

# Editorial Notes for L<sup>A</sup>T<sub>E</sub>X\*

Michael Kohlhase  
Computer Science, Jacobs University  
<http://kwarc.info/kohlhase>

November 4, 2007

## Abstract

This package defines a couple of editorial notes that simplify collaboration on a L<sup>A</sup>T<sub>E</sub>X text. These allow authors to annotate status information in the source. In draft mode, the annotations are shown for communication, and in publication mode these are suppressed.

## 1 Introduction

When collaborating on a document with multiple authors, communication about the status of a given passage and intentions about what to do with it, indications about issues still need to be resolved, and directives to other authors, e.g. calling for help, or passing the baton, etc. make up much of the overhead involved in collaboration. In particular, many of the necessary communicative acts are centered around specific points or passages in the document at hand. Therefore it is natural to embed the communicative acts in the document source itself. The simplest version of this is to special markers like “(\*\*\*) remember to rework this before publication (\*\*\*)”, where the markers “(\*\*\*)” and “(\*\*\*)” serve as a visual aid and target for search and navigation tools in the editor. Of course this is dangerous, and we have all seen published texts with such markers still present.

The L<sup>A</sup>T<sub>E</sub>X package described in here systematizes the idea and provides more conspicuous visual markers (as footnotes and margin notes) and a way of making all all of these private markers and comments invisible for publication or outside communication.

## 2 The User Interface

As usual in L<sup>A</sup>T<sub>E</sub>X, the package is loaded by `\usepackage[options]{ed}`, where [*options*] is optional and gives a comma separated list of options. Currently the `ed` package only takes two options `show` and `hide`, where `hide` is the default case, so that `\usepackage{ed}`, is equivalent to `\usepackage[hide]{ed}`. If the `show` option is given, then the editorial notes are presented as special annotations to the document, otherwise they are completely invisible — if you take care about spaces in the source code. For instance `text\ednote{...}_text` will fool T<sub>E</sub>X’s whitespace-collapsing algorithm and bring it to output two spaces in the document instead of just one as expected `text\ednote{...}_text` should be used instead! This `hide` option useful for preparing “clean” version for outside consumption without loosing the management metadata.

### 2.1 Annotation Macros

`\ednote` The main user macro of this package is `\ednote`. It is used say what you have done or what should still be done at a given point in the document. `ed.sty` formats it like a footnote, but

---

\*Version v1.3 (last revised 2007/10/03)

with a margin note that marks the place in the text, where the note is located. Otherwise, in the presence of multiple `\ednotes` in a page it may be difficult to find the referenced locations<sup>1</sup>. Editorial notes are numbered and marked in the margin for easy recognition. `\ednote` also takes an optional argument that is an identifier. This allows to cross-reference ednotes in each other.

The `\issue` macro<sup>2</sup> is a variant of `\ednote` for issues that still have to be discussed among the authors. For issues the labeling and cross-referencing in the optional argument is especially useful.

The `\tweak` macro<sup>3</sup> is a variant of `\ednote` for marking places where we have tweaked something (apart from the standard way we usually use).

The `todo` environment is a environment that can be used to mark up writing tasks. These are inserted into the text in a different font, so that they are conspicuous as a foreign part, and take an argument for a comment.

The `todolist` is a variant of the `todo` environment which is a is an itemized list.

The following is an example, generated form the the input `\begin{todolist}{an example todolist} ... \item ... \end{todolist}`. To Do: an example todolist

- lots of good examples
- a general chapter about best practices

the `newpart` environment can be used to mark up changed text blocks. `\begin{newpart}` takes an argument that is interpreted as a comment and is treated like an `\ednote` comment. The `oldpart` environment is similar to `newpart` but is used for old parts of text copied from another document that still need to be changed in a document.

The annotation macros have capitalized variants (`\Ednote`, `\Issue`, `\Tweak`, `Todo`, `Newpart`, `Oldpart`) that do not make location marks in the margin. This is convenient in situations (e.g. inside boxes or minipages) that allow footnotes, but no margin notes.

## 2.2 Generating Statistics and Explanations

Putting the macro `\ednotemessage` just before the `\end{document}` will generate a message with cardinality information for the ednotes into the log file.

