

# FiXme – a L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> package for inserting fixme notes in your documents\*

Didier Verna  
mailto:didier@lrde.epita.fr  
<http://www.lrde.epita.fr/~didier/>

November 28, 2007

## Abstract

In the process of writing a long document, it is a common practice to leave some parts unwritten or uncomplete, and come back to them later. In such cases, you probably want to stick clues about which parts need to be “fixed”, where they are located, and what needs to be done. This is what I call “fixme notes”. The purpose of this package is to provide you with convenient ways to insert FiXme notes in your documents.

The FiXme package is Copyright © 1998, 1999, 2000, 2001, 2002, 2004, 2005, 2006, 2007 Didier Verna, and distributed under the terms of the LPPL license.

## 1 Description

With FiXme, you can insert different kinds of notes in your documents, ranging from simple not-so-important notices to critical stuff that must absolutely be fixed in the final version.

FiXme gives you full control on the layout of these notes: they can be displayed inline (directly in the text), as marginal paragraphs (the default), as footnotes and even as index entries. All these possibilities can be mixed together. Additionally, you can summarize all FiXme notes in a “list of fixme’s”.

FiXme notes are also recorded in the log file, and (depending on their importance level) some of them are displayed on the terminal during compilation. A final summary is also created at the end of the compilation process.

All these features are actually available when you’re working in **draft** mode. In **final** mode, the behavior is slightly different: any remaining critical note generates an error (the compilation aborts), while non critical ones are just removed from the document’s body (they’re still recorded in the log file though).

## 2 Using FiXme

First of all, please note that the **ifthen** and **verbatim** packages are required. You don’t have to load them explicitly though. As long as L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> can locate them, they will be used automatically.

---

\*This document describes FiXme 3.3, release date 2007/11/28.

To use FiXme, simply say `\usepackage[options]{fixme}` in the preamble of your document.

## 2.1 Inserting notes

### 2.1.1 Macros

`\fixme` The main command for inserting a FiXme note in your document is the `\fixme` macro. It takes the note to insert as its mandatory argument. Notes inserted via this command are considered fatal to your document's final processing (see section 2.2).

`\fxnote` As of version 2.0, FiXme provides three new macros that insert meta-comments about the document, rather than real fixmes. These comments have three different importance levels: note, warning and (non critical) error. The corresponding macros obey the same syntax as `\fixme`. However, none of the notes inserted via these macros are fatal.

`\fxwarning`

`\fxerror`

### 2.1.2 Environments

`anfxnote` As of version 3.0, FiXme provides environments for inserting longer notes. These environments take one optional argument (should be short, perhaps a summary of the note) that will be used in the list of fixme's and in the index if required.

`anfxwarning`

`anfxerror`

`afixme` The FiXme environments behave exactly the same way as their macro counterpart, except for the layout (see section 2.3): as they are meant for longer notes, the layout is always `inline`. The `index` and `marginclue` layouts are still honored however. By default, FiXme environments are typeset in a `quotation` one.

## 2.2 Controlling the behavior of FiXme

`final` The global behavior of the package is controlled via the two standard options `final` and `draft`. These options are usually given to `\documentclass{}` which in turn passes them to all packages.

`draft`

In `draft` mode, the notes are recorded in the log file, and appear in the text as specified by the layout settings (see section 2.3). Additionally, warnings, errors and fatal errors are also displayed on the terminal.

In `final` mode, non fatal notes (those generated by the `\fx*` commands or by the `anfx*` environments) are still logged, but they're not typeset. On the other hand, any remaining fatal note (generated by the `\fixme` command or the `afixme` environment) will throw a  $\LaTeX$  error and thus interrupt or abort compilation with an informative message. This will help you track down forgotten important caveats in your document. Let me say it again: `final` documents can only have notes, warnings, and (non critical) errors left. Well, if, for some reason, you really want to compile in `final` mode with critical FiXme notes left behind, you always have the possibility to pass the `draft` option to FiXme directly...

The `final` mode has been chosen as the default because  $\LaTeX 2_{\epsilon}$  itself behaves this way.

## 2.3 Controlling the notes layout

FiXme notes can appear in several forms (that can be combined) in your document.

### 2.3.1 Global layout

`inline` The layout forms currently supported are: `inline` (directly in the text), marginal notes, marginal clues (see below), footnotes, and index entries. To activate a particular layout globally (that is, for the whole document), use the corresponding package option. By default, only the `margin` layout is active.

`margin` Each layout option has a counterpart that deactivates it. The counterpart option has the same name, prefixed with `no`. For instance, if you don't want marginal notes, use the `nomargin` option.

`marginclue` Finally, note that FiXme environments behave in a special way: they are always typeset `inline`, regardless of your layout settings (they respect your wish for index and marginal clues though).

`footnote`

`index`

### 2.3.2 Marginal Clues

Sometimes, marginal notes are too narrow for what you want to put in them, so you would switch to an `inline` or a `footnote` layout. But then, it is more difficult to keep track of the FiXme note's location on the page.

As of version 3.2, FiXme provides a special layout called "marginal clue" to help locating the notes: a marginal clue does not display the note's contents, but only an indication that there is a note at that place. So you have to use another layout form (typically `inline` or `footnote`) in order to get the actual contents.

Obviously, `margin` and `marginclue` are mutually exclusive. If you try to activate both, only the most recently activated form will be enabled (and you'll get a notice in the compilation log).

