only provided to support the transition period. They are (usually) neither complete nor necessarily the best solution! Furthermore, as they are done from the “outside”, they usually add some burden and slow down L<sup>A</sup>T<sub>E</sub>X processing, even if the package/class is not used in the document.

We will therefore remove such code as soon as possible again. In practice this means that if some package never gets updated/corrected, then it will eventually fail to work, because after one or at most two L<sup>A</sup>T<sub>E</sub>X releases we will take out the transition code to ensure that this “first aid patching” doesn’t get out of bounds.

## 1.1 Minor kernel fixes

If we encounter issues with the kernel code that should get fixed before the next main release we normally generate a patch release for L<sup>A</sup>T<sub>E</sub>X. However, depending on the complexity of the fix we might first add the fix here and generate a full patch release only when a number of such issues have accumulated. This way we lessen the impact on CTAN maintainers because for each tach release we have to make and distribute also a matching development release.

## 2 The Implementation

This file is meant to be loaded during format generation which is why we give it the extension `.ltx`.

```

1 (*kernel)
2 \def\LaTeXFirstAidDate{2024/07/05}
3 \def\LaTeXFirstAidVersion{v1.1f}
4 \ProvidesFile{latex2e-first-aid-for-external-files.ltx}
5     [\LaTeXFirstAidDate\space \LaTeXFirstAidVersion\space
6     LaTeX kernel fixes to external files and packages]
```

`\FirstAidNeededT` This is a very simple help to ensure that we only apply first aid to an unmodified package or class. It only works in the case the file has already been loaded and the csname `\ver@#1.#2` got defined (holding the current date, version, and short description info). We then compare its content to a frozen string and make the

modification #3 only if both agree. If they differ we assume that the package/class in question got updated by its maintainer.

```

7 \ExplSyntaxOn
8 \cs_new:Npn\FirstAidNeededT#1#2#3{
9   \exp_args:Ncx\str_if_eq:onF{ver@#1.#2}{#3}
10    { \typeout{=>~ First~ Aid~ for~ #1.#2~ no~ longer~ applied!^^J
11      \@spaces Expected:^^J
12      \@spaces\@spaces #3^^J
13      \@spaces but~ found:^^J
14      \@spaces\@spaces \use:c{ver@#1.#2}^^J
15      \@spaces so~ I'm~ assuming~ it~ got~ fixed.
16    } }
17   \exp_args:Ncx\str_if_eq:onT{ver@#1.#2}{#3}
18 }
19 \ExplSyntaxOff

20 </kernel>

```

## 2.1 The filehook package first aid

The filehook package implements hooks into file loading commands. These days this is already provided by the kernel albeit not with the same user interface. Until that package gets updated (to use the kernel interfaces) we provide a substitution. This does not offer all hooks from filehook but all that have been used in packages available in T<sub>E</sub>X Live.

Note that this doesn't fix currfile because that package uses filehook but relies on the internals of the old implementation.

The package has now got an update so we aren't activating the first aid. However, at the moment it basically bypasses the new hook mechanism and puts the old hooks in thereby disabling, for example, the possibility to re-order code through rules.

We therefore keep filehook-ltx.sty around as a guideline for further updates.

Replacing filehook with a leaner version would then work like this:

```

21 (*kernel)
22 %\declare@file@substitution{filehook.sty}{filehook-ltx.sty}
23 </kernel>

```

What follows is a simplified (partial) implementation of the filehook interfaces. Not implemented are:

```

\AtBeginOfFiles      \AtEndOfFiles
\AtBeginOfInputs     \AtEndOfInputs
\AtBeginOfInputFile  \AtEndOfInputFile

24 (*filehook-ltx)

25 \newcommand\AtBeginOfEveryFile [1]
26   {\AddToHook{file/before}{#1}}
27 \newcommand\AtEndOfEveryFile [1]
28   {\AddToHook{file/after}{#1}}

29 \newcommand\AtBeginOfIncludes [1]
30   {\AddToHook{include/before}{#1}}
31 \newcommand\AtEndOfIncludes [1]

