

The texdef script

Martin Scharrer
martin@scharrer-online.de

CTAN: <http://www.ctan.org/pkg/texdef>

VC: https://bitbucket.org/martin_scharrer/texdef

Version 1.8 – 2018/03/25

Abstract

This Perl script shows the definition of \LaTeX commands in a similar way as the \TeX primitive `\show`. It is intended to allow users to quickly see the definitions of user level or internal package macros as well as the values of registers. The given commands are compiled internally with \LaTeX and the output of `\show` is formatted and shown to the user.

Contents

1	Usage	1
2	Installation	4
3	Examples	5
4	Changelog	6

1 Usage

The script executable is called `texdef` (or maybe `texdef.pl`). A normal installation is supposed to also install it as `latexdef` (e.g. using a symlink to `texdef`) in order to allow for quick access to the \LaTeX format. Further format shortcuts are possible as explained in the installation section.

```
texdef [<options>] <commandname> [<commandname> ...]  
latexdef [<options>] <commandname> [<commandname> ...]
```

Other program names are possible. See the `tex` option. Command names do not need to start with `'\'`.

Options

The following options can be used:

`-tex <format>`, `-t <format>`

Use given format of TeX: 'tex', 'latex', 'context'. Variations of 'tex' and 'latex', like 'luatex', 'lualatex', 'xetex', 'xelatex' are supported. The default is given by the used program name: 'texdef' -> 'tex', 'latexdef' -> 'latex', etc.

-texoptions *<options>*

Call \TeX with the given options.

-source, -s

Try to show the original source code of the command definition (L).

-value, -v

Show value of command instead (i.e. \the\command).

-preamble, -P

Show definition of the command inside the preamble.

-Environment, -E

Every command name is taken as an environment name. This will show the definition of both \foo and \endfoo if foo is used as command name (L).

-beforeclass, -B

Show definition of the command before \documentclass .

-package *<pkg>*, -p *<pkg>*

(M) Load given tex-file, package or module depending on whether '*tex', '*latex' or 'context' is used. For LaTeX the *<pkg>* can start with '[<options>]' and end with *<pkgname>* or $\{\text{\textit{pkgname}}\}$.

-class *<class>*, -c *<class>*

(LaTeX only) Load given class instead of default ('article'). The *<class>* can start with [*<class options>*] and end with *<classname>* or $\{\text{\textit{classname}}\}$.

-environment *<env>*, -p *<env>*

(M) Show definition inside the given environment *<env>*.

-othercode *<code>*, -o *<code>*

(M) Add other code into the preamble before the definition is shown. This can be used to e.g. load PGF/TikZ libraries.

-before *<code>*, -b *<code>*

(M) Place *<code>* before definition is shown. The *<code>* can be arbitray TeX code and doesn't need be be balanced.

`-after <code>, -a <code>`

(M) Place `<code>` after definition is shown. The `<code>` can be arbitrary TeX code and doesn't need to be balanced.

`-find, -f`

Find file where the command sequence was defined (L).

`-Find, -F`

Show full filepath of the file where the command sequence was defined (L).

`-list, -l`

List user level command sequences of the given packages (L).

`-list-defs, -L`

List user level command sequences and their shorten definitions of the given packages (L).

`-list-all, -ll`

List all command sequences of the given packages (L).

`-list-defs-all, -LL`

List all command sequences and their shorten definitions of the given packages (L).

`-ignore-cmds <cs,cs,...>, -i`

Ignore the following command sequence(s) in the above lists. (M)

`-ignore-regex <regex,...>, -I`

Ignore all command sequences in the above lists which match the given Perl regular expression(s). (M)

`-pgf-keys, -k`

Takes commands as `pgfkeys` and displays their definitions. Keys must use the full path but the common `'cmd'` prefix is applied.

`-pgf-Keys, -K`

Takes commands as `pgfkeys` and displays their definitions. Keys must use the full path.

`-version, -V`

If used alone prints version of this script. (L) Together with -p or -c prints version of LaTeX package(s) or class, respectively.

-edit

Opens the file holding the macro definition. Uses --Find and --source. If the source definition can not be found the definition is printed as normal instead. (L)

-editor <editor>

Can be used to set the used editor. If not used the environment variables TEXDEF_EDITOR, EDITOR and SELECTED_EDITOR are read in this order. If none of these are set a list of default editors are tried. The <editor> string can include '%f' for the filename, '%n' for the line number and '%' for a literal '%'. If no '%' is used '+%n %f' is added to the given command.

-tempdir <directory>

Use given existing directory for temporary files.

-help, -h

Print this help and quit.

Notes:

Long option can be shorten as long the are still unique. Short options can be combined. If the option 'environment', 'before' and 'after' are used together the produced code will be inserted in the given order (reversed order for 'after').

(M) = This option can be given multiple times.

(L) = LaTeX only. Requires the packages 'filehook' and 'currfile'.

