

The `newverbs` package

Martin Scharrer
martin@scharrer-online.de

v1.0 from 2010/04/05

Abstract

This package allows the definition of `\verb` variants which add TeX code before and after the verbatim text. When used together with the `shortvrb` package it allows the definition of short verbatim characters which use this variants instead of the normal `\verb`.

1 Usage

1.1 Defining new variants of `\verb`

`\newverbcommand`
`\renewverbcommand`
`\provideverbcommand`

This macros allow the definition of `\verb` variants. The verbatim content is processed using `\verb`, but the variants can add TeX code before and after it. The syntax is `\dots\verbcommand{<macro>}{<code before>}{<code after>}`. The three definition macros use `\newcommand*`, `\renewcommand*` and `\providecommand*` internal to define `<macro>`, respectively. Afterwards `<macro>` can be used like `\verb`. The star version of `<macro>` will use `\verb*`.

See the implementation of `\qverb` in section 3.2 for an example.

1.2 Provided `\verb` variants

Two `\verb` variants are provided (i.e. with `\provideverbcommand`) by default.

`\qverb` This macro adds quote characters around the verbatim material. Two macros are used to insert the quotes: `\qverbbeginquote` (‘) and `\qverbendquote` (’). They can be redefined by the user if required.

Using `\qverb<char><verbatim material><char>` is equal to `\qverbbeginquote\verb<char><verbatim material><char>\qverbendquote`, or ‘`\verb<char><verbatim material><char>`’ when the default definition of the quote macros is used.

`\fverb` This macro adds a frame (`\fbox{}`) around the verbatim text (`\fverb+$_$+` → `$$_$`). A TeX box is used to store the content first, then the box is framed. The user can define similar command using the following code:

```
\newverbcommand{\myverb}{\begin{lrbox}{\verbbox}}
{\end{lrbox}\mycommand{\usebox{\verbbox}}}
```

The temporary box `\verbbox` is only provided inside a `\dots\verbcommand`.

1.3 Using `\verb` variants with short verbatim character

`\MakeSpecialShortVerb` This package also defines a special version of the `\MakeShortVerb` macro from the `shortvrb` package. The original command `\MakeShortVerb*{\langle char \rangle}` changes the meaning of `\langle char \rangle` so that `\langle char \rangle\langle verbatim material \rangle\langle char \rangle` is a shorter alternative to `\verb*\langle char \rangle\langle verbatim material \rangle\langle char \rangle`.

The new macro `\MakeSpecialShortVerb*{\langle verb variant \rangle}{\langle char \rangle}` does the same, but instead of `\verb*` it uses a `\langle verb variant \rangle` which needs to be defined using `\newverbcommand`. The package `shortvrb` must be loaded by the user in order to make this macro work. It is not loaded automatically by `newverbs`.

The special meaning of `\langle char \rangle` can be removed using `shortvrb`'s `\DeleteShortVerb`, i.e. the same way as for characters defined with the normal `\MakeShortVerb`. If a character was already made a short verbatim character it must be “deleted” before it can be redefined by `\MakeShortVerb` or `\MakeSpecialShortVerb`.

Examples:

`\MakeSpecialShortVerb{\qverb}{\}` will make ‘`’` a short, quoting verbatim character: `"$^&$" → ‘$^&$’`.

`\DeleteShortVerb{\}\MakeSpecialShortVerb{\fverb}{\}` will change it definition to use `\fverb`: `$^&$`.

2 Compatibility with other verbatim packages

The compatibility with other verbatim packages is not tested yet. This package relies on the normal internal definition of `\verb` and `\MakeShortVerb`. Any package which changes these might break this package. Users which encounter incompatibilities should not hesitate to contact the package author (with details!).

3 Implementation

3.1 Verb Definition Commands

`\newverbcommand` This macro calls the real macro with the to be used definition macro.
`\renewverbcommand` 1 `\newcommand*\newverbcommand{\new@verbcommand\newcommand}`
`\provideverbcommand` 2 `\newcommand*\renewverbcommand{\new@verbcommand\renewcommand}`
3 `\newcommand*\provideverbcommand{\new@verbcommand\providecommand}`

`\new@verbcommand` Depending on #1 a macro given by #2 is (re-)defined/provided. The code given by #3 and #4 is placed before and after the verb command. The trailing code is inserted by patching `\verb@egroup` which is called by `\verb` after the verbatim content.

```
4 \def\new@verbcommand#1#2#3#4{%  
5   #1*#2{%  
6     \begingroup  
7     \newverbcommand@settings  
8     \def\verb@egroup{\verb@orig@egroup#4\endgroup}%
```

```

9     #3\verb
10  }%
11 }

```

`\newverbcommand@settings` Some settings required for all new `\verb`-like commands. The original end group macro from `\verb` is saved away. Also the temp box `a` is provided with a user friendly name.

```

12 \def\newverbcommand@settings{%
13   \let\verb@orig@egroup\verb@egroup
14   \let\verbbox\@tempboxa
15 }

```

3.2 Provided New Verb Commands

`\qverb` Quoting version of `\verb`. Places a quote character before and after the verbatim content: `'verb'`.

```

16 \provideverbcommand{\qverb}{\qverbbeginquote}{\qverbendquote}

```

`\qverbbeginquote` `\qverbendquote` This macros insert the actual quotes. They can be redefined by the user to contain the required quotes.

```

17 \def\qverbbeginquote{' }
18 \def\qverbendquote{' }

```

`\fverb` A framed version of `\verb`. Stores the verbatim content first into a box. Then the box content is framed.

```

19 \newverbcommand{\fverb}
20   {\setbox\verbbox\hbox\bgroup\color@setgroup}
21   {\color@endgroup\egroup\fbbox{\box\verbbox}}

```

3.3 Make Special Short Verbatim Characters

`\MakeSpecialShortVerb` Uses the definition of `\MakeShortVerb` from `shortvrb` except with `\verb` replaced with the first argument. The second argument is then read by `\@MakeShortVerb`.

```

22 \newcommand*\MakeSpecialShortVerb[1]{%
23   \@ifstar
24     {\def\@shortvrbdef{#1*}\@MakeShortVerb}%
25     {\def\@shortvrbdef{#1}\@MakeShortVerb}%
26 }

```