```

```

32 {\AddToHook{include/end}{#1}}
33 \newcommand\AfterIncludes [1]
34 {\AddToHook{include/after}{#1}}

35 \newcommand\AtBeginOfPackages [1]
36 {\AddToHook{package/before}{#1}}
37 \newcommand\AtEndOfPackages [1]
38 {\AddToHook{package/after}{#1}}

39 \newcommand\AtBeginOfClasses [1]
40 {\AddToHook{class/before}{#1}}
41 \newcommand\AtEndOfClasses [1]
42 {\AddToHook{class/after}{#1}}

43 \newcommand\AtBeginOfFile [2]
44 {\AddToHook{file/#1/before}{#2}}
45 \newcommand\AtEndOfFile [2]
46 {\AddToHook{file/#1/after}{#2}}

```

Some commands offered a starred form

```

47 \DeclareDocumentCommand \AtBeginOfPackageFile {smm}
48 {\IfBooleanTF{#1}%
49   {\@ifpackageloaded{#2}%
50     {#3}%
51     {\AddToHook{package/#2/before}{#3}}}%
52   {\AddToHook{package/#2/before}{#3}}%
53   }

54 \DeclareDocumentCommand \AtEndOfPackageFile {smm}
55 {\IfBooleanTF{#1}%
56   {\@ifpackageloaded{#2}%
57     {#3}%
58     {\AddToHook{package/#2/after}{#3}}}%
59   {\AddToHook{package/#2/after}{#3}}%
60   }

```

Are the \* forms here of any use? I know they are use 3–4 times on CTAN but I wonder if those are real or mistaken usages.

```

61 \DeclareDocumentCommand \AtBeginOfClassFile {smm}
62 {\IfBooleanTF{#1}%
63   {\@ifclassloaded{#2}%
64     {#3}%
65     {\AddToHook{class/#2/before}{#3}}}%
66   {\AddToHook{class/#2/before}{#3}}%
67   }

68 \DeclareDocumentCommand \AtEndOfClassFile {smm}
69 {\IfBooleanTF{#1}%
70   {\@ifclassloaded{#2}%
71     {#3}%
72     {\AddToHook{class/#2/after}{#3}}}%
73   {\AddToHook{class/#2/after}{#3}}%
74   }

75 \newcommand\AtBeginOfIncludeFile [2]
76 {\AddToHook{include/#1/before}{#2}}
77 \newcommand\AtEndOfIncludeFile [2]
78 {\AddToHook{include/#1/end}{#2}}

```

```

79 \newcommand\AfterIncludeFile [2]
80   {\AddToHook{include/#1/after}{#2}}
81 </filehook-ltx>
82 <*kernel>

```

## 2.2 The `dinbrief` class first aid

Again a case of a no longer correct `\endgroup` in document. Here the fix is simply though.

```

83 \AddToHook{file/dinbrief.cls/after}[firstaid]{%
84   \FirstAidNeededT{dinbrief}{cls}{2000/03/02 LaTeX2e class}%
85   {\AddToHook{env/document/begin}{\begingroup}}%
86 }

```

## 2.3 The `unicode-math` package first aid

If `unicode-math` is used together with `doc` there is a problem because it changes the mathcodes without adjusting the use in `doc` that assume standard settings. Could be fixed on either side, but as `unicode-math` is derivating from the standard, the right place is probably a fix in this package. For now we do it here. See [github/820](https://github.com/820).

```

87 \AddToHook{package/unicode-math/after}{%
88   \AddToHook{cmd/mod@math@codes/after}{\mathcode'\|=28796 }}

```

## 2.4 The `pgfpages` and `pgfmorepages` first aid

`pgfpages` alters the `\shipout` primitive to support multiple page up scenarios. If used together with `atbegshi` that worked because the alterations done by `atbegshi` came later and so used the new definition provide by `pgfpages`. Now that the code from `atbegshi` is already in the kernel this further redefinition doesn't happen with the result that the change to `\shipout` comes to late and breaks the kernel processes.

```

89 \ExplSyntaxOn
90 \AddToHook{file/pgfpages.sty/after}[firstaid]{%

```

Undo overwriting `\shipout`:

```

91   \cs_gset_eq:NN \shipout \pgfpages@originalshipout

```

Instead overwrite the L3 programming layer name of the primitive. This is really an absolute no-go, but for now the simplest solution to keep the original code running.

