NAME

luatex, texlua, texluac − An extended version of pdfTeX using Lua as an embedded scripting language

SYNOPSIS

luatex [−−**lua=***FILE*] [*OPTION*]... [*TEXNAME*[**.tex**]] [*COMMANDS*]
luatex [−−**lua=***FILE*] [*OPTION*]... \*FIRST-LINE*
luatex [−−**lua=***FILE*] [*OPTION*]... **&***FMT* [*ARGS*]

DESCRIPTION

Run the luaTEX typesetter on *TEXNAME*, usually creating *TEXNAME***.pdf**. Any remaining *COMMANDS* are processed as luaTEX input, after *TEXNAME* is read.

Alternatively, if the first non−option argument begins with a backslash, interpret all non−option arguments as a line of luaTEX input.

Alternatively, if the first non−option argument begins with a **&**, the next word is taken as the *FMT* to read, overriding all else. Any remaining arguments are processed as above.

If no arguments or options are specified, prompt for input.

If called as **texlua** it acts as lua interpreter. If called as **texluac** it acts as lua bytecode compiler.

LuaTEX is an extended version of pdfTEX with Unicode and OpenType font support, embeded **Lua** scripting language, the **e-**TEX and **Omega** extensions, as well as integrated MetaPost engine, that can create *PDF* files as well as *DVI* files. For more information about luatex, see http://www.luatex.org, you can read LuaTEX manual using texdoc utility (**texdoc luatex**).

All LuaTEX text input and output is considered to be Unicode text.

In *DVI* mode, luaTEX can be used as a complete replacement for the TEX engine.

In *PDF* mode, luaTEX can natively handle the *PDF*, *JPG*, *JBIG2*, and *PNG* graphics formats. luaTEX cannot include PostScript or Encapsulated PostScript (EPS) graphics files; first convert them to PDF using **epstopdf** (1).

OPTIONS

When the LuaTEX executable starts, it looks for the −−**lua** commandline option. If there is no −−**lua** option, the commandline is interpreted in a similar fashion as in traditional pdfTEX and Aleph. But if the option is present, LuaTEX will enter an alternative mode of commandline parsing in comparison to the standard web2c programs. The presence of **--lua** makes most of other options unreliable, because the lua initialization file can disable kpathsea and/or hook functions into various callbacks.

−−**lua=***FILE*
The lua initialization file.

The following two options alter the executable behaviour:

−−luaonly
Start LuaTEX as a Lua interpreter. In this mode, it will set Lua's *arg[0]* to the found script name, pushing preceding options in negative values and the rest of the commandline in the positive values, just like the Lua interpreter. LuaTEX will exit immediately after executing the specified Lua script.

−−luaconly
Start LuaTEX as a Lua byte compiler. In this mode, LuaTEX is exactly like **luac** from the standalone Lua distribution, except that it does not have the −**l** switch, and that it

accepts (but ignores) the −−**luaconly** switch.

Then the regular web2c options:

−−**debug−format**
Debug format loading.

−−**draftmode**
Sets \pdfdraftmode so luaTEX doesn't write a PDF and doesn't read any included images, thus speeding up execution.

−−**enable−write18**
Synonym for −−**shell−escape**.

−−**disable−write18**
Synonym for −−**no−shell−escape**.

−−**shell−escape**
Enable the **write18**{*command*} construct, and Lua functions **os.execute()**, **os.exec()**, **os.spawn()**, and **io.popen()**. The *command* can be any shell command. This construct is normally disallowed for security reasons.

−−**no−shell-escape**
Disable the **write18**{*command*} construct and the other Lua functions, even if it is enabled in the *texmf.cnf* file.

−−**shell-restricted**
Enable restricted version of **write18**, **os.execute()**, **os.exec()**, **os.spawn()**, and **io.popen()**, only commands listed in *texmf.cnf* file are allowed.

−−**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 luaTEX 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.

−−**mktex=***FMT*
Enable mktex*FMT* generation, where *FMT* must be either *tex* or *tfm*.

−−nosocket
Disable the luasocket (network) library.

−−output−comment=*STRING*
In *DVI* mode, use *STRING* for the *DVI* file comment instead of the date. This option is ignored in*PDF* mode.

−−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.

−−output−format=*FORMAT*
Set the output format mode, where *FORMAT* must be either *pdf* or *dvi.* This also influences the set of graphics formats understood by luaTEX.

−−progname=*NAME*
Pretend to be program *NAME* (only for kpathsea).

−−recorder
Enable the filename recorder. This leaves a trace of the files opened for input and output in a file with extension *.fls.*

−−safer
Disable some Lua commands that can easily be abused by a malicious document.

−−synctex=*NUMBER*
Enable/disable SyncTEX extension.

−−version
Print version information and exit.

−−credits
Print credits and version details.

The following options are ignored:

−−8bit, **−−etex**, **−−parse−first−line**, **−−no−parse−first−line**
These are always on.

−−default−translate−file=*TCXNAME*, **−−translate−file=***TCXNAME*
These are always off.

SEE ALSO

pdftex(1), **etex**(1), **aleph**(1), **omega**(1), **lua**(1).

AUTHORS

The primary authors of LuaTEX are Hartmut Henkel, Taco Hoekwater, and Hans Hagen, with help from Martin Schröder, Karel Skoupy, and Han The Thanh.

TEX was designed by Donald E. Knuth, who implemented it using his WEB system for Pascal programs. It was ported to Unix at Stanford by Howard Trickey, and at Cornell by Pavel Curtis. The version now offered with the Unix TEX distribution is that generated by the WEB to C system (**web2c**), originally written by Tomas Rokicki and Tim Morgan.