### 2.3.3 Local Layout

`\fixme[]` As of version 3.0, FiXme provides a way to change the selected layout(s) on a per-note basis: each note insertion command takes an optional first argument that overrides the global layout. This argument consists of one or more layout options (`inline`, `margin`, `marginclue`, `footnote` and `index`) separated by commas. Remember that local layouts *override* the global ones; they don't add to it.

`\fxnote[]`

`\fxwarning[]`

`\fxerror[]`

On what occasion would one want to modify the layout for a particular note? Here is a typical situation: suppose you have a document in which FiXme notes are typeset as margin paragraphs (this is the default). You would not be able to put a note in a figure, because floats can't be nested in L<sup>A</sup>T<sub>E</sub>X (margin paragraphs are floats). In such a case, you would rather `inline` the note, which can be done with something like `\fixme[inline]{blah}`.

### 2.3.4 Class compatibility

By default, FiXme is set to use marginal notes. You should know that this is known to cause problems with at least the ACM SIG classes: `acm_proc_article-sp` and `sig-alternate`. These classes forbid the use of `\marginpar`, so if you use them, you should choose another layout for FiXme, and also avoid using marginal clues. . .

## 2.4 Controlling the notes logging

`silent` As well as being displayed in the document itself, all FiXme notes are "logged"

`nosilent`

in different ways: by default, simple notes are recorded in the log file while the others are also displayed on the terminal output.

You have the ability to suppress all kind of logging by using the `silent` option. By default, the behavior is that of `nosilent`.

## 2.5 List of FiXme's

`\listoffixmes` FiXme remembers where you put FiXme notes in a toc-like file whose extension is `lox`. The `\listoffixmes` macro generates the list of all FiXme notes in a manner similar to that of the “list of figures” for instance. A standard layout is automatically used for the ‘article’, ‘report’, ‘book’ classes and their koma-script replacements. If another class is used, the ‘article’ layout is selected. Also, note that if no FiXme note remain in the document, this macro doesn’t generate an empty list, but rather stays silent. It also stays silent in `final` mode, regardless of the presence of remaining notes.

## 3 Customizing FiXme

### 3.1 Customizing the notes layout

#### 3.1.1 Macros

The `inline`, `margin` and `footnote` layouts have two parts: a “prefix” which depends on the note level, and the note itself. The prefix is one of “FiXme note:”, “FiXme warning:”, “FiXme error:” or simply “FiXme:”, and appears in bold. The note itself appears emphasized. These layouts are implemented thanks to the corresponding `\FX*` macros that you can redefine if you wish. Each such macro takes two mandatory arguments: the prefix and the note, in that order.

`\FXInline`  
`\FXMargin`  
`\FXFootnote`  
`\FXMarginClue` The special `marginclue` layout only outputs the prefix mentioned above. Its layout is controlled by the macro `\FXMarginClue` which takes the prefix as its only mandatory argument. The `\FXIndex` macro is used for the `index` layout. All FiXme index entries appear under the “FiXme” key in the symbols section. There are 4 subcategories under this key, as many as there are note levels. By default, only the first 3 of them are used though (fatal errors do not appear under a subkey, but directly under the FiXme key). The notes are numbered in the index.

`\FXIndex`

#### 3.1.2 Environments

`\FXEnvBegin`  
`\FXEnvEnd` The optional argument of FiXme environments is always typeset according to the `inline` layout described above. By default, FiXme uses a `quotation` for displaying the environments’ contents. If you want to change that, you can redefine the macros `\FXEnvBegin` and `\FXEnvEnd` that open and close the environments.

### 3.2 Customizing the notes logging

`\FXNote`  
`\FXWarning`  
`\FXError`  
`\FXFatal` If you want a finer control on logging, you can redefine the commands used to implement it. These commands (on the left) are the ones used by `\fxnote`, `\fxwarning`, `\fxerror` and `\fixme` respectively. They take the note itself as mandatory argument.

### 3.3 Fancy fruit salad layout

`user` If, for some totally unjustified reason, you are not happy with the available layouts,  
`\FXUser` you have the ability to define your own: pass the `user` option to the package (it also has its `nouser` counterpart), and define an `\FXUser` macro in the following manner:

```
\newcommand{\FXUser}[2]{\fancy fruit salad layout job}
```

The arguments are the prefix, and the note itself, in that order.

Note that the `user` option can also be used in the optional argument of the notes insertion commands.

### 3.4 Internationalization

`english` FiXme currently supports English, French, Spanish, Italian, German, Danish and  
`french` Croatian. You can select the language you want to use by passing the corre-  
`francais` sponding option (these options are usually given directly to `\documentclass{}`  
`spanish` which in turn passes them to all packages). The `french` and `francais` options  
`italian` are synonyms. The `german` and `ngerman` options are currently equivalent.

`german` If you want a finer grain on the language-dependent parts of FiXme, the fol-  
`ngerman` lowing macros are provided and can be redefined.

`danish` The `\fixme*prefix` macros define the prefix for the four different note levels.  
`croatian` They make intensive use of the macro `\fixmelogo` ;-)

`\fixmenoteprefix` The macro `\fixmeindexname` defines the main FiXme index key. The other  
`\fixmewarningprefix` ones define the different index subkeys for each note level. Please note that an  
`\fixmeerrorprefix` empty name for a subkey means that you don't actually want a subcategory (that's  
`\fixmefatalprefix` the case by default for fatal errors). The corresponding notes will then appear  
`\fixmelogo` directly under the main FiXme key. For that reason, a non empty subkey must  
`\fixmeindexname` end with an exclamation mark.

