

**footmisc** —  
a portmanteau package  
for customising footnotes in L<sup>A</sup>T<sub>E</sub>X\*

Robin Fairbairns<sup>†</sup>

June 6, 2011

## Copyright statement

Program: `footmisc.dtx`

Copyright 1995 1996 1998 1999 2001–2003 2008 2009 Robin Fairbairns

This program is offered under the terms of the LaTeX Project Public License, version 1.3c of this license or (at your option) any later version. The latest version of this license is in <http://www.latex-project.org/lppl.txt>, and version 1.3c or later is part of all distributions of LaTeX version 2005/12/01 or later.

This work has the LPPL maintenance status ‘author-maintained’.

## History

This package originated as support of a personal project, which I was switching to L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> over the Christmas holiday period of 1993, using the first *β* release.

In its first form, it was known as the “footnote” package, but by the time I had released it to CTAN, that name had already been used by a package written by Mark Wooding. So the package is now known (as you can see) as “footmisc”.

## 1 User interface — package options

The `footmisc` package provides several different customisations of the way footnotes are represented in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> documents (the sources of the code in this package are various, but all of it has been massaged by the author; where the code comes from elsewhere, there are attributions given below, somewhere or other).

The interface to the package’s options is mostly rather simple — each one is presented as an option in the `\usepackage` command, and for most, nothing else needs to be done. For example, to use a useful and consistent set, the author invokes the package with the command `\usepackage[perpage,para,symbol*]{footmisc}`.

For a small number of options, there are additional parameters available; these are described in the subsections below.

---

\*This file has version number v5.5b, last revised 2011/06/06

<sup>†</sup>University of Cambridge Computer Laboratory, William Gates Building, J. J. Thompson Avenue, Cambridge, CB2 0FD, UK ([rf10@snail-shape.cam.ac.uk](mailto:rf10@snail-shape.cam.ac.uk))

## 1.1 Option perpage

This option resets footnote numbering for each page of the document. It needs at least two passes to do this correctly (though it comes as close as possible on the first pass). You generally have to make two passes with L<sup>A</sup>T<sub>E</sub>X anyway, to get the cross-references right, so an additional pass for this purpose shouldn't cause any additional problem. The option includes code to report that '*Label(s) may have changed*', which will help the poor user to realise that (yet) another run is in order.

## 1.2 Option para

This option (derived from code by Dominik Wujastyk and Chris Rowley) causes footnotes to be typeset as a single paragraph at the bottom of the page on which they occur. In the case that there is only one footnote on the page, no effect will be observed. However, if there are several footnotes on the page, they will be run together in the page foot, each introduced by its footnote mark. The original demand for the option came from the needs of those preparing critical editions; such documents typically have large numbers of small footnotes, which look ridiculous if each is typeset in a paragraph of its own; in most other disciplines, such multiplicities of footnotes represent mere self-indulgence: the author of this package is disgracefully guilty of this.

Please note that “old” L<sup>A</sup>T<sub>E</sub>X installations may have problems with the algorithm for `para` footnotes on very wide pages (for example, those used by the `a0poster` class). Recent L<sup>A</sup>T<sub>E</sub>X installations use an improved technique that is believed not to be susceptible to this problem.

## 1.3 Option side

This option (suggested by Frank Mittelbach) causes footnotes to be typeset using the `\marginpar` command: this has the advantage that the note appears close to its “call-up”, but has all the disadvantages associated with the `\marginpar` command (which consumes ‘float’ slots, and doesn't always place itself correctly at the top of pages in two-sided documents). Since the measure in which the footnote is to be typeset is likely to be pretty narrow, users of the `side` option are recommended also to use the `ragged` option, to avoid ugly spacing and line breaks.

There is a further problem (apart from the occasional failure to place the marginal note on the correct side of the page) in two-sided documents: one would like ‘raggedness’ to appear differently in different margins (setting the left, rather than the right, side ragged in the left margin). (The author would welcome suggestions on means of addressing the problem.)

## 1.4 Option ragged and \footnotelayout

The package provides facilities for ragged right setting of footnotes (so long as the `para` option isn't in effect). The change is effected by use of the command `\footnotelayout`; the package inserts this command into the start of the argument of `\footnotetext` (in effect: `\footnote` works, roughly, by calling the guts of `\footnotetext` at its end).

If you want to use some special effect other than ragged right, feel free to change `\footnotelayout` yourself: some intriguing (and completely undesirable) results are no doubt available. Change the setting simply by use of `\renewcommand\footnotelayout...`. The `ragged` option simply sets `\footnotelayout` to set `\raggedright` or `\RaggedRight` as appropriate. (If you intend to use the `ragged2e` package, load it before `footmisc` — if `footmisc` finds `\RaggedRight` is available, it automatically uses it in place of `\raggedright`.)

## 1.5 Option `symbol`

This option simply establishes that footnotes are ‘labelled’ by a symbol sequence. The command used is equivalent to that suggested in L<sup>A</sup>T<sub>E</sub>X manuals such as Lamport’s (the job performed by the option is very simple, and doesn’t really need a package).

Using symbols to ‘number’ your footnotes can be problematic: there is a limited number of symbols, and L<sup>A</sup>T<sub>E</sub>X will report an error if your footnotes exceed that limit. To avoid such problems, consider the `symbol*` option, or the `\setfnsymbol` command (see the next two sections), or number your footnotes by the page (see section 1.1).

## 1.6 Option `symbol*`

This is the `symbol` option, but with protection against the tedium that arises because of the instability of the `perpage` option. When executing the `perpage` option, the package often allocates footnotes to the wrong pages, only to correct itself on a later run (having warned the user of the need for the later run with a ‘*Label(s) may have changed*’ message). In these circumstances the `symbol` option is prone to producing L<sup>A</sup>T<sub>E</sub>X errors, which stop processing, and confound automatic generation procedures. In the same situation, the `symbol*` option produces information messages and a warning message at end document, and the user may scan the log for those messages *after* processing has stabilised. The option produces numbers (17 and higher, in the case of the default symbol set) in place of symbols, when the footnote number is too large.

## 1.7 The `\setfnsymbol` and `\DefineFNSymbols` commands

These commands permit the definition and use of alternative (ordered) sets of symbols for numbering footnotes. L<sup>A</sup>T<sub>E</sub>X of course comes with such a set ready-defined, but the choice of symbols isn’t universally loved.

You may define a set of symbols with the `\DefineFNSymbols` command. L<sup>A</sup>T<sub>E</sub>X’s default set would be defined by the command:

```
\DefineFNSymbols*{lamport}{*\dagger\ddagger\S\P\|}%
{**}{\dagger\dagger}{\ddagger\ddagger}}
```

Defined this way, the symbol set produces a “counter too large” error; a robust version of the set (cf. the `symbol*` option (see 1.6) using the `\DefineFNSymbols` command without the optional `*`. You may select a set of symbols by use of the `\setfnsymbol` command; so to restore use of the default set, you would type:

```
\setfnsymbol{lamport}
```

This package defines a small selection of alternative sets of symbols, using `\DefineFNsymbol`:

```
bringhurst * † ‡ § || ¶
chicago  * † ‡ § || #
wiley     * ** † ‡ § ¶ ||
```

together with a version of Lamport's original set that, with doubled versions of § and ¶, and tripled versions of everything but the vertical bars, provides a symbol range to cover counters up to 16.

This last set, known as `lamport*` is selected as the default symbol set by the package.

## 1.8 Option bottom

This option forces footnotes to the bottom of the page; this is only noticeably useful in case that `\raggedbottom` is in effect, when L<sup>A</sup>T<sub>E</sub>X would normally set the footnotes a mere `\skip\footins` distant from the bottom of the text.

There's a further infelicity in L<sup>A</sup>T<sub>E</sub>X's placing of footnotes of the bottom of pages: if a bottom float appears on a page, L<sup>A</sup>T<sub>E</sub>X places the footnote *above* it. The `bottom` option places the footnote at the foot of the page.

## 1.9 Option marginal

This option adjusts the position of footnote mark relative to the start of the line in which they appear (the the option is incompatible with option `para`, for obvious reasons).

When this option is in effect, the footnote is set `\footnotemargin` relative to the left margin of the page; the default setting for `\footnotemargin` is `-0.8em`, which means that the footnote mark will be set jutting `0.8em` into the margin. If `\footnotemargin` is a positive length, the footnote mark will be set with its right edge `\footnotemargin` from the margin. (In the absence of the option, `\footnotemargin` is set to `1.8em`; you may change that value with a `\setlength` command.)

## 1.10 Option flushmargin

This option is as option `marginal`, but sets the footnote marker flush with, but just inside the margin from, the text of the footnote.

## 1.11 Option hang

This option sets the footnote mark flush with the margin, and makes the body of the footnote hang at an indentation of `\footnotemargin` (if that is a positive distance), or the width of the marker (if `\footnotemargin ≤ 0`). The option code itself leaves `\footnotemargin` at its default value of `1.8em`.

The footnote itself may of course be longer than one paragraph; if so, the paragraphs will be separated by the vertical space specified by `\hangfootparskip`, and the second and subsequent paragraphs are indented by `\hangfootparindent`. Default values are:

```

\hangfootparskip    0.5\baselineskip
\hangfootparindent  0em

```

The user may redefine these values (using `\renewcommand`): it is best to use the font-size-dependent measures (multiples of `\baselineskip` for the skip, multiples of `em` for the indent). Note that the default has only one of the two values non-zero; both zero may result in easily-missed paragraph breaks, and both non-zero is not generally thought to be a good-looking option.

### 1.12 Option `norule`

This option suppresses the ‘normal’ footnote rule, and advances `\skip\footins` a bit to compensate

### 1.13 Option `splitrule`

This option makes puts a full-width rule above the split-off part of a split footnote. (Remember that split footnotes don’t happen if you’re doing paragraph footnotes.)

The option provides three different `\footnoterule` commands:

```

\mpfootnoterule    for use in minipages
\pagefootnoterule  for normal footnotes on regular pages
\splitfootnoterule for the tail of a split footnote

```

By default, `\mpfootnoterule` and `\pagefootnoterule` retain the original definition of `\footnoterule` (which may have been modified by a `norule` option), while `\splitfootnoterule` becomes a full-width rule.

### 1.14 The `stable` option

This option deals with the problem of placing footnotes in section titles (and so on). While there is (sometimes, just) justification for putting footnotes in titles,  $\LaTeX$ ’s treatment of the content of titles militates against them. Of course, the title argument is ordinarily a moving one, and `\footnote` is a fragile command, but the real problem comes from the way the argument actually moves — which is to two places. The argument moves to the table of contents, where the footnote will (at least) look odd. But the argument also moves to the marks that make up page headers, etc., and *there* it creates havoc, since page headers are executed in page make-up, and page make-up *must not* create footnotes.

If you use the `stable` option, the footnote won’t move to the table of contents or the page headers, but it will be typeset correctly within the title itself.