It will be replaced when the “configuration points” interface for  $\LaTeX$  becomes available. At that point the package will be able to set up a different strategy for doing shipouts and without the need to overwrite a primitive (which it did in the past and which we do below) and then this code here can be taken out again.

```

92   \cs_set_eq:NN \pgfpages@originalshipout \tex_shipout:D
93   \cs_set_eq:NN \tex_shipout:D \pgfpages@interceptshipout
94 }
95 \ExplSyntaxOff

```

Same issue with `pgfmorepages` but slightly different implementation (sigh).

```
96 \ExplSyntaxOn
97 \AddToHook{file/pgfmorepages.sty/after}[firstaid]{
98   \cs_set_nopar:Npn \pgfhookintoshipout {
99     \cs_set_eq:NN \pgfpages@originalshipout \tex_shipout:D
100    \cs_set_eq:NN \tex_shipout:D \pgfpages@interceptshipout
101   }
102 }
103 \ExplSyntaxOff
```

## 2.5 The `babel` package

Turn off the `babel` hack.

```
104 \AddToHook{file/babel.sty/before}[firstaid]{\def\BabelCaseHack{}}
```

## 2.6 The `songs` package first aid

The `songs` package uses `\obeylines` and redefines `\par` for special effect. this no longer works in L<sup>A</sup>T<sub>E</sub>X 2022-06-01 (gh issue 367). The following fixes at least one failure.

```
105 \AddToHook{file/songs.sty/after}[firstaid]{%
106   \FirstAidNeededT{songs}{sty}{2018/09/12 v3.1 Songs package}%
107   {%
108     \renewcommand\SB@obeylines{%
109       \let\obeyedline\SB@par%
110       \obeylines%
111       \let\@par\SB@par%
112     }%
113 }
```

## 2.7 The `crop` package first aid

The `crop` packages fails currently due to two L<sup>A</sup>T<sub>E</sub>X changes: It doesn't know that `\stockheight` and `\stockwidth` are now defined, and doesn't take into account that `\rlap` is robust (<https://github.com/rrthomas/crop/issues/2>). The first is addressed by setting the dimension if they are zero or negative. For the second we locally change the meaning of `\protect`

```
114 \AddToHook{file/crop.sty/after}[firstaid]{%
115   \FirstAidNeededT{crop}{sty}{2017/11/19 1.10 crop marks (mf)}%
116   {%
117     \ifdim\stockwidth > 0pt \else \stockwidth\paperwidth \fi
118     \ifdim\stockheight > 0pt \else \stockheight\paperheight \fi
119     \renewcommand*\CROP@genreflect[1]{%
120       \leavevmode
121       \dimen0\CROP@horigin
122       \kern2\dimen0
123       \begingroup
124       \set@typeset@protect %change protect
125       \reflectbox{%
126         \hb@xt@\paperwidth{%
127           \vbox to\paperheight{%
128             #1%
```

```

129             \vss
130             }%
131             \hss
132         }%
133     }%
134 \endgroup
135 }
136 }%
137 }
138 </kernel>
139 (*kernel)

```

## 2.8 The bigfoot first aid

The `bigfoot` packages makes the assumption that two `\newinsert` allocations have a recognisable order in their numbers, the second one has a lower number. This was correct in the classic  $\TeX$  implementation but with the extended allocation possibilities of all modern engines is no longer the case and there is a point where the allocations take a “jump” breaking the ordering assumption. These days we are fairly close to that point and depending on how many packages are loaded before `bigfoot` the package breaks.