`\fixmenoteindexname` `\listfixmename` defines the title for the “list of fixmes” section.

`\fixmewarningindexname`

`\fixmeerrorindexname`

`\fixmefatalindexname`

`\listfixmename`

## 4 AUC-TEX support

AUC-TEX is a powerful major mode for editing TEX documents in Emacs or XEmacs. In particular, it provides automatic completion of macro names once they are known. FiXme supports AUC-TEX by providing a style file named `fixme.el` which contains AUC-TEX definitions for the relevant macros. This file should be installed in a place where AUC-TEX can find it (usually in a subdirectory of your L<sup>A</sup>T<sub>E</sub>X styles directory). Please refer to the AUC-TEX documentation for more information on this.

## 5 Changes

v3.3 Document incompatibility between marginal layout and the ACM SIG classes, reported by Jochen Wuttke.

Honor `twoside` option in marginal layout, suggested by Jens Remus.

Support for KOMA-Script classes version 2006/07/30 v2.95b, suggested by Jens Remus.

Documentation improvements suggested by Brian van den Broek.

Fix incompatibility with `amsart` reported by Lars Madsen: `\starttoc`

takes two arguments.

Fix bug reported by Stefan Mann: a typo in the `\fixme@footnotetrue` macro name.

- v3.2 Added the `marginclue` layout option which only signals a `fixme` in the margin, without the actual contents.  
Support for Croatian thanks to Marcel Maretic <marcel@fsb.hr>.  
Fix incompatibility with `amsbook` reported by Claude Lacoursière: `\@starttoc` takes two arguments.  
Fix incompatibility with Beamer reported by Akim Demaille: protect contents of `lox` file.
- v3.1 Fix bug reported by Arnold Beckmann: the environments were visible in final mode.
- v3.0 Added environments corresponding to the note insertion commands.  
Added an optional first argument to the note insertion commands to change the layout locally.  
Fix bug reported by Akim Demaille: marginal notes could mess up the document's layout by flushing it right.
- v2.2 New option `silent` to suppress notes logging.  
Support for Danish thanks to Kim Rud Bille <krbi01@control.auc.dk>.
- v2.1 Use `\nobreakspace` instead of the tilda character. This avoids conflicts with Babel in Spanish environments.  
Fix bug reported by Knut Lickert: index entries were unconditionally built.
- v2.0 New feature: note levels.  
New feature: `FiXme` note counters and usage summary.  
Suggestions from Kasper B. Graversen <kb@dkik.dk>.  
Support for Spanish thanks to Agustín Martín <agusmba@terra.es>
- v1.5 New appearance option: `inline`.
- v1.4 Support for the koma-script classes.  
Fix bug reported by Ulf Jaenicke-Roessler: the `\listoffixmes` command didn't work when called before the first `FiXme` note.
- v1.3 Support for Italian thanks to Riccardo Murri <murri@phc.unipi.it>.
- v1.2 Support for German thanks to Harald Harders <h.harders@tu-bs.de>.

## 6 The Code

`\fixmelo` Here's the preamble:

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{fixme}[2007/11/28 v3.3]
3 Insert fixme notes in your documents]
4
5 \RequirePackage{ifthen}
6 \RequirePackage{verbatim}
```

```

7
8 \newcommand*\fixmefixme{\textsf{FixMe}}
9

```

## 6.1 List macros

`\listoffixmes` `\listoffixmes` is empty by default. The `final` or `draft` options will provide the proper definition. Lists are output in a document class dependant fashion. Classes currently recognized are article, report, book and their koma-script replacements.

In order to prevent the List of Fixme's heading from being generated when there are no fixmes, a test on the existence of the `lox` file is performed. There's a slight bug left however: after removing the last fixme, one ends up with an empty `lox` file, so the heading still appear. Previously, this was done by checking if some fixmes were given, but that was buggy: the list of fixme's could not appear before the first fixme note... I should try to detect whether the file is empty.

```

10 \let\l@fixme\l@figure
11 \newcommand\listoffixmes{}
12 \def\listoffixmes@final{}
13 \def\listoffixmes@draft{%
14   \IfFileExists{\jobname .lox}{\@listoffixmes@pretoc}{}%
15   \@starttoc{lox}%
16   \IfFileExists{\jobname .lox}{\@listoffixmes@posttoc}{}%
17 }
18

```

The `amsbook` and `amsart` classes have the very ugly idea of redefining the `\@starttoc` command to take two arguments. Therefore, I need to provide a specific version of the `\listoffixmes` macro:

```

19 \def\listoffixmes@draft@ams{\@starttoc{lox}\listfixmename}
20

```

`\listfixmename` In order to redefine the list title, users have to call `\renewcommand`. I don't like this much: it would be cleaner to provide a real command that redefines the title internally. However, I'll stick with the current scheme because it is consistent with the rest of L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>'s lists.

