The \eolgrab package

Heiko Oberdiek*

2016/05/16 v1.1

Abstract

This package implements a generic argument grabber to catch an argument that is delimited by the line end.

Contents

1 Documentation .................................................. 1
  1.1 Examples .................................................. 2
    1.1.1 Small \LaTeX{} document as example .................. 3
    1.1.2 \LaTeX{} document with environments ............... 4
  1.2 Limitations ............................................... 4

2 Implementation ................................................ 4
  2.1 Reload check and package identification .................. 4
  2.2 Catcodes .................................................. 6
  2.3 Resources ............................................... 6
  2.4 Macro \eolgrab ............................................. 7

3 Installation ................................................... 8
  3.1 Download ............................................... 8
  3.2 Bundle installation ..................................... 8
  3.3 Package installation ..................................... 8
  3.4 Refresh file name databases .............................. 9
  3.5 Some details for the interested ........................ 9

4 References .................................................... 9

5 History ........................................................ 9
  [2011/01/12 v1.0] ............................................. 9
  [2016/05/16 v1.1] ............................................. 9

6 Index ................................................................ 10

1 Documentation

The starting point for this package was a feature request of Arno Trautmann in the mailing list texhax\footnote{[1]}\cite{texhax}. A macro \eolsection should behave like \section, but the argument should be delimited by the line end instead of given in curly braces:

\begin{quote}
\eolsection My Title
\end{quote}

\footnote{Please report any issues at \url{https://github.com/ho-tex/oberdiek/issues}}

\footnote{Info page for mailing list texhax: \url{https://tug.org/mailman/listinfo/texhax}}
Phil Taylor answered this with an implementation for \eolsection. Because this feature could be useful for other macros as well, I answered with an implementation of \eolgrab as general solution [3].

Both formats plain TeX and \LaTeX are supported by the package, see the example for \eolsection below.

\begin{verbatim}
\eolgrab \{(code)\} \{argument\} \{EOL\}
\end{verbatim}

Macro \eolgrab takes two arguments. The first argument is \(\text{(code)}\), a classical undelimited TeX macro argument. The second argument is delimited by the line end \(\text{(EOL)}\). The macro calls \(\text{(code)}\) with \(\text{(argument)}\) as argument in curly braces. Because the catcode of the line end is changed, \eolgrab will not work in the argument of other macros. Macro \eolgrab is made robust if either \varepsilon-TeX's \texttt{\textbackslash \protected} or \LaTeX's \texttt{\textbackslash \DeclareRobustCommand} is available.

\begin{verbatim}
\eolgrabopt \{(code)\} \{argument\} \{EOL\}
\end{verbatim}

Macro \eolgrabopt passes \(\text{(argument)}\) as optional argument to \(\text{(code)}\) if \(\text{(argument)}\) is not empty.

\eolgrabopt\item foo
becomes to
\begin{verbatim}
\item[{foo}]
\end{verbatim}

The curly argument braces are added to support square brackets inside \(\text{(argument)}\). If the \(\text{(argument)}\) is empty:

\begin{verbatim}
\eolgrabopt\item
\end{verbatim}

then

\begin{verbatim}
\item
\end{verbatim}

is called without optional argument.

1.1 Examples

- The line

\begin{verbatim}
\eolgrab\section My Title
\end{verbatim}

is equivalent to

\begin{verbatim}
\section{My Title}
\end{verbatim}

- The next example uses the star form of \texttt{\section}. Then the command to be called consists of two tokens. Therefore the first argument of \eolgrab needs curly braces:

\begin{verbatim}
\eolgrab{\section*}My Title
\end{verbatim}

becomes

\begin{verbatim}
\section*{My Title}
\end{verbatim}

- Now \LaTeX’s \texttt{\PackageError} is used. This macro has three arguments, the package or class name, the message text and the help text. A standard help text of \LaTeX is used as given in macro \texttt{\@ehc}. The second argument, the message text is used as argument, delimited by line end:
Some error message text with several lines