This `firstaid` therefore jumps over the problematical point by pushing the count allocation to a safe value if necessary.

```

140 \AddToHook{file/bigfoot.sty/after}{%
141     \ifnum\count10<\insecunt
142         \global\count10=\insecunt
143     \fi

```

We also correct a bug that `bigfoot` tries to shift mark registers, but in  $\LaTeX$  (at least since 2015) the allocation number is not 266, so it does that to a random number of mark registers (which sometimes blows up depending on the value in 266).

```

144     \def\FN@allmarks#1{\@elt{#1}%
145         \ifnum#1<\count256 %<--- problem: 266 isn't the counter for marks
146             \expandafter\FN@allmarks\expandafter{\number\numexpr#1+\@ne}%
147             \fi}%
148 }

```

## 2.9 ulem first aid

In 2020 we fixed various kernel commands to accept `calc` syntax. The `ulem` package redefines some internals and that now conflicts with the new definitions as they involve an extra group. So we alter the definition of `\@hspace` if `ulem` was loaded. This is not perfect, obviously, so it will go out the moment `ulem` gets adjusted.

```

149 \AddToHook{file/ulem.sty/after}[firstaid]{%
150     \def\@hspace#1{\begingroup\setlength\skip@{#1}%
151         \edef\x{\endgroup\hskip\the\skip@\relax}\x}%
152 }

```

## 2.10 varwidth first aid

The `varwidth` package does a lot of low-level paragraph manipulation assuming traditional  $\TeX$  paragraphs. However, with the paragraph hooks we end up with one extra glue `Opt` item on the vertical list and if that isn't removed then the package doesn't find its penalties.

So this needs to be removed as well by adding an additional `\unskip`.

```
153 \AddToHook{file/varwidth.sty/after}[firstaid]{%
154   \FirstAidNeededT{varwidth}{sty}%
155     {2009/03/30 ver 0.92; \space Variable-width minipages}%
156     {%
157 \def\@vwid@sift{%
158   \skip@\lastskip\unskip
159   \ifdim\lastskip=\z@\unskip\fi   % <---- the first aid here (not just unskip)
160   \dimen@\lastkern\unkern
161   \count@\lastpenalty\unpenalty
162   \setbox\z@\lastbox
163   \ifvoid\z@ \advance\sift@deathcycles\@ne \else \sift@deathcycles\z@ \fi
164   \ifnum\sift@deathcycles>33
165     \let\@vwid@sift\relax
166     \PackageWarning{varwidth}{Failed to reprocess entire contents}%
167   \fi
168   \ifnum\count@=\@vwid@preeqp \@vwid@eqmodfalse\fi
169   \ifnum\count@=\@vwid@posteqp \@vwid@eqmodetrue\fi
170   \ifnum\count@=\@vwid@toppen % finished
171     \let\@vwid@sift\relax
172   \else\ifnum\count@=\@vwid@offsets
173     \@vwid@setoffsets
174   \else
175     \ifnum\count@=\@vwid@postw
176       \else
177         \@vwid@resetb % reset box \z@ or measure it
178       \fi
179     \@vwid@append
180   \fi\fi
181   \@vwid@sift}%
182   }%
183 }
```

## 2.11 The german class first aid

Handling of `\protected UTF-8`

```
184 \AddToHook{file/german.sty/after}[firstaid]{%
185   \FirstAidNeededT{german}{sty}{1998/07/08 v2.5e Support for writing german texts (br)}
186   {%
187 \let\grmn@active@ddq\@active@dq
188 \def\@active@ddq{\protect\grmn@active@ddq}%
189 \germanTeX
190 }%
191 }

192 \AddToHook{file/ngerman.sty/after}[firstaid]{%
193   \FirstAidNeededT{ngerman}{sty}{1998/07/08 v2.5e Support for writing german texts (br)}
194   {%
```



```

195 \let\grmn@active@dq@\@active@dq
196 \def\@active@dq{\protect\grmn@active@dq}%
197 \ngermanTeX
198 }%
199 }
200 </kernel>

```

## 2.12 The underscore first aid

The `underscore` package makes the underscore active. This means that the underscore can not be used in label and references unless the package option `strings` is used (which patches a selection of problematic commands like `\label` and `\ref`) or `babel` is used which redefines<sup>1</sup> a selection of problematic commands like `\@testdef` or `\@newl@bel`.