The situation with `\footnotemark` is less dire (it could in principle appear in page headers, for example); footnote marks appearing on pages other than where their text appears are none the less confusing, and the `stable` option treats `\footnotemark` in the same way that it treats `\footnote`.

### 1.15 The `multiple` option

This option deals with the case where the author needs to type things like

```

mumble\footnote{blah}\footnote{grumble}

```

Without special treatment, L<sup>A</sup>T<sub>E</sub>X would output something like

mumble<sup>1314</sup>

What the `multiple` option makes of the above is

mumble<sup>13,14</sup>

which is what most people would expect. The comma separator actually derives from the definition of `\multfootsep`, which may be changed by `\renewcommand` if the option is in effect.

The option also treats `\footnotemark` in the same way.

## 1.16 User interface — miscellaneous commands

The package also defines some miscellaneous footnote-related commands. The present group provides alternative means of producing footnote marks: `\footref` and `\mpfootnotemark`.

When you're in a minipage, `\footnote` numbers run according to the minipage's own footnote counter, and the marks are set in italic letters. However, the numbers used by `\footnotemark` make reference to the 'main' footnote counter, and are set in whatever is the current style for that: this behaviour often surprises, and there's no obvious way in standard L<sup>A</sup>T<sub>E</sub>X to "get around" it. The command `\mpfootnotemark` gets around this problem in a minipage, by generating footnote marks in the same way as those used by `\footnote`.

In fact, making reference to footnotes in general can be problematic: it can be done by noting down the value of the footnote marker in a counter (or the like) and then using the value in a subsequent `\footnotemark` or `\mpfootnotemark`. This is a tedious way of going about things, and doesn't allow representation of all possible forms of footnote mark; `\footref` is a form of reference command that sets the reference as if it were a footnote. The label should be set *within* the argument of the footnote command that is being labelled:

```
...\footnote{Note text\label{fnlabel}}
...
... potato head\footref{fnlabel}
```

## 2 User interface — interactions with other packages

The `footmisc` package modifies several parts of the L<sup>A</sup>T<sub>E</sub>X kernel; what gets modified depends on the options you select. This behaviour can cause problems with other packages, particularly those that also modify the kernel.

Known interactions are:

`setspace` The `setspace` package modifies the way line spacing is calculated in footnotes. `Footmisc` knows about this, and preserves the change. However, you *must* load `setspace` *before* `footmisc`.

`memoir class` The class emulates `setspace`, and we detect that emulation and deal with it in the same way as `setspace`.

`hyperref` The `hyperref` package has ambitions to make hyperlinks from footnote marks to the corresponding footnote body; naturally this causes grief to `footmisc`, and unfortunately no remedy is currently known. If you use `footmisc`, suppress `hyperref`'s hyper-footnotes, by loading it as:

```
\usepackage[hyperfootnotes=false,...]{hyperref}
```

Further work on the interaction between the two packages is proposed, but not yet scheduled.

`manyfoot` The `manyfoot` package permits several independent sequences of footnotes. Some preliminary work towards interworking with `footmisc` has been completed, but more remains to be done at the time of writing.

### 3 Code: Preliminaries

Well — here we go: let's make the package file:

```
1 (*package)
```

Now declare what environment we need:

```
2 \NeedsTeXFormat{LaTeX2e}[1994/12/01]
```

We need a token register in case we have to patch `\@makecol`:

```
3 \newtoks\FN@temptoken
```

`\protected@writeaux` This command is defined for future compatibility with Matt Swift's `newclude` package (still, after all this time, not out of beta status).

```
4 \providecommand\protected@writeaux{%
```

```
5   \protected@write\@auxout
```

```
6 }
```

`\l@advance@macro` We make the following (`\@@dvance@macro`) generalisable as follows (the global form isn't used in this package ... yet):

```
@@dvance@macro
```

```
@advance@macro
```

```
7 \def\l@advance@macro{\@@dvance@macro\edef}
```

```
8 \def\@@dvance@macro#1#2#3{\expandafter\@tempcnta#2\relax
```

```
9   \advance\@tempcnta#3\relax
```

```
10   #1#2{\the\@tempcnta}%
```

```
11 }
```

Now we define a jolly little macro to advance a macro count (`#1`) by a given amount (`#2`).

```
12 \let\@advance@macro\l@advance@macro
```

`\ifFN@etex` Check whether we're using `etex`

```
13 \newif\ifFN@etex
```

```
14 \ifx\dimexpr\undefined
```

```
15   \FN@etexfalse
```

```
16 \else
```

```
17   \FN@etextrue
```

```
18 \fi
```

`\footnotemargin` Finally, we define the length used by the `marginal` option, and initialise it as if we've not had the option.

```
19 \newdimen\footnotemargin
```

```
20 \footnotemargin1.8em\relax
```

## 4 Package options

Most of the code of the package is contained within the option processing, one way or another (that which isn't, is executed after `\ProcessOptions` as a result of flags set in the option processing).

### 4.1 The `symbol` option

This is a declaration that appears in the original L<sup>A</sup>T<sub>E</sub>X book. Since it appeared in the old `pagefoots.sty` (presumably since it goes so naturally with the `perpage` option), I've added this trivial piece of customisation to the package.

```
21 \DeclareOption{symbol}{\renewcommand\thefootnote{\fnsymbol{footnote}}}
```

### 4.2 The `symbol*` option

The robust version of the `symbol` option: if the current 'symbol' option doesn't provide enough variants, use arabic footnote number. We use a robust version of the "extended ordinary" symbol set, described later (in section 1.7).

```
22 \DeclareOption{symbol*}{%
23   \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
24   \AtEndOfPackage{\setfnsymbol{\lamport*-robust}}%
25 }
```

### 4.3 The `para` option

The basis of the code for this option comes from T<sub>E</sub>Xbook, p.398 ff. ("Dirty Tricks"), though it does (of course) avoid redefining `\` which has some other (somewhat significant) uses in L<sup>A</sup>T<sub>E</sub>X! The user should be aware of Knuth's note on the limitations of this method of doing the job: the T<sub>E</sub>X stack is used four times per footnote, and the stack is limited (see the T<sub>E</sub>Xbook, p.300 ff.). If you have very large numbers of footnotes (in the hundreds), and encounter the error "`! TeX capacity exceeded, sorry (... save size ...)`", you may need to break your text into smaller sections and compile the separately. Fortunately (say the comments on the original `fnpara.sty`) this is very easy to do with L<sup>A</sup>T<sub>E</sub>X, provided that you reset the footnote counter to make the joins seamless.

`\ifFN@para` Define the `para` option: now simply sets a marker for use later when defining the option's auxiliary code and when patching the output routine and so on.

```
26 \newif\ifFN@para \FN@parafalse
27 \DeclareOption{para}{\ifFN@sidefn
28   \PackageError{footmisc}{Option "\CurrentOption" incompatible with
29     option "side"}%
30   {I shall ignore "\CurrentOption"}}%
31 \else
32   \FN@paratrue
33 \fi
34 }
```

#### 4.4 The side option

`\ifFN@sidefn` Simply changes the behaviour of `\@footnotetext`; incompatible with paragraph footnotes.

```
35 \newif\ifFN@sidefn \FN@sidefnfalse
36 \DeclareOption{side}{\ifFN@para
37   \PackageError{footmisc}{Option "\CurrentOption" incompatible with
38     option "para"}%
39   {I shall ignore "\CurrentOption"}%
40 \else
41   \FN@sidefntrue
42 \fi
43 }
```

#### 4.5 The ragged option

`\footnotelayout` A very simple option that merely changes the definition of one macro. Note detection of the presence of the `ragged2e` package.

```
44 \let\footnotelayout\@empty
45 \DeclareOption{ragged}{%
46   \@ifundefined{RaggedRight}%
47   {\renewcommand\footnotelayout{\linepenalty50 \raggedright}}%
48   {\renewcommand\footnotelayout{\linepenalty50 \RaggedRight}}%
49 }
```

#### 4.6 The perpage option

`\ifFN@perpage` A footnote-numbering modification: a new algorithm replacing one from Brian T. Schellenberger, which has proved to be flawed. We simply set a marker here, and define code later depending on the state of the marker (see section 5.4).

```
50 \newif\ifFN@perpage
51 \FN@perpagefalse
52 \DeclareOption{perpage}{%
53   \FN@perpagetrue
54 }
```

#### 4.7 The PPdebug option

`\ifFN@pp@debug` Sets a flag; the messages are generated in various places throughout the code. The option is not available in the package as distributed: modify the `.ins` file to generate a version of the package that includes the option, if you feel you need it.

```
55 (*PPdebug)
56 \newif\ifFN@pp@debug \FN@pp@debugfalse
57 \DeclareOption{PPdebug}{\FN@pp@debugtrue}
58 </PPdebug>
```

#### 4.8 The bottom option

`\ifFN@bottom` All this needs to do is to set a flag to say that it should happen

```
59 \newif\ifFN@bottom \FN@bottomfalse
60 \DeclareOption{bottom}{%
61   \FN@bottomtrue
```

```
62 }
```

## 4.9 The marginal option

Again, the processing of the option is pretty trivial:

```
63 \DeclareOption{marginal}{%
64   \footnotemargin-0.8em\relax
65 }
```

## 4.10 The flushmargin option

Again, the processing of the option is pretty trivial:

```
66 \DeclareOption{flushmargin}{%
67   \footnotemargin0pt\relax
68 }
```

## 4.11 The hang option

`\ifFN@hangfoot` We need a switch, since `\@makefn` needs to be patched.

```
69 \newif\ifFN@hangfoot \FN@hangfootfalse
70 \DeclareOption{hang}{%
71   \FN@hangfoottrue
72 }
```

`\hangfootparskip` Layout parameters for hanging footnotes; `\hangfootparskip` and `\hangfootparindent`  
`\hangfootparindent` are (respectively) values to use for `\parskip` and `\parindent` when in hanging footnotes.

```
73 \newcommand*\hangfootparskip{0.5\baselineskip}
74 \newcommand*\hangfootparindent{0em}%
```

## 4.12 The norule option

Pretty simple too...

```
75 \DeclareOption{norule}{%
76   \renewcommand\footnoterule{}%
77   \advance\skip\footins 4\p@\@plus2\p@\relax
78 }
```

## 4.13 The splitrule option

`\split@prev` This is from a posting by Donald Arseneau dated 13 November 1996. The code relies on the fact that L<sup>A</sup>T<sub>E</sub>X only uses inserts for footnotes, so that if any insert is going to be split, it's going to be a footnote.

```
79 \DeclareOption{splitrule}{%
80   \gdef\split@prev{0}
```

`\pagefootnoterule` Define defaults for the three footnote rules: note, we inherit the current state of  
`\mpfootnoterule` `\footnoterule` for the two 'regular' footnote defaults, and if we've been preceded  
`\splitfootnoterule` by option `norule`, they will both become null...

```
81 \let\pagefootnoterule\footnoterule
82 \let\mpfootnoterule\footnoterule
83 \def\splitfootnoterule{\kern-3\p@ \hrule \kern2.6\p@}
```

Now redefine `\footnoterule` to distinguish the three situations.

```
84 \def\footnoterule{\relax
85   \ifx \@listdepth\@mplistdepth
      In a minipage
86     \mpfootnoterule
87   \else
88     \ifnum\split@prev=\z@
      Normal footnote on a regular page
89       \pagefootnoterule
90     \else
      Second part of a split footnote
91       \splitfootnoterule
92     \fi
      Remember a split for next page
93     \xdef\split@prev{\the\insertpenalties}%
94   \fi
95 }%
96 }
```

#### `\ifFN@stablefootnote` **4.14 The stable option**

Simply set a flag: the code of this gets executed at the very end of the package.

```
97 \newif\ifFN@stablefootnote \FN@stablefootnotefalse
98 \DeclareOption{stable}{\FN@stablefootnotetrue}
```

#### **4.15 The multiple option**

`\ifFN@multiplefootnote` Again, simply set a flag, for code that gets executed at the very very very end of the package.

```
99 \newif\ifFN@multiplefootnote \FN@multiplefootnotefalse
100 \DeclareOption{multiple}{\FN@multiplefootnotetrue}
```

#### **4.16 The start of the endgame**

Exercise the options that the user has requested...

```
101 \ProcessOptions
```

## **5 Hacking kernel commands**

Various standard commands (some of them internal ones) need to be hacked to achieve our effects, and we do all of this now, according to flags set in option processing.

