

# L'extension **comment**\*

Victor Eijkhout  
[victor@eijkhout.net](mailto:victor@eijkhout.net)

Octobre 1999

## 1 Objectif

Cette extension inclut ou exclut de façon sélective des blocs de texte dans la restitution finale. L'utilisateur peut définir de nouvelles versions de commandes de commentaire, chacune pouvant être contrôlée séparément. Des commentaires spéciaux peuvent être définis : l'utilisateur y spécifie l'action qui doit être faite pour chacune des lignes mises en commentaire.

Cette extension peut être utilisée avec PLAIN  $\text{\TeX}$  ou  $\text{\LaTeX}$ , comme avec la plupart des extensions existantes.

## 2 Utilisation

Tout le texte placé entre les commandes

```
\comment ... \endcomment
```

ou

```
\begin{comment} ... \end{comment}
```

est ignoré.

Les commandes de début et de fin doivent apparaître chacune sur une ligne à part. Il ne doit pas y avoir d'espace et d'éléments avant ou après la commande sur ces lignes. Cet environnement devrait pouvoir fonctionner avec une quantité de commentaire arbitraire et le commentaire peut être un texte quelconque.

D'autres environnements à l'image de `comment` peuvent être définis et sélectionnés et désélectionnés avec

---

\*Ce manuel correspond à la version `comment` v3.6, datée d'octobre 1999.

```
\includecomment{versiona}  
\excludecomment{versionb}
```

Ces environnements sont utilisables avec

```
\versiona ... \endversiona
```

ou

```
\begin{versiona} ... \end{versiona}
```

avec, à nouveau, les commandes de commentaires de début et de fin sur des lignes à part.

Note pour les utilisateurs de L<sup>A</sup>T<sub>E</sub>X : pour un commentaire inclus, les lignes `\begin` et `\end` agissent comme si elles n'existaient pas. En particulier, elles n'impliquent pas de groupe, aussi les instanciations et autres ne sont pas locales.

Les commentaires spéciaux sont définis par

```
\specialcomment{<nom>}{<commandes avant>}{<commandes après>}
```

où le deuxième argument et le troisième argument sont exécutés respectivement avant et après chaque bloc de commentaire. Vous pouvez utiliser ceci pour des commandes de mise en forme globales. Pour garder ces réglages locaux, vous pouvez inclure un `\begingroup` dans les « *<commandes avant>* » et `\endgroup` dans les « *<commandes après>* ». Par exemple :

```
\specialcomment{smalltt}  
  {\begingroup\ttfamily\footnotesize}\endgroup}
```

✖ You do *not* have to do an additional

```
\includecomment{smalltt}
```

To remove ‘`smalltt`’ blocks, give `\excludecomment{smalltt}` after the definition.

Processing comments can apply processing to each line.

```
\processcomment{<name>}{<each-line commands>}{<before  
commands>}{<after commands>}
```

By defining a control sequence

```
\def\Thiscomment##1{...}
```

in the before commands the user can specify what is to be done with each comment line. BUG this does not work quite yet BUG

Trick for short in/exclude macros (such as `\maybe{this snippet}`) :

```

\includecomment{cond}
\newcommand{\maybe}[1]{ }
\begin{cond}
\renewcommand{\maybe}[1]{#1}
\end{cond}

```

### 3 Basic approach of the implementation :

to comment something out, scoop up every line in verbatim mode as macro argument, then throw it away. For inclusions, in L<sup>A</sup>T<sub>E</sub>X the block is written out to a file `\CommentCutFile` (default “comment.cut”), which is then included. In plain T<sub>E</sub>X (and other formats) both the opening and closing comands are defined as noop.

### 4 Changes in version 3.1

- updated author’s address
- cleaned up some code
- trailing contents on `\begin{<env>}` line is always discarded even if you’ve done `\includecomment{<env>}`
- comments no longer define grouping!! you can even
 

```

\includecomment{env}
\begin{env}
\begin{itemize}
\end{env}

```

 Isn’t that something...
- included comments are written to file and input again.

### 5 Changes in 3.2

- `\specialcomment` brought up to date (thanks to Ivo Welch).

### 6 Changes in 3.3

- updated author’s address again
- parametrised `\CommentCutFile`

### 7 Changes in 3.4

- added GNU public license

- added `\processcomment`, because Ivo's fix (above) brought an inconsistency to light.

## 8 Changes in 3.5

- corrected typo in header.
- changed author email
- corrected `\specialcomment` yet again.
- fixed `excludecomment` of an earlier defined environment.

## 9 Changes in 3.6

- The 'cut' file is now written more verbatim, using `\meaning`; some people reported having trouble with ISO latin 1, or `umlaute.sty`.
- removed some `\newif` statements. Has this suddenly become `\outer` again?

## 10 Known bugs :

- `excludecomment` leads to one superfluous space
- `processcomment` leads to a superfluous line break