**NAME**
>       fmtutil − manage TeX formats and Metafont bases, per-user
>       fmtutil-sys − manage TeX formats and Metafont bases, system-wide
>       mktexfmt − create a TeX format or Metafont base

**SYNOPSIS**
>       **fmtutil** [*-user*|*-sys*] [*OPTION*] ... [*COMMAND*]
>       **fmtutil-sys** [*OPTION*] ... [*COMMAND*]
>       **fmtutil-user** [*OPTION*] ... [*COMMAND*]
>       **mktexfmt** *FORMAT.fmt*|*BASE.base*|*FMTNAME*

**DESCRIPTION**
>       fmtutil version r59143 (2021−05−09 04:23:44 +0200)

>       Rebuild and manage TeX fmts and Metafont bases, collectively called "formats" here. (MetaPost
>       no longer uses the past−equivalent "mems".)

>       If not operating in mktexfmt mode, exactly one command must be given, filename suffixes should
>       generally not be specified, no non−option arguments are allowed, and multiple formats can be
>       generated.

>       If the command name ends in mktexfmt, only one format can be created.  The only options
>       supported are **−−help** and **−−version**, and the command line must be either a format name, with
>       extension, or a plain name that is passed as the argument to **−−byfmt** (see below).  The full name
>       of the generated file (if any) is written to stdout, and nothing else.  The system directories are
>       used if they are writable, else the user directories.

>       By default, the return status is zero if all formats requested are successfully built, else nonzero.

**OPTIONS**
>       **−−sys**    use TEXMFSYS{VAR,CONFIG}

>       **−−user**   use TEXMF{VAR,CONFIG}

>       **−−cnffile** FILE
>               read FILE instead of fmtutil.cnf (can be given multiple times, in which case all the files
>               are used)

>       **−−dry−run**, **−n**
>               don't actually build formts

>       **−−fmtdir** DIR
>               write formats under DIR instead of TEXMF[SYS]VAR

>       **−−no−engine−subdir**
>               don't use engine−specific subdir of the fmtdir

>       **−−no−error−if−no−format**
>               exit successfully if no format is selected

>       **−−no−error−if−no−engine**=*ENGINE1*,ENGINE2,...
>               exit successfully even if a required ENGINE

>               is missing, if it is included in the list.

>       **−−no−strict**
>               exit successfully even if a format fails to build

**−−nohash**
>   don't update ls−R files

**−−recorder**
>   pass the **−recorder** option and save .fls files

**−−refresh**
>   recreate only existing format files

**−−status−file** FILE
>   append status information about built formats to FILE

**−−quiet**
>   be silent

**−−catcfg**
>   (does nothing, exists for compatibility)

**−−dolinks**
>   (does nothing, exists for compatibility)

**−−force**
>   (does nothing, exists for compatibility)

**−−test**   (does nothing, exists for compatibility)

**Commands:**

**−−all**   recreate all format files

**−−missing**
>   create all missing format files

**−−byengine** ENGINE
>   (re)create formats built with ENGINE

**−−byfmt** FORMAT
>   (re)create format FORMAT

**−−byhyphen** HYPHENFILE
>   (re)create formats that depend on HYPHENFILE

**−−enablefmt**
>   FORMAT[/ENGINE]  enable FORMAT, as built with ENGINE

**−−disablefmt** FORMAT[/ENGINE]
>   disable FORMAT, as built with ENGINE If multiple formats have the same name and
>
>   different engines, */ENGINE* specifier is required.

**−−listcfg**
>   list (enabled and disabled) configurations, filtered to available formats

**−−showhyphen** FORMAT
>   print name of hyphen file for FORMAT

**−−version**
>   show version information and exit

**−−help**   show this message and exit

## ENVIRONMENT

Explanation of trees and files normally used:

If **−−cnffile** is specified on the command line (possibly multiple times), its value(s) are used.  Otherwise, fmtutil reads all the fmtutil.cnf files found by running "kpsewhich **−all** fmtutil.cnf", in the order returned by kpsewhich.  Files specified via **−−cnffile** are first tried to be loaded directly, and if not found and the file names don't contain directory parts, are searched via kpsewhich.

In any case, if multiple fmtutil.cnf files are found, all the format definitions found in all the fmtutil.cnf files are merged.

Thus, if fmtutil.cnf files are present in all trees, and the default layout is used as shipped with TeX Live, the following files are read, in the given order.

