The letltxmacro package

Heiko Oberdiek*

2019/12/03 v1.6

Abstract

TEX's `\let` assignment does not work for LATEX macros with optional arguments or for macros that are defined as robust macros by `\DeclareRobustCommand`. This package defines `\LetLtxMacro` that also takes care of the involved internal macros.

Contents

*Please report any issues at https://github.com/ho-tex/letltxmacro/issues

1 Documentation

If someone wants to redefine a macro with using the old meaning, then one method is TEX's command `\let`:

```
\newcommand{\Macro}{\typeout{Test Macro}}
\let\SavedMacro=\Macro
\renewcommand{\Macro}{%
  \typeout{Begin}%
  \SavedMacro
  \typeout{End}%
}
```

However, this method fails, if `\Macro` is defined by `\DeclareRobustCommand` and/or has an optional argument. In both cases LATEX defines an additional internal macro that is forgotten in the simple `\let` assignment of the example above.

`\LetLtxMacro` {⟨*new macro*⟩} {⟨*old macro*⟩}

Macro `\LetLtxMacro` behaves similar to TEX's `\let` assignment, but it takes care of macros that are defined by `\DeclareRobustCommand` and/or have optional arguments. Example:

```
\DeclareRobustCommand{\Macro}[1][default]{...}
\LetLtxMacro{\SavedMacro}{\Macro}
```

Then macro `\SavedMacro` only uses internal macro names that are derived from `\SavedMacro`'s macro name. Macro `\Macro` can now be redefined without affecting `\SavedMacro`.

`\GlobalLetLtxMacro` {⟨*new macro*⟩} {⟨*old macro*⟩}

Like `\LetLtxMacro`, but the ⟨*new macro*⟩ is defined globally. Since version 2019/12/03 v1.4.

1.1 Supported macro definition commands

`\newcommand`, `\renewcommand`	latex/base
`\newenvironment`, `\renewenvironment`	latex/base
`\DeclareRobustCommand`	latex/base
`\newrobustcmd`, `\renewrobustcmd`	etoolbox
`\robustify`	etoolbox 2008/06/22 v1.6

2 Implementation

2.1 Show cases

2.1.1 `letltxmacro-showcases.tex`

```
⟨*showcases⟩
\NeedsTeXFormat{LaTeX2e}
\makeatletter
```

`\Line` The result is displayed by macro `\Line`. The percent symbol at line start allows easy grepping and inserting into the DTX file.

```
\newcommand*{\Line}[1]{%
  \typeout{\@percentchar#1}%
}

\newcommand*{\ShowCmdName}[1]{%
  \@ifundefined{#1}{}{%
    \Line{%
      \space\space(\expandafter\string\csname#1\endcsname) = %
      (\expandafter\meaning\csname#1\endcsname)%
    }%
  }%
}
\newcommand*{\ShowCmds}[1]{%
  \ShowCmdName{#1}%
  \ShowCmdName{#1 }%
  \ShowCmdName{\\#1}%
  \ShowCmdName{\\#1 }%
}
\let\\\@backslashchar
```

\ShowDef

```
\newcommand*{\ShowDef}[2]{%
  \begingroup
    \Line{}%
    \newcommand*{\DefString}{#2}%
    \@onelevel@sanitize\DefString
    \Line{\DefString}%
    #2%
    \ShowCmds{#1}%
  \endgroup
}

\typeout{}
\Line{* LaTeX definitions:}
\ShowDef{cmd}{%
  \newcommand{\cmd}[2][default]{}%
}
\ShowDef{cmd}{%
  \DeclareRobustCommand{\cmd}{}%
}
\ShowDef{cmd}{%
  \DeclareRobustCommand{\cmd}[2][default]{}%
}
\typeout{}
```

The minimal version of package etoolbox is 2008/06/12 v1.6a because it fixes `\robustify`.

```
\RequirePackage{etoolbox}[2008/06/12]%
\Line{}
\Line{* etoolbox's robust definitions:}
\ShowDef{cmd}{%
  \newrobustcmd{\cmd}{}%
}
\ShowDef{cmd}{%
  \newrobustcmd{\cmd}[2][default]{}%
}
\Line{}
\Line{* etoolbox's \string\robustify:}
\ShowDef{cmd}{%
```

```
  \newcommand{\cmd}[2][default]{} %
  \robustify{\cmd}%
}
\ShowDef{cmd}{%
  \DeclareRobustCommand{\cmd}{} %
  \robustify{\cmd}%
}
\ShowDef{cmd}{%
  \DeclareRobustCommand{\cmd}[2][default]{} %
  \robustify{\cmd}%
}
\typeout{}
\@@end
```
⟨/showcases⟩

