

**NAME**

`texprof` – A profiler for TeX source files

**SYNOPSIS**

**texprof** [*options*] [**&format**] [*file*]\*commands*]

**DESCRIPTION**

Run the TeX profiler on *file*, usually creating *file.dvi* and *file.tprof*. If the *file* argument has no extension, ".tex" will be appended to it. Instead of a file name, a list of TeX commands can be given, the first of which must start with a backslash. With a **&format** argument, the TeX profiler uses a different set of precompiled commands, contained in *format.fmt*; it is usually better to use the **-fmt** *format* option instead.

The TeX profiler is a version of TeX that measures the time TeX needs to execute individual input lines or macros. It writes the collected data to a file with extension *.tprof*. A separate program, **texprofile** (see **texprofile(1)**), is used to display the content of a *.tprof* file in a readable form.

The typical use of the TeX profiler is with pre generated formats. The **texprof** command uses the equivalent of the plain TeX format, and the **latexprof** command uses the equivalent of the L<sup>A</sup>T<sub>E</sub>X format. To investigate the timing of macros targeted to **pdf<sub>te</sub>x** or **pdf<sub>la</sub>te<sub>x</sub>**, you can use **pdf<sub>te</sub>xprof** or **pdf<sub>la</sub>te<sub>x</sub>prof**.

The TeX profiler's handling of its command-line arguments is similar to that of the other TeX programs in the *web2c* implementation.

**OPTIONS**

This version of the TeX profiler understands the following command line options.

**-cnf-line** *string*

Parse *string* as a *texmf.cnf* configuration line. See the Kpathsea manual.

**-etex** Enable the e-TeX extensions. This option is only effective in combination with **-ini**. See **etex(1)**.

**-file-line-error**

Print error messages in the form *file:line:error* which is similar to the way many compilers format them.

**-no-file-line-error**

Disable printing error messages in the *file:line:error* style.

**-fmt** *format*

Use *format* as the name of the format to be used, instead of the name by which the TeX profiler was called or a **%&** line.

**-help** Print help message and exit.

**-ini** Start in *INI* mode, which is used to dump formats. The *INI* mode can be used for typesetting, but no format is preloaded, and basic initializations like setting catcodes may be required.

**-interaction** *mode*

Sets the interaction mode. The mode can be either *batchmode*, *nonstopmode*, *scrollmode*, and *errorstopmode*. The meaning of these modes is the same as that of the

corresponding `\commands`.

**-jobname** *name*

Use *name* for the job name, instead of deriving it from the name of the input file.

**-kpathsea-debug** *bitmask*

Sets path searching debugging flags according to the bitmask. See the *Kpathsea* manual for details.

**-ltx** Enable the L<sup>A</sup>T<sub>E</sub>X extensions. This option is only effective in combination with **-ini**. See **latex(1)**.

**-mktex** *fnt*

Enable `mktex fnt`, where *fnt* must be either *tex*, *tfm*, *fnt*, or *pk*.

**-no-mktex** *fnt*

Disable `mktex fnt`, where *fnt* must be either *tex*, *tfm*, *fnt*, or *pk*.

**-output-directory** *directory*

Write output files in *directory* instead of the current directory. Look up input files in *directory* first, then along the normal search path.

**-parse-first-line**

If the first line of the main input file begins with `%&` parse it to look for a dump name.

**-no-parse-first-line**

Disable parsing of the first line of the main input file.

**-pdf** Enable the simulation of common **pdf<sub>tex</sub>** primitives. This option requires the **-ini** option and implies the **-ltx** option. See **pdf<sub>tex</sub>(1)**.

**-prof** Enable profiling as soon as T<sub>E</sub>X enters the main loop. If this option is not used, profiling needs to be switched on using the `\profileon` primitive. Profiling can be switched off with the `\profileoff` primitive.

**-progname** *name*

Pretend to be program *name*. This affects both the format used and the search paths.

**-version**

Print version information and exit.

## ENVIRONMENT

See the Kpathsea library documentation (e.g., the ‘Path specifications’ node) for precise details of how the environment variables are used. The **kpsewhich** utility can be used to query the values of the variables.

## TEXMFOUTPUT

Normally, T<sub>E</sub>X puts its output files in the current directory. If any output file cannot be opened there, it tries to open it in the directory specified in the environment variable `TEXMFOUTPUT`. There is no default value for that variable. For example, if you say *texprof paper* and the current directory is not writable and `TEXMFOUTPUT` has the value */tmp*, T<sub>E</sub>X attempts to create */tmp/paper.log*, */tmp/paper.dvi*, and */tmp/paper.tprof*. `TEXMFOUTPUT` is also checked for input files, as T<sub>E</sub>X often generates files that need to be subsequently read; for input, no suffixes (such as “.tex”) are added by default, the input name is simply checked as given.

**TEXINPUTS**

Search path for `\input` and `\openin` files. This normally starts with “.”, so that user files are found before system files. An empty path component will be replaced with the paths defined in the *texmf.cnf* file. For example, set TEXINPUTS to “.:/home/user/tex:” to prepend the current directory and “/home/user/tex” to the standard search path.

**TEXFORMATS**

Search path for format files.

**TFMFonts**

Search path for font metric (*.tfm*) files.

**SOURCE\_DATE\_EPOCH**

If set, its value, taken to be in epoch-seconds, will be used for the creation date and as the reference moment for the time related primitives of L<sup>A</sup>T<sub>E</sub>X. This is useful for making reproducible builds.

**FORCE\_SOURCE\_DATE**

If set to the value “1”, the time-related T<sub>E</sub>X primitives (`\year`, `\month`, `\day`, `\time`) are also initialized from the value of SOURCE\_DATE\_EPOCH. This is not recommended if there is any viable alternative.

Many, many more environment variables may be consulted related to path searching. See the Kpathsea manual.

**FILES**

The location of the files mentioned below varies from system to system. Use the **kpsewhich** utility to find their locations.

*\*.tfm* Metric files for T<sub>E</sub>X’s fonts.

*\*.fmt* Predigested T<sub>E</sub>X format files.

*\*.pk* *\*.pfb*

Font files used by T<sub>E</sub>X.

**NOTES**

This manual page is not meant to be exhaustive. The complete documentation for the T<sub>E</sub>X profiler can be found in *texprof.pdf*. Further information can be found in the manual of the *Kpathsea library*.

**BUGS**

This version of the T<sub>E</sub>X profiler does not implement all of the primitives that pdf<sub>TEX</sub> provides, and further, it will not produce the same side effects.

**AVAILABILITY**

The T<sub>E</sub>X profiler should compile on a large variety of machine architectures and operating systems. The function to obtain timing information is POSIX specific.

The sources of the T<sub>E</sub>X profiler are hosted at <https://github.com/ruckertm/HINT>

**SEE ALSO**

**texprofiler(1)**, **tex(1)**, **latex(1)**, **pdftex(1)**, **pdf<sub>TEX</sub>(1)**, and **kpsewhich(1)**.

**AUTHORS**

The primary author of the T<sub>E</sub>X profiler is Martin Ruckert, with eT<sub>E</sub>X extensions by Peter Breitenlohner, L<sup>A</sup>T<sub>E</sub>X extensions by Thierry Laronde, and the kpathsearch library by Karl Berry.

T<sub>E</sub>X was designed by Donald E. Knuth, who implemented it using his WEB system for Pascal programs.

Many, many more contributed to the typesetting system now known as T<sub>E</sub>X; far too many to name all of them here.