For fmtutil−sys:
TEXMFSYSCONFIG *$TEXLIVE/YYYY/texmf−config/web2c/fmtutil.cnf*
TEXMFSYSVAR     *$TEXLIVE/YYYY/texmf−var/web2c/fmtutil.cnf*
TEXMFLOCAL      *$TEXLIVE/texmf−local/web2c/fmtutil.cnf*
TEXMFDIST        *$TEXLIVE/YYYY/texmf−dist/web2c/fmtutil.cnf*

For fmtutil−user:
TEXMFCONFIG     $HOME/.texliveYYYY/texmf−config/web2c/fmtutil.cnf
TEXMFVAR          $HOME/.texliveYYYY/texmf−var/web2c/fmtutil.cnf
TEXMFHOME       *$HOME/texmf/web2c/fmtutil.cnf*
TEXMFSYSCONFIG *$TEXLIVE/YYYY/texmf−config/web2c/fmtutil.cnf*
TEXMFSYSVAR     *$TEXLIVE/YYYY/texmf−var/web2c/fmtutil.cnf*
TEXMFLOCAL      *$TEXLIVE/texmf−local/web2c/fmtutil.cnf*
TEXMFDIST        *$TEXLIVE/YYYY/texmf−dist/web2c/fmtutil.cnf*

(where YYYY is the TeX Live release version).

According to the actions, fmtutil might update one of the existing cnf files or create a new fmtutil.cnf, as described below.

Where format files are written:

By default, format files are (re)written in *$TEXMFSYSVAR/ENGINE* by fmtutil−sys, and *$TEXMFVAR/ENGINE* by fmtutil−user, where */ENGINE* is a subdirectory named for the engine used, such as "pdftex".

For mktexfmt, TEXMFSYSVAR is used if it is writable, else TEXMFVAR.

If the **−−fmtdir=***DIR* option is specified, DIR is used instead of TEXMF[SYS]VAR, but the */ENGINE* subdir is still used by default.

In all cases, if the **−−no−engine−subdir** option is specified, the */ENGINE* subdir is omitted.

Where configuration changes are saved:

If config files are given on the command line, then the first one given will be used to save any changes from **−−enable** or **−−disable**.

If the config files are taken from kpsewhich output, then the algorithm is more complicated:

1) If *$TEXMFCONFIG/web2c/fmtutil.cnf* or *$TEXMFHOME/web2c/fmtutil.cnf* appears in the list of used files, then the one listed first by kpsewhich **−−all** (equivalently, the one returned by "kpsewhich fmtutil.cnf"), is used.

2) If neither of the above two are present and changes are made, a new config file is created in *$TEXMFCONFIG/web2c/fmtutil.cnf* .

In general, the idea is that if a given config file is not writable, a higher−level one can be used. That way, the distribution's settings can be overridden system−wide using TEXMFLOCAL, and system settings can be overridden again in a particular user's TEXMFHOME or TEXMFCONF.

Resolving multiple definitions of a format:

If a format is defined in more than one config file, then the definition coming from the first−listed fmtutil.cnf is used.

Disabling formats:

fmtutil.cnf files with higher priority (listed earlier) can disable formats in lower priority (listed later) fmtutil.cnf files by writing a line like this in the higher−priority fmtutil.cnf file:

#! <fmtname> <enginename> <hyphen> <args>

The #! must be at the beginning of the line, with at least one space or tab afterward, and there must be whitespace between each word on the list.

For example, you can disable the luajitlatex format by creating the file *$TEXMFCONFIG/web2c/fmtutil.cnf* with the line

#! luajitlatex luajittex language.dat,language.dat.lua lualatex.ini

(As it happens, the luajittex−related formats are precisely why the **−−no−error−if−no−engine** option exists, since luajittex cannot be compiled on all platforms. So this is not needed.)

fmtutil−user (fmtutil **−user**) vs. fmtutil−sys (fmtutil **−sys**):

When fmtutil−sys is run or the command line option **−sys** is used, TEXMFSYSCONFIG and TEXMFSYSVAR are used instead of TEXMFCONFIG and TEXMFVAR, respectively. This is the primary difference between fmtutil−sys and fmtutil−user.

See https://tug.org/texlive/scripts−sys−user.html for details.

Other locations may be used if you give them on the command line, or these trees don't exist, or you are not using the original TeX Live.

Supporting development binaries:

If an engine name ends with "−dev", formats are created in the respective directory with the **−dev** stripped. This allows for easily running development binaries in parallel with the released binaries.

## REPORTING BUGS

Report bugs to: tex−live@tug.org
TeX Live home page: <https://tug.org/texlive/>