2.1.2 Result

```
* LaTeX definitions:

\newcommand {\cmd }[2][default]{}
  (\cmd) = (macro:->\@protected@testopt \cmd \\cmd {default})
  (\\cmd) = (\long macro:[#1]#2->)

\DeclareRobustCommand {\cmd }{}
  (\cmd) = (macro:->\protect \cmd␣ )
  (\cmd␣) = (\long macro:->)

\DeclareRobustCommand {\cmd }[2][default]{}
  (\cmd) = (macro:->\protect \cmd␣ )
  (\cmd␣) = (macro:->\@protected@testopt \cmd␣ \\cmd␣ {default})
  (\\cmd␣) = (\long macro:[#1]#2->)

* etoolbox's robust definitions:

\newrobustcmd {\cmd }{}
  (\cmd) = (\protected\long macro:->)

\newrobustcmd {\cmd }[2][default]{}
  (\cmd) = (\protected macro:->\@testopt \\cmd {default})
  (\\cmd) = (\long macro:[#1]#2->)

* etoolbox's \robustify:

\newcommand {\cmd }[2][default]{} \robustify {\cmd }
  (\cmd) = (\protected macro:->\@protected@testopt \cmd \\cmd {default})
  (\\cmd) = (\long macro:[#1]#2->)

\DeclareRobustCommand {\cmd }{} \robustify {\cmd }
  (\cmd) = (\protected macro:->)

\DeclareRobustCommand {\cmd }[2][default]{} \robustify {\cmd }
  (\cmd) = (\protected macro:->\@protected@testopt \cmd␣ \\cmd␣ {default})
  (\cmd␣) = (macro:->\@protected@testopt \cmd␣ \\cmd␣ {default})
  (\\cmd␣) = (\long macro:[#1]#2->)
```

2.2 Package

⟨*package⟩

2.2.1 Catcodes and identification

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode123=1 % {
  \catcode125=2 % }
  \catcode64=11 % @
  \def\x{\endgroup
    \expandafter\edef\csname llm@AtEnd\endcsname{%
      \endlinechar=\the\endlinechar\relax
      \catcode13=\the\catcode13\relax
      \catcode32=\the\catcode32\relax
      \catcode35=\the\catcode35\relax
      \catcode61=\the\catcode61\relax
      \catcode64=\the\catcode64\relax
      \catcode123=\the\catcode123\relax
      \catcode125=\the\catcode125\relax
    }%
  }%
\x\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode64=11 % @
\catcode123=1 % {
\catcode125=2 % }
\def\TMP@EnsureCode#1#2{%
  \edef\llm@AtEnd{%
    \llm@AtEnd
    \catcode#1=\the\catcode#1\relax
  }%
  \catcode#1=#2\relax
}
\TMP@EnsureCode{40}{12}% (
\TMP@EnsureCode{41}{12}% )
\TMP@EnsureCode{42}{12}% *
\TMP@EnsureCode{45}{12}% -
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{58}{12}% :
\TMP@EnsureCode{62}{12}% >
\TMP@EnsureCode{91}{12}% [
\TMP@EnsureCode{93}{12}% ]
\edef\llm@AtEnd{%
  \llm@AtEnd
  \escapechar\the\escapechar\relax
  \noexpand\endinput
}
\escapechar=92 % `\\
```

Package identification.

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{letltxmacro}%
  [2019/12/03 v1.6 Let assignment for LaTeX macros (HO)]
```

2.2.2 Main macros

\LetLtxMacro

```
\newcommand*{\LetLtxMacro}{%
```

```
  \llm@ModeLetLtxMacro{}%
}
```

\GlobalLetLtxMacro

```
\newcommand*{\GlobalLetLtxMacro}{%
  \llm@ModeLetLtxMacro\global
}
```

\llm@ModeLetLtxMacro

```
\newcommand*{\llm@ModeLetLtxMacro}[3]{%
  \edef\llm@escapechar{\the\escapechar}%
  \escapechar=-1 %
  \edef\reserved@a{%
    \noexpand\protect
    \expandafter\noexpand
    \csname\string#3 \endcsname
  }%
  \ifx\reserved@a#3\relax
    #1\edef#2{%
      \noexpand\protect
      \expandafter\noexpand
      \csname\string#2 \endcsname
    }%
    #1\expandafter\let
    \csname\string#2 \expandafter\endcsname
    \csname\string#3 \endcsname
    \expandafter\llm@LetLtxMacro
        \csname\string#2 \expandafter\endcsname
        \csname\string#3 \endcsname{#1}%
  \else
    \llm@LetLtxMacro{#2}{#3}{#1}%
  \fi
  \escapechar=\llm@escapechar\relax
}
```