In the first two lines of the example, the line end is suppressed by the comment character (percent), thus the argument is delimited by the line end of the third line. The result is:

\PackageError{foobar}{Some error message text\MessageBreak with several lines}\@ehc

- The original request for macro \eolsection, see above, can be implemented easily with the help of \eolgrab. Example for \LaTeX:

\usepackage{eolgrab}
\newcommand*{\eolsection}{\eolgrab\section}

Example for plain \TeX:

\input eolgrab.sty\relax
\def\eolsection{\eolgrab\section}

And a sophisticated variant for \LaTeX that also supports the star syntax and the optional argument:

\begin{document}
\tableofcontents
\eolsection Section without star and optional argument
\eolsection*{Section with star}
\eolsection[Short section title]Long section title
\end{document}

1.1.1 Small \LaTeX document as example

\RequirePackage{eolgrab}
eolgrab\documentclass{article}
eolgrab\begin{document}
eolgrab\section{Hello World}
eolgrab\emph{Some text}
eolgrab\end{document}
1.1.2 \LaTeX{} document with environments

\begin{example-env}
\documentclass{article}
\usepackage{eolgrab}
\newcommand*{\Begin}{\eolgrab \begin}
\newcommand*{\End}{\eolgrab \end}
\newcommand*{\Item}{\eolgrabopt \item}
\Begin document
\Begin itemize
\Item first item
\Item second item
\End itemize
\Begin description
\Item foo is the first syllable of foobar.
\Item bar is the second syllable of foobar.
\End description
\End document
\end{example-env}

1.2 Limitations

Macro \eolgrab needs to catch the line end. If \TeX{} reads a line, then it throws away the line end characters (carriage return, line feed) and removes spaces at the end of the line. Then it adds the character with the character code that is given by \endlinechar at the end of the line. The category code of the inserted character is given by the current value of its \catcode. If \endlinechar is not a valid character code (especially if it is negative), then no character is added.

In plain \TeX{} and \LaTeX{} the standard settings of the inserted endline character is the character with code 13 (or \texttt{^^M} in \TeX{} notation) with catcode 5 (end of line). That means the inserted end of line character behaves like a space token. For example, it is removed after macro names. Therefore \eolgrab changes the catcode.

Therefore \eolgrab has some limitations:

- Like other verbatim stuff, the macro \eolgrab cannot be used in the argument of other macros. \eolgrab want to change the catcode of the end of line character. If this character is read before, because it is processed as argument of another macro, the catcode is already set and is not reassigned later if \eolgrab changes the category code for this character code.

- The argument must not contain the end of line character. Otherwise the first end of line character is already taken as delimiter, leaving the rest of the line outside the argument.

- Because \eolgrab is probably mostly used in the line with the delimited argument. Therefore changes of \endlinechar will not affect the current line.

2 Implementation

\begin{package}
\begin{implementation}

2.1 Reload check and package identification

Reload check, especially if the package is not used with \LaTeX{}.
\begin{verbatim}
\begingroup\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 \^\texttt{M}
\end{verbatim}
\end{implementation}
\end{package}
\ProvidesPackage{eolgrab} \[2016/05/16 v1.1 Catch arguments delimited by end of line (HO)]

\section*{2.2 Catcodes}

\begingroup\catcode61\catcode48\catcode32=10\relax\def\x{\endgroup\catcode61\catcode48\catcode32=10\relax\catcode13=5 \hfill
\endlinechar=13 \hfill
\catcode35=6 \hfill
\catcode64=11 \hfill
\catcode123=1 \hfill
\catcode125=2 \hfill
\def\TMP@EnsureCode#1#2{\def\eolgrab@AtEnd{\eolgrab@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{46}{12}.\TMP@EnsureCode{47}{12}/\TMP@EnsureCode{91}{12}\TMP@EnsureCode{93}{12}\TMP@EnsureCode{94}{7}\edef\eolgrab@AtEnd\noexpand\endinput\endgroup