With the new property commands the work-around do not work. We therefore make the underscore protected and use `\ifincsname` to allow its use in csnames.

```

201 <*underscore-ltx>
202 \ProvidesPackage{underscore-ltx}[2023/09/20 LaTeX firstaid to make underscore protected ]
203 \begingroup
204 \catcode'\_=\active
205 \protected\gdef _{%
206   \ifincsname %
207     \string_%
208   \else
209     \ifx\protect\@typeset@protect
210       \ifmmode \sb \else \BreakableUnderscore \fi
211     \else
212       \ifx\protect\@unexpandable@protect \noexpand_%
213       \else \protect_%
214     \fi\fi
215   \fi
216 }
217 \global\let\ActiveUnderscore=_
218 \endgroup
219 </underscore-ltx>
220 <*kernel>
221 \AddToHook{file/underscore.sty/after}[firstaid]{%
222   \FirstAidNeededT{underscore}{sty}{2006/09/13}{\RequirePackage{underscore-ltx}}
223 </kernel>
224 <*kernel>

```

## 2.13 The acro package first aid

The package does not declare a `prop`, which causes an issue with newer routines in `expl3`.

```

225 \AddToHook{package/acro/after}[firstaid]{%
226   \FirstAidNeededT{acro}{sty}{2022/04/01 v3.8 typeset acronyms
227     and other abbreviations (CN)}
228   {\UseName{prop_new:c}{l__acro_tmpa_prop}}%

```

---

<sup>1</sup>unless the recommended option `safe=none` is used

With the 2024 June release of L<sup>A</sup>T<sub>E</sub>X it will also fail to patch `\endlongtable` and therefore errors when loading. However, the patch it tries never worked (because it was setting a local boolean at a point where it was more or less immediately reset). Thus, rather than fixing the patch approach (which requires to surround the patch with `\ExplSyntaxOn \catcode'\_ =10_` and `\ExplSyntaxOff`) we simply disable the patch for now.

```
229     \acsetup{patch/longtable=false}%
230   }%
231 }
```

## 2.14 The chemformula package first aid

Package `chemformula` uses `l3keys2e` for option processing. This used to be made available as `chemformula` also loads `xfrac`, which loaded `l3keys2e`. However, `xfrac` has now been updated to use the newer kernel method if available, so loading `chemformula` fails.

```
232 \AddToHook{package/chemformula/before}[firstaid]{%
233   \RequirePackage{l3keys2e}%
234 }
```

## 2.15 The chemnum package first aid

The package does not declare a `prop`, which causes an issue with newer routines in `expl3`.

```
235 \AddToHook{package/chemnum/after}[firstaid]{%
236   \FirstAidNeededT{chemnum}{sty}{2021/01/21 v1.3a a comprehensive
237     approach for the numbering of chemical compounds (CN)}
238   {\UseName{prop_new:c}{l__chemnum_tmpa_prop}}}%
239 }
```

## 2.16 The cleveref package first aid

The `cleveref` package expects only two data containers for its internal `\newlabel` command. This fails if `xr-hyper` is used which expands every `\newlabel` to five data container and puts the file name into the last one.

```
240 \AddToHook{package/cleveref/after}[firstaid]{%
241   \FirstAidNeededT{cleveref}{sty}{2018/03/27 v0.21.4 Intelligent cross-referencing}
242   {%
```