```

21 \newcommand*\listfixmename{}
22

```

### 6.1.1 Standard classes

`\listoffixmes@pretoc@article` Let's start with the article version:  
`\listoffixmes@posttoc@article`

```

23 \def\@listoffixmes@pretoc@article{%
24   \section*{\listfixmename%
25     \@mkboth{\MakeUppercase\listfixmename}{\MakeUppercase\listfixmename}}%
26 }
27 \def\@listoffixmes@posttoc@article{}
28

```

`\listoffixmes@pretoc@report` Here's the report version:  
`\listoffixmes@posttoc@report`

```

29 \def\@listoffixmes@pretoc@report{%
30   \if@twocolumn
31     \@restonecoltrue\onecolumn
32   \else
33     \@restonecolfalse
34   \fi
35   \chapter*{\listfixmenename%
36     \mkboth{\MakeUppercase\listfixmenename}{\MakeUppercase\listfixmenename}}%
37   }
38 \def\@listoffixmes@posttoc@report{\if@restonecol\twocolumn\fi}
39

```

\listoffixmes@pretoc@book And here's the book version:  
\listoffixmes@posttoc@book

```

40 \def\@listoffixmes@pretoc@book{%
41   \if@twocolumn
42     \@restonecoltrue\onecolumn
43   \else
44     \@restonecolfalse
45   \fi
46   \chapter*{\listfixmenename%
47     \mkboth{\MakeUppercase\listfixmenename}{\MakeUppercase\listfixmenename}}%
48   }
49 \def\@listoffixmes@posttoc@book{\if@restonecol\twocolumn\fi}
50

```

### 6.1.2 koma-script classes

The code below (version 3.3) mimics KOMA-Script version 2006/07/30 v2.95b. Older versions (using chapter\*) are no longer supported because it is simpler that way, but if some people complain, I'll have to conditionalize on the KOMA-Script version, which would be a PITA.

```

51 \newcommand*\lox@heading{\float@listhead{\listfixmenename}}
52

```

\listoffixmes@pretoc@scrartcl Let's start with the scrartcl version:  
\listoffixmes@posttoc@scrartcl

```

53 \def\@listoffixmes@pretoc@scrartcl{%
54   \begingroup%
55     \lox@heading%
56     \setparsizes{0}{0}{\z@\@plus 1fil}\par@updaterelative
57   }
58 \def\@listoffixmes@posttoc@scrartcl{%
59   \endgroup
60   }
61

```

\listoffixmes@pretoc@scrreprt Here's the scrreprt version:  
\listoffixmes@posttoc@scrreprt

```

62 \def\@listoffixmes@pretoc@scrreprt{%
63   \begingroup%
64     \if@twocolumn
65       \@restonecoltrue\onecolumn
66     \else
67       \@restonecolfalse

```



```

68     \fi
69     \lox@heading%
70     \setparsizes{0}{0}{\z@\@plus 1fil}\par@updaterelative
71   }
72 \def\@listoffixmes@posttoc@scrreprt{%
73     \if@restonecol\twocolumn\fi
74   \endgroup
75 }
76

```

\listoffixmes@pretoc@scrbook And here's the scrbook version:

```

\listoffixmes@posttoc@scrbook
77 \def\@listoffixmes@pretoc@scrbook{%
78     \begingroup%
79     \if@twocolumn
80         \@restonecoltrue\onecolumn
81     \else
82         \@restonecolfalse
83     \fi
84     \lox@heading%
85     \setparsizes{0}{0}{\z@\@plus 1fil}\par@updaterelative
86 }
87 \def\@listoffixmes@posttoc@scrbook{%
88     \if@restonecol\twocolumn\fi
89   \endgroup
90 }
91

```

## 6.2 Layout macros

\FXInline The macros to implement different layouts for the fixme notes:

\FXMargin

\FXMarginClue

\FXFootnote

```

92 \newcommand\FXInline[2]{%
93     \textbf{#1}: \emph{#2}}
94 \newcommand\FXMargin[2]{%
95     \marginpar[\footnotesize\raggedleft\textbf{#1}: \emph{#2}]{%
96         \footnotesize\raggedright\textbf{#1}: \emph{#2}}}
97 \newcommand\FXMarginClue[1]{%
98     \marginpar[\footnotesize\raggedleft\textbf{#1}~!]{%
99         \footnotesize\raggedright\textbf{#1}~!}}
100 \newcommand\FXFootnote[2]{%
101     \footnote{\textbf{#1}: \emph{#2}}}
102

```

\fixmeindexname The macros for index generation:

\FXIndex

```

103 \newcommand*\fixmeindexname{}
104 \newcommand\FXIndex[1]{%
105     \index{***@\fixmeindexname: !#1}}
106

```

\fixmefatalprefix Apart from \fixmefatalprefix, the following macros are kept only to preserve backward compatibility. They are used only by critical notes and are made to call the new versions by default:

\FixMeInline

\FixMeMargin

\FixMeFootnote

\FixMeUser

\FixMeIndex

\FixMeInfo

\FixMeWarning

```

107 \newcommand*\fixmefatalprefix{}
108 \newcommand\FiXmeInline[1]{\FXInline{\fixmefatalprefix}{#1}}
109 \newcommand\FiXmeMargin[1]{\FXMargin{\fixmefatalprefix}{#1}}
110 \newcommand\FiXmeFootnote[1]{\FXFootnote{\fixmefatalprefix}{#1}}
111 \newcommand\FiXmeUser[1]{}
112 \let\FiXmeUser\empty
113
114 \newcommand\FiXmeIndex[1]{\FXIndex{#1}}
115 \newcommand\FiXmeInfo[1]{}
116 \newcommand\FiXmeWarning[1]{}
117

```

Booleans that control which layouts are selected:

```

118 \newif\iffixme@inline
119 \newif\iffixme@margin
120 \newif\iffixme@marginclue
121 \newif\iffixme@footnote
122 \newif\iffixme@user
123 \newif\iffixme@index
124

```

Setters for these booleans (some might required special treatment):

