

ClutT_EX manual
(Version 0.7.1)

ARATA Mizuki

2026-10-31

Contents

1	About Clut_{TeX}	2
2	How to use Clut_{TeX}	3
2.1	Installation	3
2.2	Command-line usage	3
2.3	Sync _{TeX}	5
2.4	Watch mode	5
2.5	MakeIndex and Bib _{TeX}	5
2.6	For writing a large document	5
2.7	Using Makefile	6
2.8	Default output directory	6
2.9	Aliases	7
2.10	Support for <code>minted</code> and <code>epstopdf</code>	7
2.11	Check for driver file	7
2.12	Customization via a configuration file	8

Chapter 1

About Clut $\text{\TeX}$

Clut $\text{\TeX}$ is an automation tool for $\text{\LaTeX}$ document processing. Basic features are,

- Does not clutter your working directory with “extra” files, like `.aux` or `.log`.
- If multiple runs are required to generate correct document, do so.
- Watch input files, and re-process documents if changes are detected¹.
- Run `MakeIndex`, `BIB $\text{\TeX}$` , `Biber`, if requested.
- Produces a PDF, even if the engine (e.g. `p $\text{\TeX}$`) does not suport direct PDF generation. If you want a DVI file, use `--output-format=dvi` option.

The unique feature of this program is that, auxiliary files such as `.aux` or `.toc` are created in an isolated location, so you will not be annoyed with these extra files.

¹needs an external program if you are on a Unix system

Chapter 2

How to use ClutT_EX

2.1 Installation

If you are using the latest T_EX Live, you should have ClutT_EX installed. If not, upgrade your copy of T_EX Live with `tlmgr update --all`.

If you want to install ClutT_EX manually, fetch an archive from GitHub¹, extract it, and copy `bin/cluttex` or `bin/cluttex.bat` to somewhere in your PATH.

2.2 Command-line usage

Usage:

```
cluttex -e ENGINE OPTIONS [--] INPUT.tex
```

Basic options:

-e, --engine=ENGINE Set which T_EX engine/format to use. ENGINE is one of the following: `pdflatex`, `pdftex`, `lualatex`, `luatex`, `luajittex`, `xelatex`, `xetex`, `latex`, `etex`, `tex`, `platex`, `eptex`, `ptex`, `uplatex`, `euptex`, or `uptex`. Required.

-o, --output=FILE Set output file name. Default: `JOBNAME.FORMAT`

--fresh Clean auxiliary files before run. Cannot be used in conjunction with `--output-directory`.

--max-iterations=N Set maximum number of run, for resolving cross-references and etc. Default: 4

--watch[=ENGINE] Watch input files for change. May need an external program to be available. See section 2.4 for details.

--color[=WHEN] Colorize messages. WHEN is one of `always`, `auto`, or `never`. If `--color` option is omitted, `auto` is used. If WHEN is omitted, `always` is used.

¹<https://github.com/minoki/cluttex>

`--includeonly=NAMEs` Insert `\includeonly{NAMEs}`.
`--make-depends=FILE` Write Makefile-style dependencies information to `FILE`.
`--engine-executable=COMMAND` The actual $\TeX$ command to use.
`--tex-option=OPTION`, `--tex-options=OPTIONs` Pass extra options to $\TeX$.
`--dvipdfmx-option=OPTION`, `--dvipdfmx-options=OPTIONs` Pass extra options to `dvipdfmx`.
`--[no-]change-directory` Change to the output directory when run. May be useful with shell-escaping packages.
`-h`, `--help`
`-v`, `--version`
`-V`, `--verbose`
`--print-output-directory` Print the output directory and exit.
`--package-support=PKG1[,PKG2,...,PKGn]` Enable special support for shell-escaping packages. Currently, supported packages are ‘`minted`’, ‘`epstopdf`’ and ‘`pdfx`’.
`--check-driver=DRIVER` Check that the correct driver file is loaded for certain packages. `DRIVER` is one of `dvipdfmx`, `dvips`, or `dvisvgm`. Can only be used with `--output-format=dvi`.
`--source-date-epoch=TIME` Lock the creation time of PDF by setting the environment variable `SOURCE_DATE_EPOCH`. `TIME` is ‘`now`’ or an unsigned integer (typically, this value is interpreted as a Unix time).
`--config-file=FILE` Use the specified config file. See section 2.12.
Options for running auxiliary programs:
`--makeindex=COMMAND` Run `MakeIndex`.
`--bibtex=COMMAND` Run `BIB $\TeX$` .
`--biber[=COMMAND]` Run `Biber`. Default value for `COMMAND`: `biber`
`--makeglossaries[=COMMAND]` Run `makeglossaries`. Experimental.
 $\TeX$ -compatible options:
`--[no-]shell-escape`
`--shell-restricted`
`--synctex=NUMBER` Generate Sync $\TeX$ file. Note that `.synctex.gz` is created alongside the final `.pdf`. See section 2.3 for details.
`--[no-]file-line-error` Default: Yes
`--[no-]halt-on-error` Default: Yes