This are the two commands which retrieve the data from the label info. We change them to expect five arguments.

```
243     \def\cref@getref#1#2{%
244       \expandafter\let\expandafter#2\csname r@#1@cref\endcsname%
245       \expandafter\expandafter\expandafter\def%
246         \expandafter\expandafter\expandafter#2%
247         \expandafter\expandafter\expandafter{%
248           \expandafter\@firstoffive#2}}% <----- five
249     \def\cpageref@getref#1#2{%
250       \expandafter\let\expandafter#2\csname r@#1@cref\endcsname%
251       \expandafter\expandafter\expandafter\def%
252         \expandafter\expandafter\expandafter#2%
253         \expandafter\expandafter\expandafter{%
```

```

254         \expandafter\@secondoffive#2}}% <----- five
We also need to redefine the internal label commands of cleveref. This must be
done after cleveref has made its changes in begindocument so we add it to the
same hook using the hook label used by cleveref. This way it is guaranteed to
overwrite the definitions.
255     \AddToHook{begindocument}[cleveref]{%
256         \def\label@noarg#1{%
257             \cref@old@label{#1}%
258             \@bsphack%
259             \edef\@tempa{{page}{\the\c@page}}%
260             \setcounter{page}{1}%
261             \def\@tempb{\thepage}%
262             \expandafter\setcounter\@tempa%
263             \cref@constructprefix{page}{\cref@result}%
264             \protected@write\@auxout{%
265                 {\string\newlabel{#1@cref}}{\cref@currentlabel}%
266                 {[\@tempb][\arabic{page}][\cref@result]\thepage}{-}{-}}% <----- five
267             \@esphack}%
268     \def\label@optarg[#1]#2{%
269         \cref@old@label{#2}%
270         \@bsphack%
271         \edef\@tempa{{page}{\the\c@page}}%
272         \setcounter{page}{1}%
273         \def\@tempb{\thepage}%
274         \expandafter\setcounter\@tempa%
275         \cref@constructprefix{page}{\cref@result}%
276         \protected@edef\cref@currentlabel{%
277             \expandafter\cref@override@label@type%
278             \cref@currentlabel\@nil{#1}}%
279         \protected@write\@auxout{%
280             {\string\newlabel{#2@cref}}{\cref@currentlabel}%
281             {[\@tempb][\arabic{page}][\cref@result]\thepage}{-}{-}}% <----- five
282         \@esphack}%
283     }
284 }%
285 }

```

## 2.17 The `arydshln` package first aid

Making two internal commands robust to avoid expansion while constructing the array preamble.

```

286 \AddToHook{package/arydshln/after}[firstaid]{%
287     \FirstAidNeededT{arydshln}{sty}{2019/02/21 v1.76 }
288     {%
289     % add \protected
290     \protected\def\adl@vlineL#1#2#3#4{\adl@ivline#4\@nil{#1}{#2}%
291         \xdef\adl@colsL{\adl@colsL
292             \@elt{#3}{\number\@tempcnta}{\number\@tempcntb}%
293             {\adl@dashcolor}{\adl@gapcolor}}}%
294     \protected\def\adl@vlineR#1#2#3#4{\adl@ivline#4\@nil{#1}{#2}%
295         \xdef\adl@colsR{%
296             \@elt{#3}{\number\@tempcnta}{\number\@tempcntb}%
297             {\adl@dashcolor}{\adl@gapcolor}%

```

```

298             \adl@colsR}}}%
299 \let\adl@act@@vlineL\adl@vlineL
300 \let\adl@act@@vlineR\adl@vlineR
301 }%
302 }
303 </kernel>

