The spverbatim package*

Scott Pakin
scott+spverb@pakin.org

August 10, 2009

1 Introduction

LaTeX's `\verb` macro treats its argument as an unbreakable unit of text. This can lead to poor typesetting, especially when the argument is long:

> Be sure to run "`my_program | awk '$1 ~ /^[0-9]+$/ {printf "%s & %s \\\n", $2, $NF}' >` to extract the data. Otherwise, when you run "`bad_program && rm $HOME/.*#backup_file#~`", the program will delete all of your files.

The spverbatim package enables LaTeX to break lines at spaces within verbatim text:

> Be sure to run "`my_program | awk '$1 ~ /^[0-9]+$/ {printf "%s & %s \\\n", $2, $NF}' > $HOME/.myprogrc`" to extract the data. Otherwise, when you run "`bad_program && rm $HOME/.*#backup_file#~`", the program will delete all of your files.

2 Usage

`\spverb` The spverbatim package provides an `\spverb` macro that resembles `\verb` except that it allows line breaks at space characters. Like `\verb`, `\spverb` must be followed by a character that ends the verbatim text on its second occurrence:

`\spverb` ⟨*char*⟩ ⟨*literal text*⟩ ⟨*char*⟩

The following shows how the final sentence in Section 1 was entered:

```
 Otherwise, when you run
``\spverb!bad_program && rm $HOME/.*#backup_file#~!'', the program will
 delete all of your files.
```

*This document corresponds to spverbatim v1.0, dated 2009/08/10.

Although \spverb allows line breaks on *output*, it does not allow line breaks on *input* (i.e., within ⟨*literal text*⟩). Hence, the following LaTeX code is **incorrect**:

```
Otherwise, when you run ``\spverb!bad_program && rm
$HOME/.*#backup_file#~!'', the program will delete all of
your files.
```

spverbatim

In addition to the \spverb macro, spverbatim provides an spverbatim environment. spverbatim resembles verbatim except that it allows line breaks at space characters. For example,

```
\begin{spverbatim}
  my_program | awk '$1 ~ /^[0-9]+$/ {printf "%s & %s \\\n", $2, $NF}' > $HOME/.myprogrc
\end{spverbatim}
```

produces

```
my_program | awk '$1 ~ /^[0-9]+$/ {printf "%s & %s \\\n", $2, $NF}' >
$HOME/.myprogrc
```

Unlike \verb and verbatim, \spverb and spverbatim do not support a *-form in which space characters are typeset as "␣". Please contact the author if this is a feature you'd like to see in spverbatim.

3 Implementation

This section presents the complete source code for the spverbatim package. Unless you're interested in seeing precisely how spverbatim works, there's no need to read any further.

\spverb
\spverb@ve
\@xobeysp

To avoid rewriting \verb and all of its helper macros we begin a group; locally redefine the nonbreaking space macro, \@xobeysp, to produce a breaking space; and locally redefine the end-of-\verb macro, \verb@egroup, to end the extra group we began.

```
\gdef\spverb{%
  \bgroup
  \let\spverb@ve=\verb@egroup
  \def\verb@egroup{\spverb@ve\egroup}%
  \def\@xobeysp{\mbox{}\space}%
  \verb
}
```

\spv@xverbatim

`verbatim` is no ordinary LaTeX environment. Because "`\`", "`{`", and "`}`" are treated as literals within a `verbatim` environment, `\end{verbatim}` can't automatically end the environment. Rather, `\begin{verbatim}` invokes the `\@xverbatim` macro, which pattern-matches against the literal text "`\end{verbatim}`", typesets everything it finds up to that text, and finally invokes the real "`\end{verbatim}`" sequence. Here, we define an `\spv@xverbatim` macro that's just like `\@xverbatim` except that it pattern-matches against the literal text "`\end{spverbatim}`" and ends with a call to the real "`\end{spverbatim}`" sequence.

```
\begingroup
  \catcode`|=0
  \catcode`[=1
  \catcode`]=2
  \catcode`\{=12
  \catcode`\}=12
  \catcode`\\=12
  |gdef|spv@xverbatim#1\end{spverbatim}[#1|end[spverbatim]]
|endgroup
```

spverbatim

Because the `verbatim` environment already begins a new group, all we have to do to get it to preserve spaces is locally redefine the nonbreaking space macro, `\@xobeysp`, to produce a breaking space; and locally redefine `\@xverbatim` as `\spv@xverbatim` (see above) so that a `\begin{spverbatim}` is matched by an `\end{spverbatim}`, not an `\end{verbatim}`.

```
\newenvironment{spverbatim}{%
  \def\@xobeysp{\mbox{}\space}%
  \let\@xverbatim=\spv@xverbatim
  \verbatim
}{%
}
```

4 Legal notices

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.