

# 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}
```

Vous *n'avez pas* besoin d'ajouter en complément :

```
\includecomment{smalltt}
```

Pour retirer les blocs « `smalltt` », indiquez `\excludecomment{smalltt}` après la définition.

Le traitement des commentaires peut appliquer un traitement à chaque ligne.

```
\processcomment{<nom>}{<commandes par ligne>}{<commandes  
avant>}{<commandes après>}
```

En définissant la commande

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

dans les *commandes avant*), l'utilisateur peut indiquer ce qui doit être fait pour chaque ligne de commentaire. Ceci ne fonctionne pas bien pour le moment.

Voici un exemple de code pour de courtes commandes de commentaires (telle que `\masque{ce petit texte}`) :

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

### 3 Approche simplifiée de l'implémentation

Pour commenter quelque chose, ramasser chaque ligne en mode verbatim comme argument de commande et le jeter. Pour les inclusions, avec L<sup>A</sup>T<sub>E</sub>X, le bloc est placé dans un fichier `\CommentCutFile` (par défaut « `comment.cut` ») qui est alors inclus. En PLAIN TeX (et d'autres formats), les commandes d'entrée et de sortie sont définies comme des commandes sans aucun effet.

### 4 Changements en version 3.1

- mise à jour de l'adresse de l'auteur ;
- nettoyage d'une partie du code ;
- les éléments suivants `\begin{env}` sur sa ligne sont toujours ignorés même si vous avez indiqué `\includecomment{env}`  
`\begin{env}`  
`\begin{itemize}`  
`\end{env}`  
Plutôt impressionnant...
- les commentaires inclus sont écrits dans un fichier puis réinsérés.

### 5 Changements en 3.2

- mise à jour de `\specialcomment` (merci à Ivo Welch).

### 6 Changements en 3.3

- mise à jour de l'adresse de l'auteur une nouvelle fois ;

- la commande `\CommentCutFile` devient paramétrable.

## 7 Changements en 3.4

- ajout de la licence publique GNU ;
- ajout de la commande `\processcomment` car la correction d'Ivo (ci-dessus) a fait ressortir un manque de cohérence.

## 8 Changements en 3.5

- correction d'une erreur d'orthographe dans le titre ;
- mise à jour de l'adresse de l'auteur ;
- nouvelle correction de la commande `\specialcomment` ;
- correction de l'`\excludcomment` d'un environnement défini plus tôt.

## 9 Changements en 3.6

- le fichier « coupé » est maintenant écrit un peu plus en mode verbatim, en utilisant `\meaning` ; certaines personnes avaient indiqué avoir rencontré des problèmes avec l'encodage ISO latin 1 ou l'extension `umlaute`.
- retrait de conditions avec `\newif`. Cette dernière serait-elle soudain devenue une commande `\outer` ?

## 10 Anomalies connues

- la commande `\excludcomment` conduit à une espace superflue.
- la commande `\processcomment` conduit à un saut de ligne superflu.