```

## Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols	A	
\@active@dq . . . . .	\acsetup . . . . .	\AtEndOfClasses . . . . . 41
<i>187, 188, 195, 196</i>	\active . . . . .	\AtEndOfClassFile . . . . . 68
\@auxout . . . . .	\ActiveUnderscore . . . . .	\AtEndOfEveryFile . . . . . 27
<i>264, 279</i>	\AddToHook . . . . .	\AtEndOfFile . . . . . 45
\@bsphack . . . . .	<i>26, 28,</i>	\AtEndOfIncludeFile . . . . . 77
<i>258, 270</i>	<i>30, 32, 34, 36,</i>	\AtEndOfIncludes . . . . . 31
\@elt . . . . .	<i>38, 40, 42, 44,</i>	\AtEndOfPackageFile . . . . . 54
<i>144, 292, 296</i>	<i>46, 51, 52, 58,</i>	\AtEndOfPackages . . . . . 37
\@esphack . . . . .	<i>59, 65, 66, 72,</i>	
<i>267, 282</i>	<i>73, 76, 78, 80,</i>	
\@firstoffive . . . . .	<i>83, 85, 87, 88,</i>	
<i>248</i>	<i>90, 97, 104, 105,</i>	
\@hspace . . . . .	<i>114, 140, 149,</i>	
<i>150</i>	<i>153, 184, 192,</i>	
\@ifclassloaded . . . . .	<i>221, 225, 232,</i>	
<i>63, 70</i>	<i>235, 240, 255, 286</i>	
\@ifpackageloaded . . . . .	\adl@@vlineL . . . . .	
<i>49, 56</i>	<i>290, 299</i>	
\@one . . . . .	\adl@vlineR . . . . .	
<i>146, 163</i>	<i>294, 300</i>	
\@nil . . . . .	\adl@act@@vlineL . . . . .	
<i>278, 290, 294</i>	<i>299</i>	
\@par . . . . .	\adl@act@@vlineR . . . . .	
<i>111</i>	<i>300</i>	
\@secondoffive . . . . .	\adl@colsL . . . . .	
<i>254</i>	<i>291</i>	
\@spaces <i>11, 12, 13, 14, 15</i>	\adl@colsR . . . . .	
\@tempa <i>259, 262, 271, 274</i>	<i>295, 298</i>	
\@tempb <i>261, 266, 273, 281</i>	\adl@dashcolor . . . . .	
\@tempcnta . . . . .	<i>293, 297</i>	
<i>292, 296</i>	\adl@gapcolor . . . . .	
\@tempcntb . . . . .	<i>293, 297</i>	
<i>292, 296</i>	\adl@ivline . . . . .	
\@typeset@protect . . . . .	<i>290, 294</i>	
<i>209</i>	\advance . . . . .	
\@unexpandable@protect	<i>163</i>	
. . . . .	\AfterIncludeFile . . . . .	
<i>212</i>	<i>79</i>	
\@vwid@append . . . . .	\AfterIncludes . . . . .	
<i>179</i>	<i>33</i>	
\@vwid@eqmodfalse . . . . .	\arabic . . . . .	
<i>168</i>	<i>266, 281</i>	
\@vwid@eqmodetrue . . . . .	\AtBeginOfClasses . . . . .	
<i>169</i>	<i>39</i>	
\@vwid@offsets . . . . .	\AtBeginOfClassFile . . . . .	
<i>172</i>	<i>61</i>	
\@vwid@posteqp . . . . .	\AtBeginOfEveryFile . . . . .	
<i>169</i>	<i>25</i>	
\@vwid@postw . . . . .	\AtBeginOfFile . . . . .	
<i>175</i>	<i>43</i>	
\@vwid@preeqp . . . . .	\AtBeginOfIncludeFile	
<i>168</i>	. . . . .	
\@vwid@resetb . . . . .	<i>75</i>	
<i>177</i>	\AtBeginOfIncludes . . . . .	
\@vwid@setoffsets . . . . .	<i>29</i>	
<i>173</i>	\AtBeginOfPackageFile	
\@vwid@sift . . . . .	. . . . .	
. . . . .	<i>47</i>	
<i>157, 165, 171, 181</i>	\AtBeginOfPackages . . . . .	
\@vwid@toppen . . . . .	<i>35</i>	
<i>170</i>		
\_ . . . . .		
<i>204</i>		
\  . . . . .		
<i>88</i>		



<b>V</b>		<b>X</b>		<b>Z</b>	
<code>\vbox</code> .....	127	<code>\x</code> .....	151		
<code>\vss</code> .....	129	<code>\xdef</code> .....	291, 295	<code>\z@</code> ..	159, 162, 163, 177