\section*{2.3 Resources}

\begingroup\expandafter\expandafter\expandafter\endgroup\expandafter\ifx\csname RequirePackage\endcsname\relax\input ltxcmds.sty\relax\input infwarerr.sty\relax\else\RequirePackage{ltxcmds}\[2010/12/04]\RequirePackage{infwarerr}\[2010/04/08]\fi
\eolgrab@ifdefined
\ltx@ifundefined{@ifdefinable}{\def\eolgrab@ifdefinable#1#2{\ltx@ifundefined{#1}{#2}{}}}
\def\eolgrab@ifdefinable#1{\ltx@ifundefined{#1}{\edef\eolgrab@ifdefinable{\eolgrab@ifdefinable\noexpand\endinput}}}

2.4 Macro \eolgrab

\eolgrab
\eolgrab@ifdefinable{eolgrab}{%
\ltx@IfUndefined{protected}{%
\ltx@IfUndefined{DeclareRobustCommand}{%
\def\eolgrab#1%{%
\newcommand\eolgrab{}%
\DeclareRobustCommand*\eolgrab%{%
}%
\}}%
\protected\def\eolgrab#1%{%
}%
\begingroup
\endlinechar=13 %
\catcode13=\ltx@active
\eolgrab@{#1}%
\begingroup
\catcode13=\ltx@active %
\ltx@firstofone{\endgroup %
\eolgrab@
\def\eolgrab@#1#2^^M{%
\endgroup %
#1{#2}%
\eolgrab@opt}

\eolgrab@opt
\eolgrab@ifdefinable{eolgrabopt}{%
\ltx@IfUndefined{protected}{%
\ltx@IfUndefined{DeclareRobustCommand}{%
\def\eolgrabopt#1%{%
\newcommand\eolgrabopt{}%
\DeclareRobustCommand*\eolgrabopt%{%
}%
\}}%
\protected\def\eolgrabopt#1%{%
}%
\begingroup
\endlinechar=13 %
\catcode13=\ltx@active
\eolgrab@opt{#1}%
\begingroup
\catcode13=\ltx@active %
\ltx@firstofone{\endgroup %
\eolgrab@#}
\def\eolgrab@#1#2~~M{%
\endgroup %
#1(#2)%
\eolgrab@opt
3 Installation

3.1 Download

Package. This package is available on CTAN:\(^2\):


Bundle. All the packages of the bundle ‘oberdiek’ 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/oberdiek.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 oberdiek.tds.zip in the TDS tree (also known as texmf tree) of your choice. Example (linux):

```
unzip oberdiek.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 eolgrab.dtx
```

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

<table>
<thead>
<tr>
<th>File</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>eolgrab.sty</td>
<td>→ tex/generic/oberdiek/eolgrab.sty</td>
</tr>
<tr>
<td>eolgrab.pdf</td>
<td>→ doc/lates/oberdiek/eolgrab.pdf</td>
</tr>
<tr>
<td>example/eolgrab-example-ltx.tex</td>
<td>→ doc/lates/oberdiek/example/eolgrab-example-ltx.tex</td>
</tr>
<tr>
<td>example/eolgrab-example-env.tex</td>
<td>→ doc/lates/oberdiek/example/eolgrab-example-env.tex</td>
</tr>
<tr>
<td>example/eolgrab-example-sec.tex</td>
<td>→ doc/lates/oberdiek/example/eolgrab-example-sec.tex</td>
</tr>
<tr>
<td>eolgrab.dtx</td>
<td>→ source/lates/oberdiek/eolgrab.dtx</td>
</tr>
</tbody>
</table>

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.

\(^2\)CTAN:pkg/eolgrab
3.4 Refresh file name databases
If your \TeX{} distribution (\TeX{} Live, MiK\TeX{}, \ldots) relies on file name databases, you must refresh these. For example, \TeX{} Live users run \texttt{texhash} or \texttt{mktexlsr}.