```

125 \let\fixme@setinline\fixme@inlinetrue
126 \def\fixme@setmargin{%
127   \iffixme@marginclue%
128     \PackageWarning{FiXme}{%
129       Marginal notes requested; turning marginal clues off}%
130   \fixme@margincluefalse%
131   \fi%
132   \fixme@margintrue}
133 \def\fixme@setmarginclue{%
134   \iffixme@margin%
135     \PackageWarning{FiXme}{%
136       Marginal clues requested; turning marginal notes off}%
137   \fixme@marginfalse%
138   \fi%
139   \fixme@margincluetrue}
140 \let\fixme@setfootnote\fixme@footnotetrue
141 \let\fixme@setuser\fixme@usertrue
142 \let\fixme@setindex\fixme@indextrue
143

```

`\fixme@text` This macro implements the common (textual) part of all notes:

```

144 \def\fixme@text#1#2{%
145   \iffixme@inline\FXInline{#1}{#2}\fi%
146   \iffixme@margin\FXMargin{#1}{#2}\fi%
147   \iffixme@marginclue\FXMarginClue{#1}\fi%
148   \iffixme@footnote\FXFootnote{#1}{#2}\fi%
149   \iffixme@user\FXUser{#1}{#2}\fi}
150

```

Options that control which layouts are selected:

```

151 \DeclareOption{inline}{\fixme@setinline}
152 \DeclareOption{noinline}{\fixme@inlinfalse}

```

```

153 \DeclareOption{margin}{\fixme@setmargin}
154 \DeclareOption{nomargin}{\fixme@marginfalse}
155 \DeclareOption{marginclue}{\fixme@setmarginclue}
156 \DeclareOption{nomarginclue}{\fixme@margincluefalse}
157 \DeclareOption{footnote}{\fixme@footnotetrue}
158 \DeclareOption{nofootnote}{\fixme@footnotefalse}
159 \DeclareOption{user}{\fixme@usertrue}
160 \DeclareOption{nouser}{\fixme@userfalse}
161 \DeclareOption{index}{\fixme@indextrue}
162 \DeclareOption{noindex}{\fixme@indexfalse}
163

```

The following options are here for backward compatibility, but are not used any-more:

```

164 \DeclareOption{info}{}
165 \DeclareOption{noinfo}{}
166 \DeclareOption{warning}{}
167 \DeclareOption{nowarning}{}
168

```

### 6.3 Logging macros

```

\FXNote The macros forcontrolling logging (log file and terminal output):
\FXWarning
\FXError
\FXFatal
169 \newcommand\FXNote[1]{%
170   \GenericInfo{%
171     (FiXme)\@spaces\@spaces\@spaces\@spaces}{%
172     Package FiXme Note: '#1'}}
173 \newcommand\FXWarning[1]{%
174   \PackageWarning{FiXme}{'#1'}}
175 \newcommand\FXError[1]{%
176   \GenericWarning{%
177     (FiXme)\@spaces\@spaces\@spaces\@spaces}{%
178     Package FiXme Error: '#1'}}
179 \newcommand\FXFatal[1]{%
180   \GenericWarning{%
181     (FiXme)\@spaces\@spaces\@spaces\@spaces}{%
182     Package FiXme: '#1'}}
183

```

Boolean that control wether to stay silent or not:

```

184 \newif\iffixme@silent
185

```

Options that control whether to be silent or not:

```

186 \DeclareOption{silent}{\fixme@silenttrue}
187 \DeclareOption{nosilent}{\fixme@silentfalse}
188

```

### 6.4 FiXme notes

```

\thefixmecount Here are the counters:
\thefixmenotecount
\thefixmewarningcount
\thefixmeerrorcount
\thefixmefatalcount
189 \newcounter{fixmecount}
190 \newcounter{fixmenotecount}
191 \newcounter{fixmewarningcount}

```

```

192 \newcounter{fixmeerrorcount}
193 \newcounter{fixmefatalcount}
194

```

\fixmenoteprefix The different flavors of fixme notes. Proper definitions will be selected by the  
\fixmenoteindexname draft or final options. Let's start with the note notes:

```

\fixme@note@draft
\fixme@note@final
195 \newcommand*\fixmenoteprefix{}
196 \newcommand*\fixmenoteindexname{}
197 \def\fixme@note@draft#1{%
198   \fixme@text{\fixmenoteprefix}{#1}%
199   \iffixme@index\FXIndex{\fixmenoteindexname\thefixmenotecount: #1}\fi%
200   \iffixme@silent\else\FXNote{#1}\fi%
201   \addcontentsline{lox}{fixme}{\protect\fixmenoteprefix: #1}}
202 \def\fixme@note@final#1{%
203   \iffixme@silent\else\FXNote{#1}\fi%
204   \addcontentsline{lox}{fixme}{\protect\fixmenoteprefix: #1}}
205

```

\fixmewarningprefix Here are the warnings notes:

```

\fixmewarningindexname
\fixme@warning@draft
\fixme@warning@final
206 \newcommand*\fixmewarningprefix{}
207 \newcommand*\fixmewarningindexname{}
208 \def\fixme@warning@draft#1{%
209   \fixme@text{\fixmewarningprefix}{#1}%
210   \iffixme@index\FXIndex{\fixmewarningindexname\thefixmewarningcount: #1}\fi%
211   \iffixme@silent\else\FXWarning{#1}\fi%
212   \addcontentsline{lox}{fixme}{\protect\fixmewarningprefix: #1}}
213 \def\fixme@warning@final#1{%
214   \iffixme@silent\else\FXWarning{#1}\fi%
215   \addcontentsline{lox}{fixme}{\protect\fixmewarningprefix: #1}}
216

```

\fixmeerrorprefix Here are the errors notes:

```

\fixmeerrorindexname
\fixme@error@draft
\fixme@error@final
217 \newcommand*\fixmeerrorprefix{}
218 \newcommand*\fixmeerrorindexname{}
219 \def\fixme@error@draft#1{%
220   \fixme@text{\fixmeerrorprefix}{#1}%
221   \iffixme@index\FXIndex{\fixmeerrorindexname\thefixmeerrorcount: #1}\fi%
222   \iffixme@silent\else\FXError{#1}\fi%
223   \addcontentsline{lox}{fixme}{\protect\fixmeerrorprefix: #1}}
224 \def\fixme@error@final#1{%
225   \iffixme@silent\else\FXError{#1}\fi%
226   \addcontentsline{lox}{fixme}{\protect\fixmeerrorprefix: #1}}
227

```

\fixmefatalindexname And here are the fatal ones:

```

\fixme@fatal@draft
\fixme@fatal@final
228 \newcommand*\fixmefatalindexname{}
229 \def\fixme@fatal@draft#1{%
230   \iffixme@inline\FiXmeInline{#1}\fi%
231   \iffixme@margin\FiXmeMargin{#1}\fi%
232   \iffixme@marginclue\FXMarginClue{\fixmefatalprefix}\fi%
233   \iffixme@footnote\FiXmeFootnote{#1}\fi%

```

```

234 \iffixme@user\FiXmeUser{#1}\fi%
235 \iffixme@index\FiXmeIndex{%
236     \fixmefatalindexname\thefixmefatalcount: #1}\fi%
237 \iffixme@silent\else\FXFatal{#1}\fi%
238 \addcontentsline{lox}{fixme}{\protect\fixmefatalprefix: #1}}
239 \def\fixme@fatal@final#1{%
240     \PackageError{FiXme}{‘#1’ fixme left in final version}{%
241         You are processing your document in final mode,\MessageBreak
242         but you still have some critical FiXme’s left behind.\MessageBreak
243         Type X to quit, fix your document, and rerun LaTeX.}}
244
\@fixme Here are the notes insertion macros. The different versions are \let to the draft
\fxnote of final flavors defined above by the draft or final options.
\fxwarning
\fxerror
\fixme
245 \def\@@fixme#1#2{%
246     \stepcounter{fixmecount}%
247     \stepcounter{fixme#1count}%
248     \csname fixme@#1\endcsname{#2}}
249 \def\@fixme#1[#2]#3{%
250     \bgroup%
251     \fixme@inlinefalse%
252     \fixme@marginfalse%
253     \fixme@margincluefalse%
254     \fixme@footnotefalse%
255     \fixme@userfalse%
256     \fixme@indexfalse%
257     \@for\fixme@option:=#2\do{%
258         \expandafter\ifx\csname fixme@\fixme@option true\endcsname\relax
259             \PackageError{FiXme}{Unknown layout option ‘\fixme@option’}{%
260                 You have given an invalid option to \string\fixme, or an \string\fx %
261                 command.\MessageBreak
262                 Valid options are ‘inline’, ‘margin’, ‘marginclue’, ‘footnote’, %
263                 ‘user’,\MessageBreak
264                 and ‘index’.\MessageBreak
265                 Type X to quit, fix your document, and rerun LaTeX.}%
266             \else%
267                 \csname fixme@set\fixme@option\endcsname%
268                 \fi}%
269     \@@fixme{#1}{#3}%
270     \egroup}
271 \newcommand\fxnote{\@ifnextchar[%]
272     {\@fixme{note}}{\@@fixme{note}}}
273 \newcommand\fxwarning{\@ifnextchar[%]
274     {\@fixme{warning}}{\@@fixme{warning}}}
275 \newcommand\fxerror{\@ifnextchar[%]
276     {\@fixme{error}}{\@@fixme{error}}}
277 \newcommand\fixme{\@ifnextchar[%]
278     {\@fixme{fatal}}{\@@fixme{fatal}}}
279

```

## 6.5 FiXme environments

The FiXme environments use the corresponding macro version to handle inline (and possibly index) layout, and to perform logging. In final mode, verbatim's comment environment is used to suppress output.

```

280 \newcommand\FXEnvBegin{\begin{quotation}}
281 \newcommand\FXEnvEnd{\end{quotation}}
282
283 \def\@fxenvbegin@draft#1#2{%
284   \FXEnvBegin
285   \edef\fixme@args{inline%
286     \iffixme@index,index\fi%
287     \iffixme@marginclue,marginclue\fi}%
288   \csname#1\endcsname[\fixme@args]{#2}}
289 \def\@fxenvbegin@final#1#2{
290   \edef\fixme@args{inline%
291     \iffixme@index,index\fi%
292     \iffixme@marginclue,marginclue\fi}%
293   \csname#1\endcsname[\fixme@args]{#2}
294   \comment
295 }
296
297 \def\@fxenvend@draft{\FXEnvEnd}
298 \def\@fxenvend@final{\endcomment}
299
300 \newenvironment{anfxnote}[1][ ]{%
301   \@fxenvbegin{fxnote}{#1}}{%
302   \@fxenvend}
303 \newenvironment{anfxwarning}[1][ ]{%
304   \@fxenvbegin{fxwarning}{#1}}{%
305   \@fxenvend}
306 \newenvironment{anfxerror}[1][ ]{%
307   \@fxenvbegin{fxerror}{#1}}{%
308   \@fxenvend}
309 \newenvironment{afixme}[1][ ]{%
310   \@fxenvbegin{fixme}{#1}}{%
311   \@fxenvend}
312

```

## 6.6 Language processing

Options that control which language is selected:

```

313 \DeclareOption{english}{%
314   \renewcommand*\fixmenoteprefix{\fixmelo\nobreakspace Note}
315   \renewcommand*\fixmewarningprefix{\fixmelo\nobreakspace Warning}
316   \renewcommand*\fixmeerrorprefix{\fixmelo\nobreakspace Error}
317   \renewcommand*\fixmefatalprefix{\fixmelo}
318   \renewcommand*\fixmeindexname{\fixmelo}
319   \renewcommand*\fixmenoteindexname{**a@Notes:!!}
320   \renewcommand*\fixmewarningindexname{**b@Warnings:!!}
321   \renewcommand*\fixmeerrorindexname{**c@Errors:!!}
322   \renewcommand*\fixmefatalindexname{}
323   \renewcommand*\listfixmename{List of Corrections}}
324 \DeclareOption{french}{%

```

```

325 \renewcommand*\fixmenoteprefix{\fixmelogo\nobreakspace Note}
326 \renewcommand*\fixmewarningprefix{\fixmelogo\nobreakspace Attention}
327 \renewcommand*\fixmeerrorprefix{\fixmelogo\nobreakspace Erreur}
328 \renewcommand*\fixmefatalprefix{\fixmelogo}
329 \renewcommand*\fixmeindexname{\fixmelogo}
330 \renewcommand*\fixmenoteindexname{**a@Notes:!!}
331 \renewcommand*\fixmewarningindexname{**b@Avertissements:!!}
332 \renewcommand*\fixmeerrorindexname{**c@Erreurs:!!}
333 \renewcommand*\fixmefatalindexname{}
334 \renewcommand*\{listfixmename\}{Liste des Corrections}}
335 \DeclareOption{francais}{\ExecuteOptions{french}}
336 \DeclareOption{spanish}{%
337 \renewcommand*\fixmenoteprefix{\fixmelogo\nobreakspace Nota}
338 \renewcommand*\fixmewarningprefix{\fixmelogo\nobreakspace Aviso}
339 \renewcommand*\fixmeerrorprefix{\fixmelogo\nobreakspace Error}
340 \renewcommand*\fixmefatalprefix{\fixmelogo}
341 \renewcommand*\fixmeindexname{\fixmelogo}
342 \renewcommand*\fixmenoteindexname{**a@Notas:!!}
343 \renewcommand*\fixmewarningindexname{**b@Avisos:!!}
344 \renewcommand*\fixmeerrorindexname{**c@Errores:!!}
345 \renewcommand*\fixmefatalindexname{}
346 \renewcommand*\{listfixmename\}{Lista de Correcciones}}
347 \DeclareOption{italian}{%
348 \renewcommand*\fixmenoteprefix{\fixmelogo\nobreakspace Nota}
349 \renewcommand*\fixmewarningprefix{\fixmelogo\nobreakspace Avviso}
350 \renewcommand*\fixmeerrorprefix{\fixmelogo\nobreakspace Errore}
351 \renewcommand*\fixmefatalprefix{\fixmelogo}
352 \renewcommand*\fixmeindexname{\fixmelogo}
353 \renewcommand*\fixmenoteindexname{**a@Note:!!}
354 \renewcommand*\fixmewarningindexname{**b@Avvisi:!!}
355 \renewcommand*\fixmeerrorindexname{**c@Errori:!!}
356 \renewcommand*\fixmefatalindexname{}
357 \renewcommand*\{listfixmename\}{Corrigenda}}
358 \DeclareOption{german}{%
359 \renewcommand*\fixmenoteprefix{\fixmelogo\nobreakspace Anm}
360 \renewcommand*\fixmewarningprefix{\fixmelogo\nobreakspace Warnung}
361 \renewcommand*\fixmeerrorprefix{\fixmelogo\nobreakspace Fehler}
362 \renewcommand*\fixmefatalprefix{\fixmelogo}
363 \renewcommand*\fixmeindexname{\fixmelogo}
364 \renewcommand*\fixmenoteindexname{**a@Anmerkungen:!!}
365 \renewcommand*\fixmewarningindexname{**b@Warnungen:!!}
366 \renewcommand*\fixmeerrorindexname{**c@Fehler:!!}
367 \renewcommand*\fixmefatalindexname{}
368 \renewcommand*\{listfixmename\}{Verzeichnis der Korrekturen}}
369 \DeclareOption{ngerman}{%
370 \ExecuteOptions{german}}
371 \DeclareOption{danish}{%
372 \renewcommand*\fixmenoteprefix{\fixmelogo\nobreakspace Note}
373 \renewcommand*\fixmewarningprefix{\fixmelogo\nobreakspace Advarsel}
374 \renewcommand*\fixmeerrorprefix{\fixmelogo\nobreakspace Fejl}
375 \renewcommand*\fixmefatalprefix{\fixmelogo}
376 \renewcommand*\fixmeindexname{\fixmelogo}
377 \renewcommand*\fixmenoteindexname{**a@Noter:!!}
378 \renewcommand*\fixmewarningindexname{**b@Advarsler:!!}

```