## 5.1 The output routine

Now; do we need to mess about with the output routine? If either `para` or `bottom` has been invoked, we do.

```
102 \let \if@tempwa \ifFN@bottom
103 \ifFN@para \@tempwatrue \fi
104 \if@tempwa
```

... so we've patching to do.

First, we ensure that `\@makecol` is as expected from the time at which these macros were written: since we're going to patch it, we had better be sure that we're patching the right thing. (There was a minuscule change to the definition 1999, but this doesn't as far as I can tell make any difference to the semantics of the definition we base our patch on.)

```
105 \@ifl@t@r\fmtversion{2005/12/01}{%
106   \CheckCommand*\@makecol{\ifvoid \footins
107     \setbox\@outputbox \box\@cclv
108   \else
109     \setbox\@outputbox \vbox{%
110       \boxmaxdepth \@maxdepth
111       \unvbox\@cclv
112       \vskip \skip\footins
113       \color@begingroup
114         \normalcolor\footnoterule
115         \unvbox\footins
116       \color@endgroup
117     }%
118   \fi
119   \let \@elt \relax
120   \xdef\@freelist{\@freelist\@midlist}%
121   \global\let\@midlist\@empty
122   \@combinefloats
123   \ifvbox\@kludgeins
124     \@makespecialcolbox
125   \else
126     \setbox\@outputbox \vbox to\@colht{%
127       \@texttop \dimen@ \dp\@outputbox
128       \unvbox\@outputbox
129       \vskip -\dimen@\@textbottom
130     }%
131   \fi
132   \global\maxdepth\@maxdepth
133 }
134 }-%
135 \@ifl@t@r\fmtversion{2003/12/01}{%
136   \CheckCommand*\@makecol{\ifvoid \footins
137     \setbox\@outputbox \box\@cclv
138   \else
139     \setbox\@outputbox \vbox{%
140       \boxmaxdepth\@maxdepth
141       \@tempdima \dp\@cclv
142       \unvbox\@cclv
143       \vskip \skip\footins
144     \color@begingroup
```

```

145         \normalcolor
146         \footnoterule
147         \unvbox\footins
148     \color@endgroup
149 }%
150 \fi
151 \let \@elt \relax
152 \xdef\@freelist{\@freelist\@midlist}%
153 \global\let\@midlist\@empty
154 \@combinefloats
155 \ifvbox\@kludgeins
156     \@makespecialcolbox
157 \else
158     \setbox\@outputbox \vbox to\@colht{%
159         \@texttop
160         \dimen@\dp\@outputbox
161         \unvbox\@outputbox
162         \vskip -\dimen@
163         \@textbottom
164     }%
165 \fi
166 \global\maxdepth\@maxdepth
167 }%
168 }{%
169 \ifl@t@r\fmtversion{1999/12/01}{%
170     \CheckCommand*\@makecol{\ifvoid \footins
171         \setbox\@outputbox \box\@cclv
172     \else
173         \setbox\@outputbox \vbox{%
174             \boxmaxdepth\@maxdepth
175             \@tempdima\dp\@cclv
176             \unvbox\@cclv
177             \vskip \skip\footins
178             \color@begingroup
179             \normalcolor\footnoterule
180             \unvbox\footins
181             \color@endgroup
182         }%
183     \fi
184     \xdef\@freelist{\@freelist\@midlist}%
185     \global\let\@midlist\@empty
186     \@combinefloats
187     \ifvbox\@kludgeins
188         \@makespecialcolbox
189     \else
190         \setbox\@outputbox \vbox to\@colht{%
191             \@texttop \dimen@\dp\@outputbox
192             \unvbox\@outputbox
193             \vskip -\dimen@\@textbottom
194         }%
195     \fi
196     \global\maxdepth\@maxdepth
197 }%
198 }%

```

```

199 }{%
200   \CheckCommand*\@makecol{\ifvoid \footins
201     \setbox\@outputbox \box\@cclv
202   \else
203     \setbox\@outputbox \vbox{%
204       \boxmaxdepth\@maxdepth
205       \unvbox\@cclv
206       \vskip \skip\footins
207       \color@begingroup
208         \normalcolor\footnoterule
209         \unvbox\footins
210       \color@endgroup
211     }%
212   \fi
213   \xdef\@freelist{\@freelist\@midlist}%
214   \global\let\@midlist\@empty
215   \@combinefloats
216   \ifvbox\@kludgeins
217     \@makespecialcolbox
218   \else
219     \setbox\@outputbox \vbox to\@colht{%
220       \@texttop \dimen@\dp\@outputbox
221       \unvbox\@outputbox
222       \vskip -\dimen@\@textbottom
223     }%
224   \fi
225   \global\maxdepth\@maxdepth
226 }%
227 }%
228 }

```

If we're doing paragraph footnotes, the output routine needs different code to place the actual text. We prepare this code here, since it's potentially used in two different places.

We prepare the code in a token register to be used at the appropriate place in the patching of `\@makecol`; thus it becomes a token register containing code to place stuff in a token register

```
229 \ifFN@para
```

We make a box out of the paragraph of footnotes, and then stuff the contents of the box into that which is going to be `\shipped out`.

```

230   \FN@temptoken{%
231     \toks@\expandafter{\the\toks@
232     \vskip\skip\footins
233     \color@begingroup
234       \normalcolor\footnoterule
235     \global\setbox\FN@tempboxc\vbox{\makefootnoteparagraph}%
236     \unvbox\FN@tempboxc
237     \color@endgroup
238   }%
239 }%

```

If we're not doing paragraph footnotes, we insert the little bit of code that would have been replaced by the stuff above:

```
240 \else
```

```

241 \FN@temptoken{%
242 \toks@\expandafter{\the\toks@
243 \vskip\skip\footins
244 \color@begingroup
245 \normalcolor\footnoterule
246 \unvbox\footins
247 \color@endgroup
248 }%
249 }%
250 \fi

```

Now we start building up the revised version of `\@makecol`. The definition starts out in `\toks@`; first the `bottom` version:

```

251 \ifFN@bottom
252 \toks@{\setbox\@outputbox \box\@cclv
253 \xdef\@freelist{\@freelist\@midlist}%
254 \global\let\@midlist\@empty
255 \@combinefloats
256 \ifvoid\footins
257 \else
258 \setbox\@outputbox \vbox\bgroup
259 \boxmaxdepth\@maxdepth
260 \unvbox\@outputbox
261 \vfill\relax
262 }
263 \the\FN@temptoken
264 \toks@\expandafter{\the\toks@\egroup\fi}

```

Not putting stuff at the bottom: footnotes are placed using the kernel's algorithm.

```

265 \else
266 \toks@{\ifvoid\footins
267 \setbox\@outputbox\box\@cclv
268 \else
269 \setbox\@outputbox \vbox\bgroup
270 \boxmaxdepth\@maxdepth
271 \unvbox\@cclv
272 }
273 \the\FN@temptoken

```

Finally, close the `\setbox` and the `\ifvoid` and tag the parts of the definition of `\@makecol` up to the end of the definition of the `bottom` version on to `\toks@`.

```

274 \toks@\expandafter{\the\toks@
275 \egroup
276 \fi
277 \xdef\@freelist{\@freelist\@midlist}%
278 \global\let\@midlist\@empty
279 \@combinefloats
280 }%
281 \fi

```

Finally, create the new definition from the resulting object with the remainder of the original `\@makecol` tagged on at the end.

```

282 \toks@\expandafter{\the\toks@
283 \ifvbox\@kludgeins

```

```

284     \@makespecialcolbox
285   \else
286     \setbox\@outputbox \vbox to\@colht{%
287       \@texttop \dimen@\dp\@outputbox
288       \unvbox\@outputbox
289       \vskip -\dimen@\@textbottom
290     }%
291   \fi
292   \global\maxdepth\@maxdepth
293 }
294 \edef\@makecol{\the\toks@}

```

All of the above occurred conditionally on the ‘or’ of `\ifFN@para` and `\ifFN@bottom`, so we now close the conditional.

```
295 \fi
```

## 5.2 The requirements of `\@footnotetext`

`\ifFN@baselinestretch` Whatever we do, we are going to patch `\@footnotetext`; so first of all, we’ll check `\FN@singlespace` it’s not been hacked by anyone other than `setspace.sty` (while we’re at it we also record whether `setspace` is loaded). so we do this here:

```

296 \newif\ifFN@setspace
297 \@ifpackageloaded{setspace}{%
298   \FN@setspacetrue
299   \@ifclassloaded{memoir}{%
    we’re seeing memoir’s emulation of setspace
300     \let\FN@baselinestretch\m@m@singlespace
301   }{%
    we’re seeing setspace in its own right
302     \let\FN@baselinestretch\setspace@singlespace
303   }%
304 }{%
305   \FN@setspacefalse
306 }

```

There’s substantial patching to be done if we’re doing paragraph footnotes:

```

307 \ifFN@para
308   \renewcommand\@footnotetext[1]{%
309     \insert\footins{%
    insert compatibility code with setspace.sty if necessary
310       \ifFN@setspace
311         \let\baselinestretch\FN@baselinestretch
312       \fi
313       \reset@font\footnotesize
314       \interlinepenalty\interfootnotelinepenalty
315       \splittopskip\footnotesep
316       \splitmaxdepth \dp\strutbox
317       \floatingpenalty\@MM
318       \hsize\columnwidth
319       \@parboxrestore
320       \protected@edef\@currentlabel{\csname p@footnote\endcsname\@thefnmark}%
321       \color@begingroup

```

We set the paragraph in an `\hbox` and apply the fudge factor here:

```
322     \setbox\FN@tempboxa=\hbox{%
```