`--interaction=STRING` `STRING` is one of `batchmode`, `nonstopmode`, `scrollmode`, or `errorstopmode`. Default: `nonstopmode`

`--jobname=STRING`

`--fmt=FORMAT`

`--output-directory=DIR` Set output directory for `TEX` engine. Auxiliary files are produced in this directory. Default: Based on the `temporary-directory` configuration.

`--output-format=FORMAT` Set output format. Possible values are `pdf` or `dvi`. Default: `pdf`

Long options, except `TEX`-compatible ones, need two hyphens (e.g. `-synctex=1` is accepted, but not `--color`). Combining multiple short options, like `-Ve pdflatex`, is not supported.

2.3 SyncT_EX

You can generate SyncT_EX data with `--synctex=1` option.

Although ClutT_EX has “Don’t clutter your working directory” as its motto, the `.synctex.gz` file is always produced alongside the PDF file. This is because SyncT_EX cannot find its data file if it’s not in the same directory as the PDF.

2.4 Watch mode

If `--watch` option is given, ClutT_EX enters *watch mode* after processing the document.

On Windows, a built-in filesystem watcher is implemented. On other platforms, an auxiliary program `fswatch`² or `inotifywait` needs to be installed. The auxiliary program will be detected automatically, but it could also be specified by the `ENGINE` argument.

2.5 MakeIndex and BibT_EX

If you want to generate index or bibliography, using MakeIndex or BibT_EX, set `--makeindex`, `--bibtex`, or `--biber` option. You need to explicitly specify the command name as an argument (e.g. `--makeindex=makeindex`, `--bibtex=bibtex`).

If you want to use Biber to process bibliography, the option to use is `--biber`, not `--bibtex=biber`.

2.6 For writing a large document

When writing a large document with L^AT_EX, you usually split the T_EX files with `\include` command. When doing so, `\includeonly` can be used to eliminate processing time. But writing `\includeonly` in the T_EX source file is somewhat

²<http://emcrisostomo.github.io/fswatch/>

inconvenient. After all, `\includeonly` is about *how* to process the document, not about its content.

Therefore, Clut_{TEX} provides an command-line option to use `\includeonly`. See section 2.7 for example.

Tips: When using `includeonly`, avoid using `--makeindex` or `--biber`.

Another technique for eliminating time is, setting `--max-iterations=1`. It stops Clut_{TEX} from processing the document multiple times, which may take several extra minutes.

2.7 Using Makefile

You can create Makefile to avoid writing Clut_{TEX} options each time. Example:

```
main.pdf: main.tex chap1.tex chap2.tex
    cluttex -e lualatex -o $@ --makeindex=mendex $<

main-preview.pdf: main.tex chap1.tex chap2.tex
    cluttex -e lualatex -o $@ --makeindex=mendex --max-iterations=1 $<

chap1-preview.pdf: main.tex chap1.tex
    cluttex -e lualatex -o $@ --max-iterations=1 --includeonly=chap1 $<

chap2-preview.pdf: main.tex chap2.tex
    cluttex -e lualatex -o $@ --max-iterations=1 --includeonly=chap2 $<
```

With `--make-depends` option, you can let Clut_{TEX} infer sub-files and omit them from Makefile. Example:

```
main.pdf: main.tex
    cluttex -e lualatex -o $@ --make-depends=main.pdf.dep $<

-include main.pdf.dep
```

After initial `make` run, `main.pdf.dep` will contain something like this:

```
main.pdf: ... main.tex ... chap1.tex chap2.tex
```

Note that `--make-depends` option is still experimental, and may not work well with other options like `--makeindex`.