\llm@LetLtxMacro

```
\def\llm@LetLtxMacro#1#2#3{%
  \escapechar=92 %
  \expandafter\llm@CheckParams\meaning#2:->\@nil{%
    \begingroup
      \def\@protected@testopt{%
        \expandafter\@testopt\@gobble
      }%
      \def\@testopt##1##2{%
        \toks@={##2}%
      }%
      \let\llm@testopt\@empty
      \edef\x{%
        \noexpand\@protected@testopt
        \noexpand#2%
        \expandafter\noexpand\csname\string#2\endcsname
      }%
      \expandafter\expandafter\expandafter\def
      \expandafter\expandafter\expandafter\y
      \expandafter\expandafter\expandafter{%
        \expandafter\llm@CarThree#2{}{}{}\llm@nil
      }%
      \ifx\x\y
```

```
        #2%
        \def\llm@testopt{%
          \noexpand\@protected@testopt
          \noexpand#1%
        }%
      \else
        \edef\x{%
          \noexpand\@testopt
          \expandafter\noexpand
          \csname\string#2\endcsname
        }%
        \expandafter\expandafter\expandafter\def
        \expandafter\expandafter\expandafter\y
        \expandafter\expandafter\expandafter{%
          \expandafter\llm@CarTwo#2{}{}\llm@nil
        }%
        \ifx\x\y
          #2%
          \def\llm@testopt{%
            \noexpand\@testopt
          }%
        \fi
      \fi
      \ifx\llm@testopt\@empty
      \else
        \llm@protected\xdef\llm@GlobalTemp{%
          \llm@testopt
          \expandafter\noexpand
          \csname\string#1\endcsname
          {\the\toks@}%
        }%
      \fi
    \expandafter\endgroup\ifx\llm@testopt\@empty
      #3\let#1=#2\relax
    \else
      #3\let#1=\llm@GlobalTemp
      #3\expandafter\let
          \csname\string#1\expandafter\endcsname
          \csname\string#2\endcsname
    \fi
  }{%
    #3\let#1=#2\relax
  }%
}
```

\llm@CheckParams

```
\def\llm@CheckParams#1:->#2\@nil{%
  \begingroup
    \def\x{#1}%
  \ifx\x\llm@macro
    \endgroup
    \def\llm@protected{}%
    \expandafter\@firstoftwo
  \else
    \ifx\x\llm@protectedmacro
      \endgroup
      \def\llm@protected{\protected}%
      \expandafter\expandafter\expandafter\@firstoftwo
```

```
231     \else
232       \endgroup
233       \expandafter\expandafter\expandafter\@secondoftwo
234     \fi
235   \fi
236 }
```

`\llm@macro`

```
237 \def\llm@macro{macro}
238 \@onelevel@sanitize\llm@macro
```

`\llm@protectedmacro`

```
239 \def\llm@protectedmacro{\protected macro}
240 \@onelevel@sanitize\llm@protectedmacro
```

`\llm@CarThree`

```
241 \def\llm@CarThree#1#2#3#4\llm@nil{#1#2#3}%
```

`\llm@CarTwo`

```
242 \def\llm@CarTwo#1#2#3\llm@nil{#1#2}%

243 \llm@AtEnd%
244 ⟨/package⟩
```

3 Installation

3.1 Download

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

CTAN:macros/latex/contrib/letltxmacro/letltxmacro.dtx The source file.

CTAN:macros/latex/contrib/letltxmacro/letltxmacro.pdf Documentation.

Bundle. All the packages of the bundle 'letltxmacro' are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

CTAN:install/macros/latex/contrib/letltxmacro.tds.zip

TDS refers to the standard "A Directory Structure for TEX Files" (CTAN:pkg/tds). Directories with `texmf` in their name are usually organized this way.

3.2 Bundle installation

Unpacking. Unpack the `letltxmacro.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip letltxmacro.tds.zip -d ~/texmf
```

3.3 Package installation

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

```
tex letltxmacro.dtx
```

[1] CTAN:pkg/letltxmacro

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

```
letltxmacro.sty           → tex/latex/letltxmacro/letltxmacro.sty
letltxmacro.pdf           → doc/latex/letltxmacro/letltxmacro.pdf
letltxmacro-showcases.tex → doc/latex/letltxmacro/letltxmacro-showcases.tex
letltxmacro.dtx           → source/latex/letltxmacro/letltxmacro.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.

3.4 Refresh file name databases

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

3.5 Some details for the interested

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 intention:

```
latex \let\install=y\input{letltxmacro.dtx}
```

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

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

4 History

[2008/06/09 v1.0]

- First version.

[2008/06/12 v1.1]

- Support for etoolbox's `\newrobustcmd` added.

[2008/06/13 v1.2]

- Support for etoolbox's `\robustify` added.

[2008/06/24 v1.3]

- Test file adapted for etoolbox 2008/06/22 v1.6.

[2010/09/02 v1.4]

- `\GlobalLetLtxMacro` added.

[2016/05/16 v1.5]

- Documentation updates.

[2019/12/03 v1.6]

- Documentation updates.

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; plain numbers refer to the code lines where the entry is used.