This needs a parameter; the rule should be moved to the beginning of the footnote paragraph, but the `\ignorespaces` should be left here.

```
323     \@makefntext{\ignorespaces#1\strut
```

We insert a penalty here to help line breaking in the footnote paragraph; the value is taken from the `TEXbook`.

```
324         \penalty-10\relax
325         \hskip\footglue
326     }% end of \@makefntext parameter
327 }% end of \hbox
328 \dp\FN@tempboxa=0pt
329 \ifFN@etex
330     \ht\FN@tempboxa=\dimexpr\wd\FN@tempboxa *
331                     \footnotebaselineskip / \columnwidth\relax
332 \else
333     \ht\FN@tempboxa=\fudgefactor\wd\FN@tempboxa
334 \fi
335 \box\FN@tempboxa
336 \color@endgroup
337 }%
338 \FN@mf@prepare
339 }
```

If we're not doing paragraph footnotes, we now simply tag a `\FN@mf@prepare` command on the end of the definition; of course, there are different definitions according as whether we're using *side* footnotes...

```
340 \else
341     \ifFN@sidefn
342         \renewcommand\@footnotetext[1]{%
343             \marginpar{%
insert compatibility code with setspace.sty if necessary
344                 \ifFN@setspace
345                     \let\baselinestretch\FN@baselinestretch
346                 \fi
347                 \reset@font\footnotesize
348                 \protected@edef\@currentlabel{%
349                     \csname p@footnote\endcsname\@thefnmark
350                 }%
351                 \color@begingroup
352                 \@makefntext{%
353                     \ignorespaces#1%
354                 }%
355                 \color@endgroup
356             }%
357             \FN@mf@prepare
358         }%
359 \else
360     \renewcommand\@footnotetext[1]{%
361         \insert\footins{%
```

insert compatibility code with setspace if necessary

```
362     \ifFN@setspace
363         \let\baselinestretch\FN@baselinestretch
364     \fi
365     \reset@font\footnotesize
366     \interlinepenalty\interfootnotelinepenalty
367     \splittopskip\footnotesep
368     \splitmaxdepth \dp\strutbox
369     \floatingpenalty\@MM
370     \hsize\columnwidth
371     \@parboxrestore
372     \protected@edef\@currentlabel{%
373         \csname p@footnote\endcsname\@thefnmark
374     }%
375     \color@begingroup
376     \@makefnmark{%
377         \rule{z}\footnotesep
378         \ignorespaces#1\@finalstrut\strutbox
379     }%
380     \color@endgroup
381 }%
382 \FN@mf@prepare
383 }%
384 \fi
385 \fi
```

### 5.3 Support code for paragraph footnotes

This code used (most inefficiently) to be in the argument of the `\DeclareOption`; this no doubt comes of that code having been written over Christmas 1993...

Now all executed under the `para` conditional set in the option declaration.

```
386 \ifFN@para
```

```
\FN@tempboxa We need some temporary boxes, and LATEX only defines one
\FN@tempboxb 387 \let\FN@tempboxa\@tempboxa
\FN@tempboxb 388 \newbox\FN@tempboxb
389 \newbox\FN@tempboxc
```

```
\footglue A direct crib from the TEXbook:
```

```
390 \newskip\footglue \footglue=1em plus.3em minus.3em
```

```
\@makefnmark The standard classes set the footnote mark flush with the text of the footnote, but that's not appropriate for paragraph footnotes, we find.
```

There's not much point in patching this code from the original, since the only things it has in common with the original are the footnote mark and the footnote text (which last is the argument). Note that the `\leavevmode` isn't necessary except in the case of footnotes in minipages, which otherwise end up with the `\@makefnmark` being executed in restricted vertical mode, which results in its `\hbox` ending up in a line of its own.

```
391 \long\def\@makefnmark#1{\leavevmode
392     \@makefnmark\nobreak
393     \hskip.5em\relax#1%
394 }
```

`\footnotebaselineskip` We need to record a value for the baseline skip when in footnotes:

```
395 \newdimen\footnotebaselineskip
396 {%
397   \footnotesize
398   \global
399   \footnotebaselineskip=\normalbaselineskip
400 }
```

`\fudgefactor` Now we derive a fudge factor from the `baselineskip` we've just established (we use `\dimexpr` if we're in `etex`, so there's no need for the fudge factor in that case).

```
401 \ifFN@etex
402 \else
403   \@tempdima=\footnotebaselineskip \multiply\@tempdima by 1024
404   \divide \@tempdima by \columnwidth \multiply\@tempdima by 64
405   \xdef\fudgefactor{\strip@pt\@tempdima }%
406 \fi
```

`\makefootnoteparagraph` For use in the output routine

```
407 \long\def\makefootnoteparagraph{\unvbox\footins \makeboxoffhboxes
408   \setbox\FN@tempboxa=\hbox{\unhbox\FN@tempboxa \removehboxes}
  Now we are ready to set the paragraph:
409   \hsize\columnwidth
410   \@parboxrestore
411   \baselineskip=\footnotebaselineskip
412   \noindent
413   \rule{\z@}{\footnotesep}%
414   \unhbox\FN@tempboxa\par
415 }
```

`\makeboxoffhboxes` Support code for `\makefootnoteparagraph`

```
\removehboxes 416 \def\makeboxoffhboxes{\setbox\FN@tempboxa=\hbox{}%
417   \loop
418     \setbox\FN@tempboxb=\lastbox
419     \ifhbox\FN@tempboxb
420     \setbox\FN@tempboxa=\hbox{\box\FN@tempboxb\unhbox\FN@tempboxa}%
421     \repeat
422   }
423 \def\removehboxes{\setbox\FN@tempboxa=\lastbox
424   \ifhbox
425     \FN@tempboxa{\removehboxes}%
426     \unhbox\FN@tempboxa
427   \fi
428 }
429 \fi
```

## 5.4 The other footnote commands

`\ifFN@pp@footnotehint` A conditional needed by the `perpage` code: must be defined outside the `perpage` conditional

```
430 \newif\ifFN@pp@footnotehint
```

`\c@pp@next@reset` Counter used to store information about the next reset of the footnote number, in perpage mode.

```
431 \newcounter{pp@next@reset}%
```

`\ifFN@pp@towrite` A conditional that mediates the interaction between the `perpage` option and the `multiple` option.

```
432 \newif\ifFN@pp@towrite
433 \FN@pp@towritefalse
```

`\ifFN@pp@lastseq` We prevent endless processions of diagnostics ‘footnote sequence lost’ by using this conditional (again, needs to be defined outside the `perpage` conditional:

```
434 (*PPdebug)
435 \newif\ifFN@pp@lastseq
436 \global\FN@pp@lastseqfalse
437 </PPdebug)
```

Now, do we need to patch `\footnote` for per-page footnotes?

```
438 \ifFN@perpage
439   \CheckCommand*\footnote{\@ifnextchar [%]
440     \xfootnote
441     {%
442       \stepcounter\@mpfn \protected@xdef\@thefnmark{\thempfn}%
443       \@footnotemark \@footnotetext
444     }%
445   }
446   \renewcommand*\footnote{\@ifnextchar [%]
447     \xfootnote
448     {%
449       \stepcounter\@mpfn \protected@xdef\@thefnmark{\thempfn}%
450       \FN@pp@footnote\@footnotemark
```

In case that we’re *not* running `multiple` option, `\@footnotemark` *won’t* have written details to the `.aux` file, so do it now:

```
451   \ifFN@pp@towrite
452     \FN@pp@writetemp
453     \FN@pp@towritefalse
454   \fi
455   \@footnotetext
456 }%
457 }
```

And the analogous change for `\footnotemark`

```
458 \CheckCommand*\footnotemark{%
459   \@ifnextchar [%]
460   \xfootnotemark
461   {%
462     \stepcounter{footnote}%
463     \protected@xdef\@thefnmark{\thefootnote}%
464     \@footnotemark
465   }%
466 }
467 \renewcommand*\footnotemark{%
468   \@ifnextchar [%]
469   \xfootnotemark
```

```

470   {%
471     \stepcounter{footnote}%
472     \protected@xdef\@thefnmark{\thefootnote}%
473     \FN@pp@footnote\@footnotemark
      again, tidy up if we're not doing multiple option
474     \ifFN@pp@towrite
475       \FN@pp@writetemp
476       \FN@pp@towritefalse
477     \fi
478   }%
479 }

```

`\FN@pp@initialstab` Now the supporting commands. . .

if we encounter no information in the `.aux` file, we make a first stab resetting footnote on the page number counter

```

480 \gdef\FN@pp@initial@stab{\@addtoreset{footnote}{page}}
481 \AtBeginDocument{\FN@pp@initial@stab}

```

We use a counter to keep pace with the footnotes: this counter is used in the data that's written to the `.aux` file, and matched to create the correct footnote numbers on the second and subsequent passes.

```

482 \newcounter{fnserial}

```

`\FN@pp@cpage` The package requires a “knowledge” of the current page number. It's kept in `\FN@pp@cpage`

```

483 \def\FN@pp@cpage{0}

```

`\footnotehint` However, the progress of page numbers isn't predictable, so we have a flag saying ‘reset footnote number’. The flag is for indirect use by people who diddle with the page number, via the `\footnotehint` command, as well as various places where we *know* there could be a discontinuity.

```

484 \FN@pp@footnotehinttrue
485 \newcommand{\footnotehint}{%
486   \setcounter{footnote}{0}%
487   \protected@writeaux\relax{\protect\FN@pp@footnotehinttrue}%
488   \@tempcnta\c@fnserial
489   \advance\@tempcnta\ne
490   \global\c@pp@next@reset\@tempcnta
491 }
492 \AtBeginDocument{\protected@writeaux\relax{%
493   \protect\providecommand{\protect\FN@pp@footnotehinttrue}{}%
494 }%
495 }

```

`\FN@pp@lastfoot` Dummy value for the number of the last footnote we came across.

```

496 \def\FN@pp@lastfoot{-1}

```

`\FN@pp@footnote@aux` The command `\FN@pp@footnote@aux` is written to the `.aux` file for every footnote counter allocated (other than in `minipages`):

**#1** is the footnote serial number

**#2** is the page the footnote was actually written on

```

497 \newcommand{\FN@pp@footnote@aux}[2]{%

```

```

498 \ifnum\FN@pp@lastfoot<#1
499 \ifFN@pp@footnotehint
500 \FN@pp@resetfn{#1}{#2}%
501 \FN@pp@footnotehintfalse
502 \else
503 \gdef\@tempa{#2}%
504 \ifx\@tempa\FN@pp@cpage
505 \else
506 \FN@pp@resetfn{#1}{#2}%
507 \fi
508 \fi
509 \def\FN@pp@lastfoot{#1}%
510 (*PPdebug)
511 \else
512 \ifFN@pp@debug
513 \typeout{not considering footnote serial number #1
514 (last valid was \FN@pp@lastfoot)}%
515 \fi
516 (/PPdebug)
517 \fi