2.8 Default output directory

The auxiliary files like `.aux` are generated somewhere in the temporary directory, by default. The directory name depends on the following three parameters:

- The absolute path of the input file
- `--jobname` option
- `--engine` option

On the other hand, the following parameters doesn't affect the directory name:

- `--includeonly`
- `--makeindex`, `--bibtex`, `--biber`, `--makeglossaries`

If you need to know the exact location of the automatically-generated output directory, you can invoke Clut_{TEX} with `--print-output-directory`. For example, `clean` target of your Makefile could be written as:

```
clean:
    -rm -rf $(shell cluttex -e pdflatex --print-output-directory main.tex)
```

Clut_{TEX} itself doesn't erase the auxiliary files, unless `--fresh` option is set. Note that, the use of a temporary directory means, the auxiliary files may be cleared when the computer is rebooted.

2.9 Aliases

Some Unix commands change its behavior when it is called under a different name. There are several examples in _{TEX} Live:

- `extractbb` and `dvipdfmx` are aliases for `xdvipdfmx`.
- `repstopdf` is an alias for `epstopdf`.

If Clut_{TEX} is called as `cl(ENGINE)`, the `--engine` option is set accordingly. For example, `cllualatex` is an alias for `cluttex --engine lualatex` and `clxelatex` for `cluttex --engine xelatex`.

2.10 Support for minted and epstopdf

In general, packages that execute external commands (shell-escape) don't work well with `-output-directory`. Therefore, they don't work well with Clut_{TEX}.

However, some packages provide a package option to let them know the location of `-output-directory`. For example, `minted` provides `outputdir`, and `epstopdf` provides `outdir`.

Clut_{TEX} can supply them the appropriate options, but only if it knows that the package is going to be used. To let Clut_{TEX} what packages are going to be used, use `--package-support` option.

For example, if you want to typeset a document that uses `minted`, run the following:

```
cluttex -e pdflatex --shell-escape --package-support=minted document.tex
```

2.11 Check for driver file

Clut_{TEX} can check that the correct driver file is loaded when certain packages are loaded. Currently, the list of supported packages are `graphics`, `color`, `expl3`, `hyperref`, and `xy`.

The check is always done with PDF mode. To check the driver with DVI mode, use `--check-driver` option.

2.12 Customization via a configuration file

Some behavior of Clut_T_EX can be customized via a configuration file.

The configuration file for Clut_T_EX is looked for in the following order:

1. `--config-file` command-line option.
2. `CLUTTEX_CONFIG_FILE` environment variable.
3. (Unix only) `$XDG_CONFIG_HOME/cluttex/config.toml`
4. (Unix only) `$HOME/.config/cluttex/config.toml`
5. (Windows only) `%APPDATA%\cluttex\config.toml`

The configuration file for Clut_T_EX must be written in TOML³ format.

Available keys are:

temporary-directory (string) The location to create auxiliary files. The default value for `--output-directory` will be somewhere under **temporary-directory**. If omitted, somewhere in the temporary directory will be used.

color.{type,execute,warning,diagnostic,information} (table) Set the colors for messages from Clut_T_EX. The table may have the following keys: **fore** (color string), **back** (color string), **bold** (boolean), **dim** (boolean), **underline** (boolean), **blink** (boolean), **reverse** (boolean), **italic** (boolean), **strike** (boolean). A color string is one of the following: **default**, **black**, **red**, **green**, **yellow**, **blue**, **magenta**, **cyan**, **white**, **brightblack**, **brightred**, **brightgreen**, **brightyellow**, **brightblue**, **brightmagenta**, **brightcyan**, **brightwhite**.

An example configuration file might look like:

```
# The actual output directory will be something like
# "/home/user/.cache/cluttex/cluttex-<hash>"
temporary-directory = "/home/user/.cache/cluttex"

[color]
type = { reverse = true }
error = { fore = "brightred", back = "brightwhite" }
```

³<https://toml.io/en/>