The `\edexplanation` macro generates an explanation of the best practices into the document. So that if you introduce collaboration partners to the `ed` package, you can give them an explanation “in-place”.

## 2.3 Configuration

The font shape of editorial annotations is governed by the parameter `\ednoteshape` the default is sans serif, specialize it to say italic by `\def\ednoteshape{\it}`

## 2.4 Best Practices

In a situation with multiple authors, we it is advisable to use the `\ednote` in the following form: `\ednote{author: some explanatory text}` to specify the originator of the note.

<sup>1</sup>EDNOTE: this is an example of an ednote

<sup>2</sup>ISSUE: this is an example of an issue

<sup>3</sup>TWEAK: this is an example of an tweak

### 3 The Implementation

The implementation is rather standard. We first set up the options for the package. The main switch is `\showednotes`, which governs the visibility of the annotations.

```
1 (*package)
2 \newif\ifshowednotes\showednotesfalse
```

the next step is to declare the package options, they just set `\showednotes` switch accordingly.

```
3 \DeclareOption{show}{\showednotestrue}
4 \DeclareOption{hide}{\showednotesfalse}
5 \ProcessOptions
```

The next step is to load the `verbatim` package, so that we can comment out the `todo` lists.

```
6 \ifshowednotes\else\usepackage{verbatim}\fi
```

This ends the package setup code, so we can come to the implementation of the functionality of the package.

`\ednoteshape` We start with the configuration part, predefining `\epdnoteshape` to be sans serif.

```
7 \def\ednoteshape{\sffamily}
```

The next step is to set up a counter for the editorial annotations

```
8 \newcounter{ednote}
```

`\ed@foot` The internal macro `\ed@foot` is used to actually make the annotations, it is used by the interface macros to give the annotations. It takes three arguments: A comment text that goes into the footnote, a type descriptor, and an identifier.

```
9 \def\ed@foot#1#2#3% text, type, label
10 {\def\test{#3}\def\empty{}\footnotetext[\value{ednote}]%
11 {\scshape{#2}\if\test\empty\else\label{ed:#3}[{#3}]\fi:} \ednoteshape #1}}
```

`\Ed@note` Another internal macro `\Ed@note` adds label management to `\ed@foot`

```
12 \def\Ed@note#1#2#3% text, type, label
13 {\addtocounter{ednote}{1}\message{#2!}%
14 \ifshowednotes\footnotemark[\arabic{ednote}]\ed@foot{#1}{#2}{#3}\fi}
```

`\ed@note` `\ed@note` is a variant of `\Ed@note` that also makes an identifying mark in the margin.

```
15 \def\ed@note#1#2#3% text, type, label
16 {\Ed@note{#1}{#2}{#3}\ifshowednotes\marginpar{#2(\arabic{ednote})}\fi}
```

`\ednote` with the `\Ed@note` and `\ed@note` macros it is very simple to get the desired functionality of `\Ednote` and `\ednote`:

```
17 \newcommand{\Ednote}[2] [] {\Ed@note{#2}{EdNote}{#1}}
18 \newcommand{\ednote}[2] [] {\ed@note{#2}{EdNote}{#1}}
```

`\tweak` and of course for `\issue`:

```
19 \newcommand{\tweak}[2] [] {\ed@note{#2}{Tweak}{#1}}
20 \newcommand{\Tweak}[2] [] {\Ed@note{#2}{Tweak}{#1}}
```

`\issue`

```
21 \newcommand{\issue}[2] [] {\ed@note{#2}{Issue}{#1}}
22 \newcommand{\Issue}[2] [] {\Ed@note{#2}{Issue}{#1}}
```

`new@part` For the text status environments `newpart` and `oldpart` we also set up an internal macro that does the work.

```
23 \def\new@part#1#2#3% text, mess, start
24 {\addtocounter{ednote}{1}\edef\new@number{\theednote}\message{#2!\new@number}}
25 \ifshowednotes\ed@foot{#1}{#2}{}\marginpar{#3(\new@number)}\fi}
26 \def\endnew@part#1% end
27 {\ifshowednotes\marginpar{#1(\new@number)}\fi}
```

`newpart` We instantiate it for the `newpart` environment

```
28 \newenvironment{Newpart}[1]{\new@part{#1}{New Part}{BegNP}}{}
29 \newenvironment{newpart}[1]{\new@part{#1}{New Part}{BegNP}}{\endnew@part{EndNP}}
```