```

Since we've now had at least one item of footnote information from a .aux file, we can't allow footnote resetting per page.

```

518 \global\let\FN@pp@initial@stab\relax
519 }

```

**\FN@pp@resetfn** Set the flag to reset the footnote number; this constructs a chain through the footnote serial numbers at the start of each page

```

520 \newcommand{\FN@pp@resetfn}[2]{%
521 \gdef\FN@pp@cpage{#2}%
522 \expandafter\gdef
523 \csname FN@pp@next-\FN@pp@prev@foot\endcsname{#1}%
524 \def\FN@pp@prev@foot{#1}%
525 \expandafter\xdef
526 \csname FN@pp@next-\FN@pp@prev@foot\endcsname{\the\@MM}%
527 }

```

**\FN@pp@prev@foot** The base of the footnote serial number chain (this element is never looked at: footnote serial 1 must always have number 1)

```

528 \newcommand{\FN@pp@prev@foot}{root}

```

again, avoid confusion; also establish \FN@pp@footnote@aux in the .aux file

```

529 \AtBeginDocument{\protected@writeaux\relax{%
530 \protect\providecommand{\protect\FN@pp@footnote@aux}[2]{}}%
531 \c@pp@next@reset@one
532 }

```

At end document, establish a new \FN@pp@footnote@aux which checks whether numbers have changed during the scan of the .aux files for changed labels and the like

```

533 \AtEndDocument{\let\FN@pp@footnote@aux\FN@pp@footnote@endaux
534 \def\FN@pp@lastfoot{-1}%
535 \FN@pp@footnotehintfalse
536 \renewcommand{\FN@pp@prev@foot}{root}
537 }

```

`\FN@pp@footnote@endaux` The footnote analysis command for end document

```
538 \newcommand{\FN@pp@footnote@endaux}[2]{%
539 \ifnum\FN@pp@lastfoot<#1
540 \ifFN@pp@footnotehint
541 \FN@pp@resetfn@end{#1}{#2}%
542 \FN@pp@footnotehintfalse
543 \else
544 \gdef\@tempa{#2}%
545 \ifx\@tempa\FN@pp@cpage
546 \else
547 \FN@pp@resetfn@end{#1}{#2}%
548 \fi
549 \fi
550 \def\FN@pp@lastfoot{#1}%
551 (*PPdebug)
552 \else
553 \ifFN@pp@debug
554 \typeout{not considering footnote serial number #1
555 (last valid was \FN@pp@lastfoot)}%
556 \fi
557 </PPdebug)
558 \fi
559 }
```

`\FN@pp@resetfn@end` Deal with the .aux file footnote details, at end document

```
560 \newcommand{\FN@pp@resetfn@end}[2]{%
561 \def\@tempa{#1}%
562 \expandafter\ifx\csname FN@pp@next-\FN@pp@prev@foot\endcsname\@tempa%
563 \else
564 \@tempswatruue
565 (*PPdebug)
566 \ifFN@pp@debug
567 \expandafter\ifx\csname FN@pp@next-\FN@pp@prev@foot\endcsname\relax
568 \ifFN@pp@lastseq\else
569 \typeout{footnote sequence lost between pages
570 \FN@pp@cpage\space and #2}%
571 \global\FN@pp@lastseqtrue
572 \fi
573 \else
574 \typeout{footnotes changed between pages \FN@pp@cpage\space and #2:
575 next was \csname FN@pp@next-\FN@pp@prev@foot\endcsname, now #1}%
576 \global\FN@pp@lastseqfalse
577 \fi
578 \fi
579 </PPdebug)
580 \fi
581 \gdef\FN@pp@prev@foot{#1}%
582 \gdef\FN@pp@cpage{#2}%
583 }
```

`\clearpage` Now, how do we cope with `\included` documents? We can't insert anything (the `\FN@pp@clearpage` `includex` package offers `\AtBeginIncludedDocument` but there's no corresponding command for files included by the kernel `\include`. So we insert a footnote hint at every `\clearpage`

```

584 \let\FN@pp@@clearpage\clearpage
585 \renewcommand{\clearpage}{\footnotehint\FN@pp@@clearpage}

```

\FN@pp@footnote The business end of the option: a macro to decide on footnote numbers, called from \footnote and \footnotemark (see above).

```

586 \def\FN@pp@footnote{%
587   \if@minipage\else
588     \global\advance\c@@fnserial\@ne
589     \if@filesw

```

In case we're also doing multiple option, we now save up the command to write to the .aux file, and mark we've done so

```

590     \xdef\FN@pp@writetemp{%
591       \noexpand\protected@writeaux\relax{%
592         \string\FN@pp@footnote@aux
593         {\the\c@@fnserial}{\noexpand\thepage}}%
594     }%
595   }%
596   \FN@pp@towritetrue
597 \fi
598 \ifnum\c@pp@next@reset>\c@@fnserial
599 \else
600   \global\expandafter\csname c@\@mpfn\endcsname\@ne
601   \protected@xdef\@thefnmark{\thempfn}%

```

Now look at the next element in the chain:

```

602   \expandafter\let\expandafter\@tempa
603   \csname FN@pp@next-\number\c@pp@next@reset\endcsname

```

If the chain is broken here, set the next reset point to something (one hopes) infeasibly large... a weak point?

```

604   \ifx\@tempa\relax
605     \global\c@pp@next@reset\@MM
606   \else
607     \global\c@pp@next@reset\@tempa
608   \fi
609 \fi
610 \fi
611 }

```

End of code loaded when perpage option is given

```
612 \fi
```

Finally, if we're not doing paragraph footnotes, we redefine \@makefntext to take account of the value of \footnotemargin, to impose \footnotelayout, and to make the footnote body text hang, if appropriate.

```

613 \ifFN@para
614 \else

```

hanging footnote version:

```

615   \long\def\@makefntext#1{%
616     \ifFN@hangfoot
617       \bgroup

```

get the marker so we can measure it:

```
618     \setbox\@tempboxa\hbox{%
619       \ifdim\footnotemargin>0pt
620         \hb@xt@\footnotemargin{\@makefnmark\hss}%
621       \else
622         \@makefnmark
623       \fi
624     }%
```

use the width of the box to set up hanging (potentially for more than one paragraph); note that the hanging `\parskip` and `\parindent` are set *after* we've executed `\leavevmode(!)`

```
625     \leftmargin\wd\@tempboxa
626     \rightmargin\z@
627     \linewidth \columnwidth
628     \advance \linewidth -\leftmargin
629     \parshape \@ne \leftmargin \linewidth
630     \footnotesize
```

stop the `\parshape` being overwritten:

```
631     \setpar{\@par}}%
```

and finally put the marker in its chosen place:

```
632     \leavevmode
633     \llap{\box\@tempboxa}%
634     \parskip\hangfootparskip\relax
635     \parindent\hangfootparindent\relax
636     \else
```

ordinary (non-hanging) footnote version:

```
637     \parindent1em
638     \noindent
639     \ifdim\footnotemargin>\z@
640       \hb@xt@ \footnotemargin{\hss\@makefnmark}%
641     \else
642       \ifdim\footnotemargin=\z@
643         \llap{\@makefnmark}%
644       \else
645         \llap{\hb@xt@ -\footnotemargin{\@makefnmark\hss}}%
646       \fi
647     \fi
648     \fi
649     \footnotelayout#1%
```

if we're hanging, close the hang group

```
650     \ifFN@hangfoot
651       \par\egroup
652     \fi
653   }
654 \fi
```

## 6 Remaining requirements

We have to insert the code that executes the `stable` and `multiple` options. Since `stable` may suppress the setting of a footnote altogether, we put the `multiple`

option first, as otherwise we might get isolated superscripted commas that separate footnotes that have otherwise been suppressed.

## 6.1 The code that executes the multiple option

`\multiplefootnotemarker` This (revised) code derives from a suggestion by Alexander Rozhenko (the author of the *manyfoot* package): the intention is that *footmisc* and *many-foot* should be able to ‘interwork’, in the sense that each would recognise the other’s footnote marks and behave appropriately. The trick is that both `\FN@mf@prepare` and `\FN@mf@check` insert a marker (a cancelling pair of kerns (of opposite signs), which is detected in following `\footnote` or `\footnotemark` commands. Note we have to take special precautions to ensure that the kerns are the last things added to the horizontal list by the commands.

```

655 \ifFN@multiplefootnote
656   \providecommand*\multiplefootnotemarker}{3sp}
657   \providecommand*\multfootsep}{,}
658   \CheckCommand*\@footnotemark{%
659     \leavevmode
660     \ifhmode\edef\@x@sf{\the\spacefactor}\nobreak\fi
661     \@makefnmark
662     \ifhmode\spacefactor\@x@sf\fi
663     \relax
664   }
665   \renewcommand*\@footnotemark{%
666     \leavevmode
667     \ifhmode
668       \edef\@x@sf{\the\spacefactor}%
669       \FN@mf@check
670       \nobreak
671       \fi
672     \@makefnmark

```

if we’re also doing option `perpage`, write its stuff to the `.aux` file for it, so the *wotsit* node doesn’t interfere with our `\kern` detection.

```

673   \ifFN@pp@twrite
674     \FN@pp@writetemp
675     \FN@pp@towritefalse
676   \fi
677   \FN@mf@prepare
678   \ifhmode\spacefactor\@x@sf\fi
679   \relax
680 }
681 \def\FN@mf@prepare{%
682   \kern-\multiplefootnotemarker
683   \kern\multiplefootnotemarker\relax
684 }
685 \def\FN@mf@check{%
686   \ifdim\lastkern=\multiplefootnotemarker\relax
687     \edef\@x@sf{\the\spacefactor}%
688     \unkern
689     \textsuperscript{\multfootsep}%
690     \spacefactor\@x@sf\relax

```

```

691   \fi
692 }
      If we're not doing multiple, just create an empty \FN@mf@prepare
693 \else
694   \let\FN@mf@prepare\relax
695 \fi

```

## 6.2 The code that executes the stable option

```

\FN@stablefootnote \FN@sf@@footnote The basic idea is to use the 'original' code of \footnote (which this package
may have hacked around something chronic) only if we're in typesetting mode
(as determined by the state of the \protect command. Otherwise, the command
becomes an elaborate multistage 'gobble'.
696 \ifFN@stablefootnote
697 \let\FN@sf@@footnote\footnote
698 \def\footnote{\ifx\protect\@typeset@protect
699   \expandafter\FN@sf@@footnote
700   \else
701     \expandafter\FN@sf@gobble@opt
702   \fi
703 }

\FN@sf@gobble@opt \FN@sf@gobble@twobracket Define \FN@sf@gobble@opt as a robust command that gobbles either an optional
and a mandatory argument, or just a mandatory one.
704 \edef\FN@sf@gobble@opt{\noexpand\protect
705   \expandafter\noexpand\csname FN@sf@gobble@opt \endcsname}
706 \expandafter\def\csname FN@sf@gobble@opt \endcsname{%
707   \@ifnextchar [%]
708     \FN@sf@gobble@twobracket
709     \@gobble
710 }
711 \def\FN@sf@gobble@twobracket[#1]#2{}

\FN@sf@@footnotemark \FN@sf@gobble@optonly \FN@sf@gobble@bracket Now the same for \footnotemark
\FN@sf@gobble@optonly 712 \let\FN@sf@@footnotemark\footnotemark
\FN@sf@gobble@bracket 713 \def\footnotemark{\ifx\protect\@typeset@protect
714   \expandafter\FN@sf@@footnotemark
715   \else
716     \expandafter\FN@sf@gobble@optonly
717   \fi
718 }
719 \edef\FN@sf@gobble@optonly{\noexpand\protect
720   \expandafter\noexpand\csname FN@sf@gobble@optonly \endcsname}
721 \expandafter\def\csname FN@sf@gobble@optonly \endcsname{%
722   \@ifnextchar [%]
723     \FN@sf@gobble@bracket
724     {}%
725 }
726 \def\FN@sf@gobble@bracket[#1]{}
727 \fi

