The makerobust package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2006/03/18 v1.0

Abstract

Package makerobust provides \MakeRobustCommand that converts an existing macro to a robust one.

Contents

1 User interface

LaTeX offers \DeclareRobustCommand to define a robust macro that does not break if it is used in moving arguments. Sometimes a macro is already defined, but not robust. For example, \(and \) are not robust, inside \section the user must use \protect explicitly. This could be avoided by making \(and \) robust.

\MakeRobustCommand{⟨*cmd*⟩}

\MakeRobustCommand redefines the macro ⟨*cmd*⟩ by using \DeclareRobustCommand and the existing definition of the macro ⟨*cmd*⟩.

- It is an error if ⟨*cmd*⟩ is undefined. If you want to define a robust command, then you can use \DeclareRobustCommand directly.
- If the macro has previously been defined by \DeclareRobustCommand then the redefinition of \MakeRobustCommand is omitted, because the macro is already robust. Only an information entry is written to the .log file. Thus you do not get a warning or an error if the macro is already robust because of an updated LaTeX or package that defines the macro.
- Two macros are defined for a macro, defined by \DeclareRobustCommand. Example:

```
\DeclareRobustCommand{\foobar}{definition text}
```

Then the macro "`\foobar`" contains the protection code and, depending on the protection mode, calls the internal macro "`\foobar `". Notice the space at the end of the macro name. This internal macro "`\foobar `" now contains the definition "`definition text`", given in `\DeclareRobustCommand`.

Sometimes it can happen, that the internal macro already exists. This can be caused by a previous `\DeclareRobustCommand` followed by `\renewcommand`. Then the redefinition by `\MakeRobustCommand` would be safe.

However, it can also be possible that the macro is already robust, using the internal macro, but with a different protection code. The redefinition by `\MakeRobustCommand` would then generate an infinite loop.

Therefore `\MakeRobustCommand` raises an error message, if the internal macro (with space at the end) already exists.

1.1 Example

```
⟨*example⟩
\documentclass{article}
\usepackage{makerobust}
\MakeRobustCommand\(
\MakeRobustCommand\)
\pagestyle{headings}
\begin{document}
\tableofcontents
\section{Einstein: \(E=mc^2\)}
\newpage
Second page.
\end{document}
⟨/example⟩
```

2 Implementation

```
⟨*package⟩
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{makerobust}%
  [2006/03/18 v1.0 Making a macro robust (HO)]
\def\MakeRobustCommand#1{%
  \begingroup
  \@ifundefined{\expandafter\@gobble\string#1}{%
    \endgroup
    \PackageError{makerobust}{%
      Macro \string`\string#1\string' is not defined%
    }\@ehc
  }{%
    \global\let\MR@gtemp#1%
    \let#1\@undefined
    \expandafter\let\expandafter\MR@temp
        \csname\expandafter\@gobble\string#1 \endcsname
    \DeclareRobustCommand#1{}%
    \ifx#1\MR@gtemp
      \endgroup
      \PackageInfo{makerobust}{%
        \string`\string#1\string' is already robust%
      }%
    \else
      \@ifundefined{MR@temp}{%
        \global\let\MR@gtemp#1%
        \endgroup
        \expandafter\let\csname\expandafter\@gobble\string#1 \endcsname#1%
        \let#1\MR@gtemp
      }{%
```

```
        \endgroup
        \PackageError{makerobust}{%
          Internal macro \string`\string#1 \string' already exists%
        }\@ehc
      }%
    \fi
  }%
}
```

⟨/package⟩

3 Installation

CTAN. This package is available on CTAN[1]:

CTAN:macros/latex/contrib/oberdiek/makerobust.dtx The source file.

CTAN:macros/latex/contrib/oberdiek/makerobust.pdf Documentation.

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain-TEX:

```
tex makerobust.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
makerobust.sty          →  tex/latex/oberdiek/makerobust.sty
makerobust.pdf          →  doc/latex/oberdiek/makerobust.pdf
makerobust-example.tex  →  doc/latex/oberdiek/makerobust-example.tex
makerobust.dtx          →  source/latex/oberdiek/makerobust.dtx
```

If you have a `docstrip.cfg` that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

Refresh file databases. If your TEX distribution (teTEX, mikTEX, ...) rely on file databases, you must refresh these. For example, teTEX users run `texhash` or `mktexlsr`.

3.1 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is pdftk, e.g. unpack the file into the current directory:

```
pdftk makerobust.pdf unpack_files output .
```

Unpacking with LATEX. The `.dtx` chooses its action depending on the format:

plain-TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intension:

```
latex \install=y\input{makerobust.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

[1] ftp://ftp.ctan.org/tex-archive/

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLaTeX:

```
pdflatex makerobust.dtx
makeindex -s gind.ist makerobust.idx
pdflatex makerobust.dtx
makeindex -s gind.ist makerobust.idx
pdflatex makerobust.dtx
```

4 History

[2006/03/18 v1.0]

- First version.

5 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.