`oldpart` and of course for the `oldpart` environment

```
30 \newenvironment{Oldpart}[1]{\new@part{#1}{Old Part}{BegOP}}{}
31 \newenvironment{oldpart}[1]{\new@part{#1}{Old Part}{BegOP}}{\endnew@part{EndOP}}
```

`todo` How we define the `todo` environment depends on the `\showednotes` switch (or the package option). If we hide annotations, `todo` is set to `comment` from the `comment` package, otherwise the body is set in sans serif font for emphasis.

```
32 \newcommand{\Todo}[2] [] {\Ed@note{#2}{ToDo}{#1}\ifshowednotes\bgroup\sffamily\else\comment\fi}
33 \def\endTodo{\ifshowednotes\egroup\else\endcomment\fi}
34 \newcommand{\todo}[2] [] {\ed@note{#2}{ToDo}{#1}\ifshowednotes\bgroup\sffamily\else\comment\fi}
35 \def\endtodo{\endTodo}
```

`todolist` How we define the `todolist` environment depends on the `\showednotes` switch (or the package option). If we hide annotations, `todolist` is set to `comment` from the `comment` package, otherwise it is set to an `itemize`.

```
36 \def\Todolist#1{% the comment
37 \ifshowednotes\message{todolist!}{\sffamily To Do: #1}\bgroup\sffamily\begin{itemize}%
38 \else\comment%
39 \fi}
40 \def\endTodolist{\ifshowednotes\end{itemize}\egroup\else\endcomment\fi}
41 \def\todolist#1{\ifshowednotes\marginpar{\sffamily To Do}}\Todolist{#1}\fi}
42 \def\endtodolist{\endTodolist}
```

### 3.1 Generating Statistics and Explanations

`\ednotemessage` The `\ednotemessage` makes use of the counter `ednote` and generates a message.

```
43 \def\ednotemessage{\ifnum\value{ednote}>0\typeout{}}%
44 \typeout{There are still \arabic{ednote} EdNotes and Issues to resolve!}%
45 \typeout{}\fi}
```

`\edexplanation` The `\edexplanation` macro makes use of the `todolist` environment.

```
46 \long\def\edexplanation{\todolist{we will use the ednote system to communicate}
47 \item use the {\tt{\char92ednote\char123author: some explanatory text\char125}}
48 like a footnote to say what you have done or what should still be
49 done\ednote{MiKo: this is an example of an ednote}. Ednotes are numbered and
50 marked in the margin for easy recognition.
51 \item use the {\tt{\char92issue\char123author: explanation of the
52 issue\char125}} variant of ednote for issues\issue{this is an example of
53 an issue} that still have to be discussed.
54 \item finally, the {\tt{todolist}} environment is a list environment that can be
55 used to mark up todo lists. This explanation is an example of a todo list, it
56 is inserted into the text in a different font.
57 \item the {\tt{newpart}} environment can be used to mark up changed text blocks.
58 {\tt{\char92begin\char123newpart\char125}} takes an argument that is
59 interpreted as a comment and is treated like an {\tt{\char92ednote}} comment.
60 \item the {\tt{oldpart}} environment is similar to {\tt{newpart}} but is used
61 for old parts of text copied from another document that still need to be
62 changed in a document.
63 \item putting the macro {\tt{\char92ednotemessage}} just before the
64 {\tt{\char92end\char123document\char125}} will generate a message with
65 cardinality information for the ednotes into the log file.
66 \item all of these text decorations and meta-annotations are only inserted into
67 the text, if the {\tt{show}} package option in the {\tt{\char92 usepackage}}
```

```
68 directive in the preamble of the document is set: {\tt{\char92
69   usepackage[show]\char123ed\char125}} will show the decorations, while
70 {\tt{\char92 usepackage\char123ed\char125}} will not. This is useful for
71 preparing ‘‘clean’’ version for outside consumption without loosing the
72 management metadata.
73 \endtodolist}
74 \end{package}
```

## Change History

v1.0		v1.2	
General: First Version with Documentation	1	General: adding todo environment	1
v1.1		v1.3	
General: Added capitalized variants	1	General: rationalizing todo environment	1