```

```
\setfnsymbol
\FN@fnsymbol@lamport
```

## 7 Symbol option variants

Lamport’s choice of symbols for `\fnsymbol` wasn’t entirely “traditional”, so we (now) provide alternatives. The `\setfnsymbol` command offers a small number of choices, and the user may define more still, using the `\DefineFnsymbols` or `\DefineFnsymbolsTM` commands, defined below.

```
728 \newcommand\setfnsymbol[1]{%
729   \@bsphack
730   \@ifundefined{FN@fnsymbol@#1}%
731   {%
732     \PackageError{footmisc}{Symbol style "#1" not known}%
733     \@eha
734   }{%
735     \expandafter\let\expandafter\@fnsymbol\csname
736       FN@fnsymbol@#1\endcsname
737   }%
738   \@esphack
739 }
```

The default selection is Lamport’s original, as represented in current L<sup>A</sup>T<sub>E</sub>X — we preserve it in case we need to “get back” to it.

```
740 \let\FN@fnsymbol@lamport\@fnsymbol
741 \</package>
```

```
\if@tempswb
\@tempswbfalse
\@tempwbtrue
\DefineFnsymbols
\@DefineFnsymbols
\@DefineFnsymbols@
\FN@build@symboldef
```

We need another temp conditional

```
742 \newif\if@tempswb
```

The macro `\DefineFnsymbols` allows the user to define a set of footnote symbols, to be used with the `\setfnsymbol` command. Syntax:

```
\DefineFnsymbols[*]{<set name>}[<style>]{<symbol list>}
```

If the optional asterisk is present, the set defined will produce an error if the symbol number is too large; otherwise it will quietly change to numbering in place of symbol use (a warning is produced at the end of the document). The set name is the future argument of `\setfnsymbol`. The style (default `text`) gives the style the symbols are typeset (this is the *correct* method, but unfortunately not all symbols, even for Lamport’s original set for L<sup>A</sup>T<sub>E</sub>X `\fnsymbol` may be expressed this way in a sufficiently old L<sup>A</sup>T<sub>E</sub>X distribution). The symbol list is a set of objects to be used when the set is selected.

Example of use:

define a direct replacement for Lamport’s original `\fnsymbol` command —

```
\DefineFnsymbols*{lamport}[math]{*\dagger\ddagger\mathsection
\mathparagraph\|{**}{\dagger\dagger}{\ddagger\ddagger}}%
}
```

Note that doubled-up (and worse — see below) symbols need braces around them.

```
743 \newcommand{\DefineFnsymbols}{%
744   \ifstar{\@tempwbtrue\@DefineFnsymbols}%
745   {\@tempwbfalse\@DefineFnsymbols}%
746 }
747 \newcommand{\@DefineFnsymbols}[1]{%
748   \@ifnextchar[% ]
749   {\@DefineFnsymbols@{#1}}{\@DefineFnsymbols@{#1}[text]}%
```

```

750 }
751 \def\@DefineFNSymbols@#1[#2]#3{%
752   \expandafter\ifx\csname FN@fnsymbol@#1\endcsname\relax
753     \PackageInfo{footmisc}{Declaring symbol style #1}%
754   \else
755     \PackageWarning{footmisc}{Redeclaring symbol style #1}%
756   \fi
757   \toks@{ }%
758   \def\@tempb{\end}%
759   \FN@build@symboldef#3\end
760   \def\@tempc{math}%
761   \def\@tempd{#2}%
762   \expandafter\xdef\csname FN@fnsymbol@#1\endcsname##1{%
763     \ifx\@tempc\@tempd
764       \noexpand\ensuremath
765     \else
766       \noexpand\nfss@text
767     \fi
768     {%
769       \noexpand\ifcase##1%
770       \the\toks@
771       \noexpand\else
772       \if@tempswb
773         \noexpand\ctrerr
774       \else
775         \noexpand\@arabic##1\noexpand\FN@orange##1%
776       \fi
777       \noexpand\fi
778     }%
779   }%
780 }
781 \def\FN@build@symboldef#1{%
782   \def\@tempa{#1}%
783   \ifx\@tempa\@tempb
784   \else
785     \toks@\expandafter{\the\toks@\or#1}%
786     \expandafter\FN@build@symboldef
787   \fi
788 }

```

\DefineFNSymbolsTM Now do the same job for the “modern” way of having both text and maths variants  
\@DefineFNSymbolsTM of everything.

```

\FN@build@symboldefTM 789 \newcommand{\DefineFNSymbolsTM}{%
790   \ifstar{\@tempswbtrue\@DefineFNSymbolsTM}%
791   {\@tempswbfalse\@DefineFNSymbolsTM}}%
792 \newcommand{\@DefineFNSymbolsTM}[2]{%
793   \expandafter\ifx\csname FN@fnsymbol@#1\endcsname\relax
794     \PackageInfo{footmisc}{Declaring symbol style #1}%
795   \else
796     \PackageWarning{footmisc}{Redeclaring symbol style #1}%
797   \fi
798   \toks@{ }%
799   \def\@tempb{\end}%
800   \FN@build@symboldefTM#2\end\@null

```

```

801 \expandafter\xdef\csname FN@fnsymbol@#1\endcsname##1{%
802   \noexpand\ifcase##1%
803     \the\toks@
804   \noexpand\else
805     \if@tempswb
806       \noexpand\@ctrerr
807     \else
808       \noexpand\@arabic##1\noexpand\FN@orange##1%
809     \fi
810   \noexpand\fi
811 }%
812 }%

```

Note that this version has two variants of every definition, so needs two stopper codes above.

```

813 \def\FN@build@symboldefTM#1#2{%
814   \def\@tempa{#1}%
815   \ifx\@tempa\@tempb
816   \else
817     \toks@\expandafter{\the\toks@\or\TextOrMath{#1}{#2}}%
818     \expandafter\FN@build@symboldefTM
819   \fi
820 }

```

`\TextOrMath` This is a stripped down (e-TeX only) version of what appears in `fixltx2e`. If the command's already defined, we assume it's that version.

```

821 \@ifundefined{TextOrMath}{%
822   \@ifundefined{eTeXversion}{%
823     \PackageError{footmisc}{Can't define commands for footnote symbol}%
824     {Use e-LaTeX, or load package fixltx2e before
825     footmisc}%
826 }{%
827   \protected\expandafter\def\csname TextOrMath\space\endcsname{%
828     \ifmmode \expandafter\@secondoftwo
829     \else \expandafter\@firstoftwo \fi
830   }
831   \edef\TextOrMath#1#2{%
832     \expandafter\noexpand\csname TextOrMath\space\endcsname
833     {#1}{#2}%
834   }%
835 }%
836 }{}

```

`\FN@orange` `\@fnsymbol@orange` Macros to deal with footnote symbols going out of range (when they're allowed to—e.g., in the `symbol*` option).

```

\@diagnose@fnsymbol@orange 837 \def\FN@orange#1{%
838   \@bsphack
839   \PackageInfo{footmisc}{Footnote number \number#1 out of range}%
840   \protect\@fnsymbol@orange
841   \@esphack
842 }
843 \global\let\@diagnose@fnsymbol@orange\relax
844 \AtEndDocument{\@diagnose@fnsymbol@orange}
845 \def\@fnsymbol@orange{%

```

```

846 \gdef\@diagnose@fnsymbol@orange{%
847   \PackageWarningNoLine{footmisc}{Some footnote number(s)
848     were out of range
849     \MessageBreak
850     see log for details%
851   }%
852 }%
853 }

```

`\textbardbl` This is defined in recent L<sup>A</sup>T<sub>E</sub>X releases, but not in (for example) that distributed with the last release of t<sub>E</sub>X. Since it's needed in some symbol set definitions (including Lamport's) we define it here.

```

854 \@ifundefined{textbardbl}{%
855   \DeclareTextSymbol{textbardbl}{OMS}{107}%
856   \DeclareTextSymbolDefault{textbardbl}{TS1}}{}%

```

(This definition comes from the L<sup>A</sup>T<sub>E</sub>X sources.)

`\FN@fnsymbol@bringhurst` These macros provide replacement orderings (and symbol sets) for footnote symbols, plus a robust version of the original Lamport set, and an extended version of Lamport's original

```

\FN@fnsymbol@chicago
\FN@fnsymbol@wiley
\FN@fnsymbol@lamport-robust
\FN@fnsymbol@lamport
857 \DefineFNSymbolsTM*{bringhurst}{%
858   \textasteriskcentered *
859   \textdagger \dagger
860   \textdaggerdbl \ddagger
861   \textsection \mathsection
862   \textbardbl \||%
863   \textparagraph \mathparagraph
864 }%
865 \DefineFNSymbolsTM*{chicago}{%
866   \textasteriskcentered *
867   \textdagger \dagger
868   \textdaggerdbl \ddagger
869   \textsection \mathsection
870   \textbardbl \||%
871   \#\#%
872 }%
873 \DefineFNSymbolsTM*{wiley}{
874   \textasteriskcentered *
875   {\textasteriskcentered\textasteriskcentered}{**}%
876   \textdagger \dagger
877   \textdaggerdbl \ddagger
878   \textsection \mathsection
879   \textparagraph \mathparagraph
880   \textbardbl \||%
881 }%
882 \DefineFNSymbolsTM{lamport-robust}{
883   \textasteriskcentered *
884   \textdagger \dagger
885   \textdaggerdbl \ddagger
886   \textsection \mathsection
887   \textparagraph \mathparagraph
888   \textbardbl \||%
889   {\textasteriskcentered\textasteriskcentered}{**}%