2 Installation

The latest official release of the script can be obtained from CTAN under <http://mirrors.ctan.org/support/texdef/texdef.pl>. The latest develop version (which can be unstable!) can be obtained from the source code repository under https://bitbucket.org/martin_scharrer/texdef/src/tip/texdef.pl.

For installation the script file texdef.pl must simply renamed to texdef and be copied to a location suitable for executables, i.e. a directory in the PATH, e.g. /usr/bin or /usr/local/bin for Linux/Unix. The script should be set as *executable*, i.e. `chmod +x texdef.pl` for Linux/Unix. This is not required for MS Windows.

This script can show the definitions of commands with different formats of TeX (tex, etex, latex, pdftex, pdfetex, pdflatex, xetex, xelatex, context, ...). The format can be given using an command line option or over the *script name*, i.e. if the script is called texdef it will use tex, but called latexdef it will use latex and so on. The script can be simply copied several times, but creating *symbolic links* to the real script is enough on operation systems/file systems which support this. At least the two variations texdef and latexdef should be installed.

Requirements

The `texdef` program is a Perl script which needs a Perl interpreter to work. Informations about installing Perl can be found at <http://www.perl.org/get.html>. The script calls a TeX compiler (`tex`, `latex`, ...) to display the definition of commands/macros. Therefore (L)TeX must also be installed. Informations about installing (L)TeX can be found at <http://www.latex-project.org/ftp.html>.

Quick Install Nodes for Linux/Unix

```
cp texdef.pl /usr/local/bin/  
cd /usr/local/bin  
ln -s texdef.pl texdef  
ln -s texdef.pl latexdef  
ln -s texdef.pl etexdef  
ln -s texdef.pl luatexdef  
ln -s texdef.pl lualatexdef  
ln -s texdef.pl pdftexdef  
ln -s texdef.pl pdfplatexdef  
ln -s texdef.pl xetexdef  
ln -s texdef.pl xelatexdef  
ln -s texdef.pl 'some other TeX format of your choice'
```

or use the Makefile (defaults to `/usr/local/bin`)

```
make install INSTALLDIR=/your/path
```

3 Examples

Show the definition of `\chapter` with different classes (article (default), book and scrbook):

```
latexdef chapter  
latexdef -c book chapter  
latexdef -c scrbook chapter
```

Show value of `\textwidth` with different class options:

```
latexdef -c [a4paper]{book} -v paperwidth  
latexdef -c [letter]{book} -v paperwidth
```

Show definition of TikZ's `\draw` outside and inside a `tikzpicture` environment:

```
latexdef -p tikz draw  
latexdef -p tikz --env tikzpicture draw
```

Show definition of TikZ's `\draw` inside a node, inside a beamer frame in `handout` mode:

```
latexdef -c [handout]beamer -p tikz --env frame \  
--env tikzpicture -b '\node {' -a '};' draw
```

List all user level command sequences (macros) defined by the `xspace` LaTeX package:

```
latexdef -l -p xspace
```

4 Changelog

v1.8 from 2018/03/25

- Added 'dvitex' and 'dvi_latex' as formats. These will set DVI mode on 'tex' or 'latex'.
- Escaped further braces in regexs to avoid warnings or errors.
- Added option '--texoptions' to allow the passing of compiler options for special cases.

v1.7c from 2017/12/09

- Fixed 'Unescaped left brace in regex is deprecated' warning. Thanks goes to Paulo Cereda <cereda.paulo@gmail.com> for providing the patch.

v1.7b from 2012/05/15

- Added support for `\chardef`, `\countdef`, `\dimendef`, `\mathchardef`, `\myskip`, `\skipdef` and `\toksdef`.
- Added support for `\newbox`, `\newcount`, `\newdimen`, `\newif`, `\newinsert`, `\newread`, `\newskip`, `\newtoks`, `\newwrite`.
- Added support for `\newif` and `\newcount`.
- Changed internal processing order to put more frequently used elements first.

v1.7a from 2012/05/08

- Fix for currfile v0.6.

v1.7 from 2012/05/07

- Changed '--help' to return status code 0.
- Changed Windows default editor to texworks. Fixed If-statement.
- Fixed bug which reported name of included subpackages.
- Added option '--edit'.
- Added '--editor' option and better editor handling.
- Added variable substitution for editor.
- Added support for 'TEXDEF_EDITOR' variable. Added Mac 'open' command.
- Fixed for behaviour when -s is used (correct name in message; do not follow protected macros etc.)

v1.6 from 2012/05/02

- Changed implementation of -E option to work better with new environment source code.
- Added code to detect environment source definitions.
- Added -E option.
- Changed 'flavour' to 'format'.
- Added support for `\let`.
- Added fallback for when the source code can not be found. Added support for -F with 'latex.ltx'.

v1.5 from 2012/04/29

- Added experimental `--source` option to show source code of the definitions.
- Script now works correctly under MS Windows.

v1.4 from 2011/07/28

- Added -F option to display file path.
- Added basic support for pdfkeys using -k and -K options.
- Added -V option to print version numbers of LaTeX packages or classes.
- Added '`--tempdir`' option to specify the location of the temporary files. Useful for debugging.