```

379 \renewcommand*\fixmeerrorindexname{**c@Fejl:!!}
380 \renewcommand*\fixmefatalindexname{}
381 \renewcommand*{\listfixmenename}{Rettelser}}
382 \DeclareOption{croatian}{%
383 \renewcommand*\fixmenoteprefix{\fixmelogo\nobreakspace Poruka}
384 \renewcommand*\fixmewarningprefix{\fixmelogo\nobreakspace Upozorenje}
385 \renewcommand*\fixmeerrorprefix{\fixmelogo\nobreakspace Gre\v ska}
386 \renewcommand*\fixmefatalprefix{\fixmelogo}
387 \renewcommand*\fixmeindexname{\fixmelogo}
388 \renewcommand*\fixmenoteindexname{**a@Poruke:!!}
389 \renewcommand*\fixmewarningindexname{**b@Upozorenja:!!}
390 \renewcommand*\fixmeerrorindexname{**c@Greske:!!}
391 \renewcommand*\fixmefatalindexname{}
392 \renewcommand*{\listfixmenename}{Popis korekcija}}
393

```

## 6.7 Document status processing

\fixme@note Here's where the draft or final version of the internal macros are selected.

```

\fixme@warning
\fixme@error 394 \DeclareOption{final}{
395 \let\fixme@note\fixme@note@final
396 \let\fixme@warning\fixme@warning@final
397 \let\fixme@error\fixme@error@final
398 \let\fixme@fatal\fixme@fatal@final
399 \let\@fxenvbegin\@fxenvbegin@final
400 \let\@fxenvend\@fxenvend@final
401 \let\listoffixmes\listoffixmes@final
402 }
403
404 \DeclareOption{draft}{
405 \@ifclassloaded{article}{
406 \let\@listoffixmes@pretoc\@listoffixmes@pretoc@article
407 \let\@listoffixmes@posttoc\@listoffixmes@posttoc@article}{
408 \@ifclassloaded{report}{
409 \let\@listoffixmes@pretoc\@listoffixmes@pretoc@report
410 \let\@listoffixmes@posttoc\@listoffixmes@posttoc@report}{
411 \@ifclassloaded{book}{
412 \let\@listoffixmes@pretoc\@listoffixmes@pretoc@book
413 \let\@listoffixmes@posttoc\@listoffixmes@posttoc@book}{
414 \@ifclassloaded{scrartcl}{
415 \let\@listoffixmes@pretoc\@listoffixmes@pretoc@scrartcl
416 \let\@listoffixmes@posttoc\@listoffixmes@posttoc@scrartcl}{
417 \@ifclassloaded{scrreprt}{
418 \let\@listoffixmes@pretoc\@listoffixmes@pretoc@scrreprt
419 \let\@listoffixmes@posttoc\@listoffixmes@posttoc@scrreprt}{
420 \@ifclassloaded{scrbook}{
421 \let\@listoffixmes@pretoc\@listoffixmes@pretoc@scrbook
422 \let\@listoffixmes@posttoc\@listoffixmes@posttoc@scrbook}{
423 \@ifclassloaded{amsbook}{
424 \let\listoffixmes@draft\listoffixmes@draft@ams}{
425 \@ifclassloaded{amsart}{
426 \let\listoffixmes@draft\listoffixmes@draft@ams}{
427 %% Use the article layout by default.

```



```

428 \let\@listoffixmes@pretoc\@listoffixmes@pretoc@article
429 \let\@listoffixmes@posttoc\@listoffixmes@posttoc@article}
430 }
431 }
432 }
433 }
434 }
435 }
436 }
437 \let\fixme@note\fixme@note@draft
438 \let\fixme@warning\fixme@warning@draft
439 \let\fixme@error\fixme@error@draft
440 \let\fixme@fatal\fixme@fatal@draft
441 \let\@fxenvbegin\@fxenvbegin@draft
442 \let\@fxenvend\@fxenvend@draft
443 \let\listoffixmes\listoffixmes@draft
444 }
445

```

Put us in `english` and `final` mode, and enable the marginal notes. Note that as documented, marginal notes are incompatible with the ACM SIG classes. Initially, I thought I would detect these classes and issue an error if marginal layout (or clue) is active. However, I changed my mind, because nothing prevents somebody to write a new class on top of these ones and authorize `\marginpar` back again. Normally these classes issue an error if `\marginpar` is used. However, the 2.3 / June 2007 versions are buggy and the error actually triggers a stack overflow in L<sup>A</sup>T<sub>E</sub>X... Patch on its way. Oh boy, these classes are a mess.

```

446 \ExecuteOptions{english,final,margin,nosilent}
447 \ProcessOptions*
448

```

Finally, ensure backward compatibility for `\fixme` with `\FiXmeUser`, and output a summary giving the number of fixme notes at the end of the compilation:

```

449 \AtBeginDocument{%
450   \ifx\FiXmeUser\empty
451     \renewcommand\FiXmeUser[1]{\FXUser{\fixmefatalprefix}{#1}}
452   \fi}
453
454 \AtEndDocument{%
455   \GenericWarning{%
456     (FiXme)\@spaces\@spaces\@spaces\@spaces
457   }{%
458     Package FiXme Summary: Number of notes: \thefixmenotecount,\MessageBreak%
459     Number of warnings: \thefixmewarningcount,\MessageBreak%
460     Number of errors: \thefixmeerrorcount,\MessageBreak%
461     Number of fatal errors: \thefixmefatalcount,\MessageBreak%
462     Total: \thefixmecount\@gobble%
463   }}

```

Well, I think that's it. Enjoy using FiXme!