```

```

890 {\textdagger\textdagger}{\dagger\dagger}%
891 {\textdaggerdbl\textdaggerdbl}{\ddagger\ddagger}%
892 }
893 \DefineFNsymbolsTM*{\lamport*}{%
894 \textasteriskcentered *
895 \textdagger \dagger
896 \textdaggerdbl \ddagger
897 \textsection \mathsection
898 \textparagraph \mathparagraph
899 \textbardbl \||%
900 {\textasteriskcentered\textasteriskcentered}{**}%
901 {\textdagger\textdagger}{\dagger\dagger}%
902 {\textdaggerdbl\textdaggerdbl}{\ddagger\ddagger}%
903 {\textsection\textsection}{\mathsection\mathsection}%
904 {\textparagraph\textparagraph}{\mathparagraph\mathparagraph}%
905 {\textasteriskcentered\textasteriskcentered\textasteriskcentered}{***}%
906 {\textdagger\textdagger\textdagger}{\dagger\dagger\dagger}%
907 {\textdaggerdbl\textdaggerdbl\textdaggerdbl}{\ddagger\ddagger\ddagger}%
908 {\textsection\textsection\textsection}%%
909 {\mathsection\mathsection\mathsection}%
910 {\textparagraph\textparagraph\textparagraph}%%
911 {\mathparagraph\mathparagraph\mathparagraph}%
912 }
913 \setfnsymbol{\lamport*}
914 \DefineFNsymbolsTM{\lamport*-robust}{%
915 \textasteriskcentered *
916 \textdagger \dagger
917 \textdaggerdbl \ddagger
918 \textsection \mathsection
919 \textparagraph \mathparagraph
920 \textbardbl \||%
921 {\textasteriskcentered\textasteriskcentered}{**}%
922 {\textdagger\textdagger}{\dagger\dagger}%
923 {\textdaggerdbl\textdaggerdbl}{\ddagger\ddagger}%
924 {\textsection\textsection}{\mathsection\mathsection}%
925 {\textparagraph\textparagraph}{\mathparagraph\mathparagraph}%
926 {\textasteriskcentered\textasteriskcentered\textasteriskcentered}{***}%
927 {\textdagger\textdagger\textdagger}{\dagger\dagger\dagger}%
928 {\textdaggerdbl\textdaggerdbl\textdaggerdbl}{\ddagger\ddagger\ddagger}%
929 {\textsection\textsection\textsection}%%
930 {\mathsection\mathsection\mathsection}%
931 {\textparagraph\textparagraph\textparagraph}%%
932 {\mathparagraph\mathparagraph\mathparagraph}%
933 }

```

## 8 Other miscellaneous commands

### 8.1 Footnote references

`\footref` Syntax: `\footref{<label-name>}`

One often wishes to refer to a footnote; in some circumstances, `\footnotemark` just isn't good enough (for example, inside a `minipage`, when `\footnotemark` creates a reference to footnotes outside the `minipage`).

`\footref` addresses this problem by making a label reference that actually looks like a `\footnotemark`. (The command is available in the `memoir` class, and we therefore `\providecommand` it rather than defining it “outright”.)

```
934 \providecommand*\footref}[1]{%
935   \begingroup
936   \unrestored@protected@xdef\@thefnmark{\ref{#1}}%
937   \endgroup
938   \@footnotemark
939 }
```

## 8.2 Minipage `\footnotemarks`

`\mpfootnotemark` Syntax: `\mpfootnotemark[(number)]`

Here we define `\mpfootnotemark`, which has the same syntax as `\footnotemark`, and which applies the semantics of `\footnotemark` to the minipage footnote series.

```
940 \newcommand\mpfootnotemark{%
941   \@ifnextchar[%
942     \@xmpfootnotemark
943     {%
944       \stepcounter\@mpfn
945       \protected@xdef\@thefnmark{\thempfn}%
946       \@footnotemark
947     }%
948 }
949 \def\@xmpfootnotemark[#1]{%
950   \begingroup
951   \csname c@\@mpfn\endcsname #1\relax
952   \unrestored@protected@xdef\@thefnmark{\thempfn}%
953   \endgroup
954   \@footnotemark
955 }

956 \endinput
957 </package>
```

## 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.

<b>Symbols</b>	<code>\bsphack</code> . . . . . 729, 838	<code>\@eha</code> . . . . . 733
<code>\#</code> . . . . . 871	<code>\colht</code> . . . . . 126,	<code>\@elt</code> . . . . . 119, 151
<code>\@advance@macro</code> . 7, <u>7</u> , 8	158, 190, 219, 286	<code>\@empty</code> 44, 121, 153,
<code>\@par</code> . . . . . 631	<code>\combinefloats</code> . . . . .	185, 214, 254, 278
<code>\@DefineFNsymbols</code> . <u>743</u>	. . . . . 122, 154,	<code>\esphack</code> . . . . . 738, 841
<code>\@DefineFNsymbols@</code> . <u>743</u>	186, 215, 255, 279	<code>\finalstrut</code> . . . . . 378
<code>\@DefineFNsymbolsTM</code> <u>789</u>	<code>\ctrerr</code> . . . . . 773, 806	<code>\firstoftwo</code> . . . . . 829
<code>\@addtoreset</code> . . . . . 480	<code>\currentlabel</code> . . . . .	<code>\fnsymbol</code> 23, 735, 740
<code>\@advance@macro</code> . . . . . <u>7</u>	. . . . . 320, 348, 372	<code>\fnsymbol@orange</code> . <u>837</u>
<code>\@arabic</code> . . . . . 775, 808	<code>\@diagnose@fnsymbol@orange\@footnotemark</code> 443,	
<code>\@auxout</code> . . . . . 5	. . . . . <u>837</u>	450, 464, 473,

655, 938, 946, 954	\@tempboxa . . . . .	178, 207, 233, 244, 321, 351, 375
\@footnotetext 308, 342, 360, 443, 455	. 387, 618, 625, 633	\color@endgroup . . . . .
\@freelist 120, 152, 184, 213, 253, 277	\@tempc . . . . . 760, 763	. . . . 116, 148, 181, 210, 237, 247, 336, 355, 380
\@ifclassloaded . . . 299	\@tempcnta 8–10, 488–490	\columnwidth 318, 331, 370, 404, 409, 627
\@ifl@t@r . 105, 135, 169	\@tempd . . . . . 761, 763	\CurrentOption . . . . .
\@ifpackageloaded . 297	\@tempdima . . . . .	. . . . 28, 30, 37, 39
\@ifstar . . . . . 744, 790	. 141, 175, 403–405	
\@kludgeins . . . 123, 155, 187, 216, 283	\@tempswattrue . 103, 564	
\@listdepth . . . . . 85	\@tempswbfalse . . . . .	
\@makecol . . . . 106, 136, 170, 200, 294	. . . . 742, 745, 791	
\@makefnmark . . 392, 620, 622, 640, 643, 645, 661, 672	\@tempswbtrue . . . . .	
\@makefntext 323, 326, 352, 376, 391, 615	. . . . 742, 744, 790	<b>D</b>
\@makespecialcolbox . . . . . 124, 156, 188, 217, 284	\@textbottom . . 129, 163, 193, 222, 289	\DeclareOption 21, 22, 27, 36, 45, 52, 57, 60, 63, 66, 70, 75, 79, 98, 100
\@maxdepth . . . 110, 132, 140, 166, 174, 196, 204, 225, 259, 270, 292	\@texttop . . . . 127, 159, 191, 220, 287	\DeclareTextSymbol . 855
\@midlist . 120, 121, 152, 153, 184, 185, 213, 214, 253, 254, 277, 278	\@thefnmark . . . 320, 349, 373, 442, 449, 463, 472, 601, 936, 945, 952	\DeclareTextSymbolDefault . . . . . 856
\@mpfn . . . . . 442, 449, 600, 944, 951	\@typeset@protect . . . . .	\DefineFNsymbols . . 743
\@mplistdepth . . . . 85	. . . . . 698, 713	\DefineFNsymbolsTM . . . . .
\@ne . . . . . 489, 531, 588, 600, 629	\@x@sf . . . . 660, 662, 668, 678, 687, 690	. 789, 857, 865, 873, 882, 893, 914
\@null . . . . . 800	\@xfootnote . . . 440, 447	\dimen@ 127, 129, 160, 162, 191, 193, 220, 222, 287, 289
\@outputbox 107, 109, 126–128, 137, 139, 158, 160, 161, 171, 173, 190–192, 201, 203, 219–221, 252, 258, 260, 267, 269, 286–288	\@xfootnotemark 460, 469	\dimexpr . . . . . 14, 330
\@parboxrestore . . . . .	\@xmpfootnotemark . . . . .	
. . . . 319, 371, 410	. . . . . 942, 949	<b>F</b>
\@plus . . . . . 77	<b>A</b>	\floatingpenalty . . . . .
\@secondoftwo . . . . 828	\AtBeginDocument . . . . .	. . . . . 317, 369
\@setpar . . . . . 631	. . . . 481, 492, 529	\fmtversion 105, 135, 169
\@tempa 503, 504, 544, 545, 561, 562, 602, 604, 607, 782, 783, 814, 815	\AtEndDocument 533, 844	\FN@baselinestretch . . . . .
\@tempb 758, 783, 799, 815	\AtEndOfPackage . . . 24	. . . . . 300, 302, 311, 345, 363
	<b>B</b>	\FN@bottomfalse . . . 59
	\baselineskip . . 73, 411	\FN@bottomtrue . . . . 61
	\baselinestretch . . . . .	\FN@build@symboldef 743
	. . . . 311, 345, 363	\FN@build@symboldefTM . . . . .
	\boxmaxdepth 110, 140, 174, 204, 259, 270	. . . . . 789
	<b>C</b>	\FN@etexfalse . . . . . 15
	\c@@fnserial . . . . .	\FN@etextrue . . . . . 17
	. 488, 588, 593, 598	\FN@fnsymbol@bringhurst . . . . .
	\c@footnote . . . . . 23	. . . . . 857
	\c@pp@next@reset . . . . .	\FN@fnsymbol@chicago . . . . .
	. 431, 490, 531, 598, 603, 605, 607	. . . . . 857
	\CheckCommand . . . . .	\FN@fnsymbol@lampoport . . . . .
	. 106, 136, 170, 200, 439, 458, 658	. . . . . 728, 857
	\clearpage . . . . . 584	\FN@fnsymbol@lampoport-robust . . . . .
	\color@begingroup . . . . .	. . . . . 857
	. . . . 113, 144,	\FN@fnsymbol@wiley . 857
		\FN@hangfootfalse . . 69
		\FN@hangfoottrue . . 71
		\FN@mf@check . . . . . 655