3.5 Some details for the interested

Unpacking with \LaTeX{}. The \texttt{.dtx} chooses its action depending on the format:

plain \TeX{}: Run \texttt{docstrip} and extract the files.

\LaTeX{}: Generate the documentation.

If you insist on using \LaTeX{} for \texttt{docstrip} (really, \texttt{docstrip} does not need \LaTeX{}), then inform the autodetect routine about your intention:

\begin{verbatim}
latex \let\install=y\input{eolgrab.dtx}
\end{verbatim}

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

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

\begin{verbatim}
\PassOptionsToClass{a4paper}{article}
\end{verbatim}

An example follows how to generate the documentation with pdf\LaTeX{}:

\begin{verbatim}
pdflatex eolgrab.dtx
makeindex -s gind.ist eolgrab.idx
pdflatex eolgrab.dtx
makeindex -s gind.ist eolgrab.idx
pdflatex eolgrab.dtx
\end{verbatim}

4 References


5 History

[2011/01/12 v1.0]
\begin{itemize}
\item First public version.
\end{itemize}

[2016/05/16 v1.1]
\begin{itemize}
\item Documentation updates.
6 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.

Symbols
\@PackageError .......................... 176
@ehc ..................................... 178
@ifdefinable .............................. 183
@ifnextchar ................................. 9
@ifstar .................................... 6
@undefined ................................ 113

A
\aftergroup ................................ 84

B
\Begin ....................................... 38, 41, 42, 48
\begin ....................................... 20, 30, 38

C
\catcode 57, 58, 60, 61, 62, 63, 64, 65, 66, 67, 68, 88, 89, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 124, 125, 127, 128, 129, 133, 134, 135, 136, 137, 138, 139, 142, 143, 145, 146, 147, 148, 152, 154, 199, 215, 216
\csname .................................. 69, 76, 105, 121, 166, 183

D
\DeclareRobustCommand .................. 192, 209
\documentclass ......................... 2, 29, 36

E
\emph ....................................... 32
\empty ...................................... 72, 73
\End ....................................... 39, 47, 53, 54
\end ......................................... 25, 33, 39
\endcsname 69, 76, 105, 121, 131, 166, 183
\endinput .................................. 84, 164
\endlinechar ............................... 59, 90, 126, 132, 144, 198, 215
\eolgrab .................................. 2, 7, 12, 17, 29, 30, 31, 32, 33, 38, 39, 186
\eolgrab@ 200, 224
\eolgrab@AtEnd 150, 151, 164, 236
\eolgrab@ifdefinable 173, 186, 203
\eolgrab@opt 217, 227
\eolgrabopt 2, 40, 203
\eoloptsection 10, 16
\eolsection 5, 22, 23, 24

I
\ifx ........................................ 70, 73, 76, 105, 113, 116, 167
\immediate ................................ 78, 107
\input ...................................... 167, 168

L
\ltx@active ............................... 199, 216, 221
\ltx@backslashchar ........................ 177
\ltx@firstofone ........................... 177
\ltx@ifempty ................................ 188, 205
\ltx@IfUndefined 173, 187, 188, 201, 205
\ltx@ifundefined .......................... 175

M
\makeatletter ................................ 4
\makeatother ................................ 19

N
\newcommand 5, 16, 38, 39, 40, 191, 208

P
\PackageInfo ................................ 81
\protected .................................. 195, 212
\ProvidesPackage .......................... 74, 122

R
\RequirePackage ............................ 28, 170, 171

S
\section .................................... 7, 12, 17, 31

T
\tableofcontents ............................ 21
\the ....................................... 132, 133, 134, 135, 136, 137, 138, 139, 152
\TMP@EnsureCode 149, 156, 157, 158, 159, 160, 161, 162, 163

U
\usepackage .................................. 3, 37

W
\write ...................................... 78, 107

X
\x 69, 70, 73, 77, 81, 83, 106, 111, 121, 130, 142