

A New Implementation of LaTeX's `verbatim` and `verbatim*` Environments.

Rainer Schöpf
Internet: Schoepf@Uni-Mainz.DE

Bernd Raichle
Stettener Str. 73
D-73732 Wäldenbronn
Federal Republic of Germany
Internet: raichle@azu.Informatik.Uni-Stuttgart.DE

Chris Rowley
The Open University
Parsifal College
Finchley Road
London NW3 7BG, UK
Internet: C.A.Rowley@open.ac.uk

2001/03/12

Abstract

This package reimplements the LaTeX `verbatim` and `verbatim*` environments. In addition it provides a `comment` environment that skips any commands or text between `\begin{comment}` and the next `\end{comment}`. It also defines the command `verbatiminput` to input a whole file `verbatim`.

1 Usage notes

LaTeX's `verbatim` and `verbatim*` environments have a few features that may give rise to problems. These are:

- Due to the method used to detect the closing `\end{verbatim}` (i.e. macro parameter delimiting) you cannot leave spaces between the `\end` token and `{verbatim}`.
- Since TeX has to read all the text between the `\begin{verbatim}` and the `\end{verbatim}` before it can output anything, long `verbatim` listings may overflow TeX's memory.

Whereas the first of these points can be considered only a minor nuisance the other one is a real limitation.

This package file contains a reimplementation of the `verbatim` and `verbatim*` environments which overcomes these restrictions. There is, however, one incompatibility between the old and the new implementations of these environments: the old version would treat text on the same line as the `\end{verbatim}` command as if it were on a line by itself.

This new version will simply ignore it.

(This is the price one has to pay for the removal of the old `verbatim` environment's size limitations.) It will, however, issue a warning message of the form

```
LaTeX warning: Characters dropped after \end{verbatim}!
```

This is not a real problem since this text can easily be put on the next line without affecting the output.

This new implementation also solves the second problem mentioned above: it is possible to leave spaces (but *not* begin a new line) between the `\end` and the `{verbatim}` or `{verbatim*}`:

```
\begin {verbatim*}
  test
  test
\end {verbatim*}
```

Additionally we introduce a `comment` environment, with the effect that the text between `\begin{comment}` and `\end{comment}` is simply ignored, regardless of what it looks like. At first sight this seems to be quite different from the purpose of `verbatim` listing, but actually the implementation of these two concepts turns out to be very similar. Both rely on the fact that the text between `\begin{...}` and `\end{...}` is read by TeX without interpreting any commands or special characters. The remaining difference between `verbatim` and `comment` is only that the text is to be typeset in the first case and to be thrown away in the latter. Note that these environments cannot be nested.

`\verbatiminput` is a command with one argument that inputs a file `verbatim`, i.e. the command `verbatiminput{xx.yy}` has the same effect as

```
\begin{verbatim}
<Contents of the file xx.yy>
\end{verbatim}
```

This command has also a `*`-variant that prints spaces as `␣`.