<code>\FN@mf@prepare</code> . . . . .	<code>\FN@sf@gobble@bracket</code>	<code>\fudgefactor</code> . . . . .
. 338, 357, 382, <u>655</u>	. . . . . <u>712</u>	333, <u>401</u>
<code>\FN@multiplefootnotefalse</code>	<code>\FN@sf@gobble@opt</code> .	<b>H</b>
. . . . . 99	. . . . . 701, <u>704</u>	<code>\hangfootparindent</code> .
<code>\FN@multiplefootnotetrue</code>	<code>\FN@sf@gobble@optonly</code>	. . . . . <u>73</u> , 635
. . . . . 100	. . . . . <u>712</u>	<code>\hangfootparskip</code> <u>73</u> , 634
<code>\FN@orange</code> 775, 808, <u>837</u>	<code>\FN@sf@gobble@twobracket</code>	<code>\hb@xt@</code> . . . . . 620, 640, 645
<code>\FN@parafalse</code> . . . . . 26	. . . . . <u>704</u>	<code>\hrule</code> . . . . . 83
<code>\FN@paratrue</code> . . . . . 32	<code>\FN@sidefnfalse</code> . . . . . 35	<code>\hsize</code> . . . . . 318, 370, 409
<code>\FN@perpagefalse</code> . . . . . 51	<code>\FN@sidefntrue</code> . . . . . 41	<b>I</b>
<code>\FN@perpagetrue</code> . . . . . 53	<code>\FN@singlespace</code> . . . . . <u>296</u>	<code>\if@filesw</code> . . . . . 589
<code>\FN@pp@clearpage</code> . . . . . <u>584</u>	<code>\FN@stablefootnotefalse</code>	<code>\if@minipage</code> . . . . . 587
<code>\FN@pp@cpage</code> . . . . .	. . . . . 97	<code>\if@tempswa</code> . . . . . 102, 104
. 483, 504, 521,	<code>\FN@stablefootnotetrue</code>	<code>\if@tempswb</code> <u>742</u> , 772, 805
545, 570, 574, 582	. . . . . 98	<code>\ifFN@baselinestretch</code>
<code>\FN@pp@debugfalse</code> . . . . . 56	<code>\FN@tempboxa</code> . . . . .	. . . . . <u>296</u>
<code>\FN@pp@debugtrue</code> . . . . . 57	. 322, 328, 330,	<code>\ifFN@bottom</code> <u>59</u> , 102, 251
<code>\FN@pp@footnote</code> . . . . .	333, 335, <u>387</u> ,	<code>\ifFN@etex</code> <u>13</u> , 329, 401
. . . . . 450, 473, <u>586</u>	408, 414, 416,	<code>\ifFN@hangfoot</code> . . . . .
<code>\FN@pp@footnote@aux</code>	420, 423, 425, 426	. . . . . 69, 616, 650
. 497, 530, 533, 592	<code>\FN@tempboxb</code> . . . . .	<code>\ifFN@multiplefootnote</code>
<code>\FN@pp@footnote@endaux</code>	. . . . . <u>387</u> , 418–420	. . . . . <u>99</u> , 655
. . . . . 533, <u>538</u>	<code>\FN@tempboxc</code> . . . . .	<code>\ifFN@para</code> <u>26</u> , 36, 103,
<code>\FN@pp@footnotehintfalse</code>	. . . . . 235, 236, 389	229, 307, 386, 613
. . . . . 501, 535, 542	<code>\FN@temptoken</code> . . . . . 3,	<code>\ifFN@perpage</code> . . . . . <u>50</u> , 438
<code>\FN@pp@footnotehinttrue</code>	230, 241, 263, 273	<code>\ifFN@pp@debug</code> . . . . .
. . . . . 484, 487, 493	<code>\fnsymbol</code> . . . . . 21	. . . . . <u>55</u> , 512, 553, 566
<code>\FN@pp@initial@stab</code>	<code>\footglue</code> . . . . . <u>325</u> , <u>390</u>	<code>\ifFN@pp@footnotehint</code>
. . . . . 480, 481, 518	<code>\footins</code> . . . . . 77, 106,	. . . . . 430, 499, 540
<code>\FN@pp@initialstab</code> . . . . . <u>480</u>	112, 115, 136,	<code>\ifFN@pp@lastseq</code> . . . . .
<code>\FN@pp@lastfoot</code> <u>496</u> ,	143, 147, 170,	. . . . . <u>434</u> , 568
498, 509, 514,	177, 180, 200,	<code>\ifFN@pp@towrite</code> . . . . .
534, 539, 550, 555	206, 209, 232,	. . . . . <u>432</u> , 451, 474, 673
<code>\FN@pp@lastseqfalse</code>	243, 246, 256,	<code>\ifFN@setspace</code> . . . . .
. . . . . 436, 576	266, 309, 361, 407	. . . . . 296, 310, 344, 362
<code>\FN@pp@lastseqtrue</code> . . . . . 571	<code>\footnote</code> . . . . .	<code>\ifFN@sidefn</code> 27, <u>35</u> , 341
<code>\FN@pp@prev@foot</code> . . . . .	. 439, 446, 697, 698	<code>\ifFN@stablefootnote</code>
. . . . . 523, 524,	<code>\footnotebaselineskip</code>	. . . . . <u>97</u> , <u>696</u>
526, <u>528</u> , 536,	. 331, <u>395</u> , 403, 411	<code>\ifmmode</code> . . . . . 828
562, 567, 575, 581	<code>\footnotehint</code> . . . . . <u>484</u> , 585	<code>\ignorespaces</code> . . . . .
<code>\FN@pp@resetfn</code> . . . . .	<code>\footnotelayout</code> <u>44</u> , 649	. . . . . 323, 353, 378
. . . . . 500, 506, <u>520</u>	<code>\footnotemargin</code> <u>19</u> ,	<code>\insert</code> . . . . . 309, 361
<code>\FN@pp@resetfn@end</code> . . . . .	64, 67, 619, 620,	<code>\insertpenalties</code> . . . . . 93
. . . . . 541, 547, <u>560</u>	639, 640, 642, 645	<code>\interfootnotelinepenalty</code>
<code>\FN@pp@towritefalse</code>	<code>\footnotemark</code> . . . . .	. . . . . 314, 366
. 433, 453, 476, 675	. 458, 467, 712, 713	<code>\interlinepenalty</code> . . . . .
<code>\FN@pp@towritetrue</code> . . . . . 596	<code>\footnoterule</code> 76, 81,	. . . . . 314, 366
<code>\FN@pp@writetemp</code> . . . . .	82, 84, 114, 146,	<b>L</b>
. 452, 475, 590, 674	179, 208, 234, 245	<code>\l@advance@macro</code> . . . . . <u>7</u>
<code>\FN@setspacefalse</code> . . . . . 305	<code>\footnotesep</code> . . . . .	<code>\lastbox</code> . . . . . 418, 423
<code>\FN@setspacetrue</code> . . . . . 298	. 315, 367, 377, 413	<code>\lastkern</code> . . . . . 686
<code>\FN@sf@footnote</code> . . . . . <u>696</u>	<code>\footnotesize</code> . . . . . 313,	<code>\leavevmode</code> . . . . .
<code>\FN@sf@footnotemark</code>	347, 365, 397, 630	. . . . . 391, 632, 659, 666
. . . . . <u>712</u>	<code>\footref</code> . . . . . <u>934</u>	

<code>\leftmargin</code>	625, 628, 629	<code>\parshape</code>	629	<code>\textdagger</code>	859,
<code>\linepenalty</code>	47, 48	<code>\parskip</code>	634		867, 876, 884,
<code>\linewidth</code>	627–629	<code>\ProcessOptions</code>	101		890, 895, 901,
<code>\llap</code>	633, 643, 645	<code>\protected</code>	827		906, 916, 922, 927
<b>M</b>					
<code>\m@msinglespace</code>	300	<code>\protected@edef</code>	320, 348, 372	<code>\textdaggerdbl</code>	860,
<code>\makefootnoteparagraph</code>	235, 407	<code>\protected@write</code>	5		868, 877, 885,
<code>\makehboxoffboxes</code>	407, 416	<code>\protected@writeaux</code>	4, 487, 492, 529, 591		891, 896, 902,
<code>\marginpar</code>	343	<code>\protected@xdef</code>	442, 449,		907, 917, 923, 928
<code>\mathparagraph</code>	863, 879,		463, 472, 601, 945	<code>\TextOrMath</code>	817, 821
	887, 898, 904,	<b>R</b>			
	911, 919, 925, 932	<code>\RaggedRight</code>	48	<code>\textparagraph</code>	863, 879,
<code>\mathsection</code>	861, 869, 878,	<code>\raggedright</code>	47		887, 898, 904,
	886, 897, 903,	<code>\ref</code>	936		910, 919, 925, 931
	909, 918, 924, 930	<code>\removehboxes</code>	408, 416	<code>\textsection</code>	861, 869, 878,
<code>\maxdepth</code>	132,	<code>\repeat</code>	421		886, 897, 903,
	166, 196, 225, 292	<code>\reset@font</code>	313, 347, 365		908, 918, 924, 929
<code>\mpfootnotemark</code>	940	<code>\rightmargin</code>	626	<code>\textsuperscript</code>	689
<code>\mpfootnoterule</code>	81, 86	<code>\rule</code>	377, 413	<code>\thefootnote</code>	21, 23, 463, 472
<code>\multfootsep</code>	655	<b>S</b>			
<code>\multiplefootnotemarker</code>	655	<code>\setfnsymbol</code>	24, 728, 913	<code>\thempfn</code>	442,
<code>\multiply</code>	403, 404	<code>\setspace@singlespace</code>	302		449, 601, 945, 952
<b>N</b>					
<code>\NeedsTeXFormat</code>	2	<code>\spacefactor</code>	660, 662,	<code>\thepage</code>	593
<code>\nfss@text</code>	766		668, 678, 687, 690	<code>\toks@</code>	231,
<code>\nobreak</code>	392, 660, 670	<code>\split@prev</code>	79, 88, 93		242, 252, 264,
<code>\noindent</code>	412, 638	<code>\splitfootnoterule</code>	81, 91		266, 274, 282,
<code>\normalbaselineskip</code>	399		316, 368		294, 757, 770,
<code>\normalcolor</code>	114, 145,	<code>\splitmaxdepth</code>	315, 367		785, 798, 803, 817
	179, 208, 234, 245	<code>\splittopskip</code>	442,	<b>U</b>	
<code>\number</code>	603, 839		449, 462, 471, 944	<code>\undefined</code>	14
<b>P</b>					
<code>\PackageError</code>	28, 37, 732, 823	<code>\string</code>	592	<code>\unhbox</code>	408, 414, 420, 426
<code>\PackageInfo</code>	753, 794, 839	<code>\strip@pt</code>	405	<code>\unkern</code>	688
<code>\PackageWarning</code>	755, 796	<code>\strut</code>	323	<code>\unrestored@protected@xdef</code>	936, 952
<code>\PackageWarningNoLine</code>	847	<code>\strutbox</code>	316, 368, 378	<code>\unvbox</code>	111, 115,
<code>\pagefootnoterule</code>	81, 89	<b>T</b>			
<code>\parindent</code>	635, 637	<code>\textasteriskcentered</code>	858, 866,		128, 142, 147,
			874, 875, 883,		161, 176, 180,
			889, 894, 900,		192, 205, 209,
			905, 915, 921, 926		221, 236, 246,
		<code>\textbardbl</code>	854, 862, 870,		260, 271, 288, 407
			880, 888, 899, 920	<b>V</b>	
				<code>\vfill</code>	261
				<b>Z</b>	
				<code>\z@</code>	88, 377,
					413, 626, 639, 642