NAME

latexmk − generate LaTeX document

SYNOPSIS

latexmk [options] [file ...]

DESCRIPTION

LatexMk completely automates the process of generating a LaTeX document. Essentially, it is a highly specialized relative of the general *make* utility. Given the source files for a document, *latexmk* issues the appropriate sequence of commands to generate a .dvi, .ps, .pdf or hardcopy version of the document. It can also be set to run continuously with a previewer; the latex program, etc, are rerun whenever one of the source files is modified.

Latexmk will normally determine which are the source files by examining the log file. It has an option to parse the TeX file instead --- see later. When *latexmk* is run, it will examine the timestamps on the source files. If any of the source files have been changed since the last document generation, *latexmk* will run the various LaTeX processing programs as necessary. In particular, it will repeat the run of LaTeX often enough to resolve all cross references; depending on the macro packages used. With some macro packages and document styles four, or even more, runs may be needed.

If the option **-lt** to scan the LaTeX file is used, the main LaTeX file and all included files (recursively) are scanned for LaTeX commands for including other TeX files and figure files. Then on subsequent runs, *latexmk* with the **-lt** switch reads the dependency information from this file. If the dependencies of the document are changed (e.g., by adding or removing an \input command), an additional pass of *latexmk -i* or *latexmk -I* will update the dependency file.

Latexmk has two different previewing options. In the simple **-pv** option, a dvi, postscript or pdf previewer is automatically run after generating the dvi, postscript or pdf version of the document. The type of viewer is selected automatically depending on the **-ps** or **-pdf** option.

Particularly when a document is reaching the final stages of editing, it is often useful to have a previewer open continuously and have it update its display whenever changes are made to the source file(s). For this, *latexmk* provides the powerful **-pvc** option (mnemonic: "preview continuously"). With this option, *latexmk* runs a previewer for the document, and then *latexmk* repeatedly monitors the source files of the document to see if any changes have been made since the last dependent file was produced. When changes are detected, *latexmk* runs the appropriate LaTeX commands to regenerate the .dvi, .ps and/or .pdf files (depending on the which of the **-ps** and **-pdf** options was specified, the .ps file. A good previewer (like *gv*) will then automatically update its display. Thus the user can simply edit a file and, when the changes are written to disk, *latexmk* completely automates the cycle of updating the .dvi (and possibly the .ps and .pdf) file, and refreshing the previewer's display. It's not quite WYSIWYG, but usefully close.

For other previewers, the user will have to manually make the previewer update its display, which can be (xdvi and gsview) as forcing a redraw of its display.

Latexmk has the ability to print a banner in gray diagonally across each page when making the postscript file.

LATEXMK OPTIONS AND ARGUMENTS

file One or more files can be specified. If no files are specified, *latexmk* will run on all files in the current working directory with a ".tex" extension and any files that match any file masks specified in the *$texfile_search* variable.

If a file is specified without an extension, then the ".tex" extension is automatically added, just as LaTeX does. (e.g. If you specify:

```
latexmk foo
```

then *latexmk* will operate on the file "foo.tex".

-bm <message>
A banner message to print diagonally across each page when converting the dvi file to postscript. The message must be a single argument on the command line so be careful with quoting spaces and such.

Note that if the **-bm** option is specified, the **-ps** option is assumed and the postscript file is always generated, even if it is newer than the dvi file.

-bi <intensity>
How dark to print the banner message. A decimal number between 0 and 1. 0 is black and 1 is white, default is 0.95 which is OK unless your toner cartridge is getting low.

-bs <scale>
A decimal number that specifies how large the banner message will be printed. Experimentation is necessary to get the right scale for your message, as a rule of thumb the scale should be about equal to 1100 divided by the number of characters in the message. Default is 220.0 which is just right for 5 character messages.

--commands
List the commands used by *latexmk* for processing files, and then exit.

-c Clean up (remove) all unnecessary files generated by *latex* and *bibtex* except dvi, postscript and pdf.

-C Clean up (remove) all unnecessary files generated by *latex* and *bibtex* including aux, dep, dvi, postscript and pdf.

-c1 Clean up (remove) all unnecessary files generated by *latex* and *bibtex* except aux and dep.

-d Set draft mode. This prints the banner message "DRAFT" across your page when converting the dvi file to postscript. Size and intensity can be modified with the **-bs** and **-bi** options. The **-bm** option will override this option as this is really just a short way of specifying:

```
latexmk -bm DRAFT
```

Note that if the **-d** option is specified, the **-ps** option is assumed and the postscript file is always generated, even if it is newer than the dvi file.

-dF Dvi file filtering. The argument to this option is a filter which will generate a filtered dvi file with the extension ".dviF". All extra processing (e.g. conversion to postscript, preview, printing) will then be performed on this filtered dvi file.

Example usage: To use dviselect to select only the even pages of the dvi file:

```
latexmk -dF 'dviselect even' foo.tex
```

-dvi Generate dvi version of document.

-dvi- Turn off generation of dvi version of document. (This may get overridden, if some other file is made (a .ps file) that is generated from the dvi file, or if no generated file at all is requested.)

--diagnostics
Whenever a log file is parsed to determine the included files, print a list of included files.

-f Force *latexmk* to continue document processing despite errors. Normally, when *latexmk* detects that *latex* has found an error which will not be resolved by further processing, the program terminates.

-F Force *latexmk* to include files that don't exist when generating dependency files. A warning is produced instead of an error message and the program terminating. If the file name is not an absolute path, it is assumed to be relative to the current working directory.

-g Force *latexmk* to process document, disregarding the timestamps of the source files. This option is particularly useful when updating style files, which are not included in the file dependency information.

-h, --help
Print help information.

-i Generate new dependency file if root file is newer than dependency file or dependency file does not exist. The dependency information is taken from the log file or the source file, depending on the setting made by the **-il** and **-it** switches.

-il Extract dependency information from log file. This is normally the best method, so it is the default. However, some packages do not put information on files read in the log file, and then it may be better to tell *latexmk* to get the information from the TeX file(s) --- see the switch **-it**. (Default)

-it Extract dependency information by scanning the source TeX file(s), rather than the log file. This is the method used by earlier versions of *latexmk*. It relies on parsing TeX files, which can be confused by definitions of new commands, etc. Normally it is better to tell *latexmk* to scan the log file --- see the switch **-il**.

-I Always generate new dependency file, even if newer dependency file exists. The dependency information is taken from the log file or the source file, depending on the setting made by the **-il** and **-it** switches.

-l Run in landscape mode, using the landscape mode previewers and dvi to postscript converters.

-p Print out the file using lpr after generating the postscript version.

-pdf Generate pdf version of document using pdflatex.

-pdf- Turn off generation of pdf version of document using pdflatex. This can be used to override a setting in a configuration file. It may get overridden if some other option uses the pdf file.

-ps Generate postscript version of document.

-ps- Turn off generation of postscript version of document. This can be used to override a setting in a configuration file. It may get overridden by some other option that requires a postscript file, for example a request for printing.

-pF Postscript file filtering. The argument to this option is a filter which will generate a filtered postscript file with the extension ".psF". All extra processing (e.g. preview, printing) will then be performed on this filtered postscript file.

Example usage: Use psnup to print two pages on the one page:

latexmk -ps -pF ’psnup -2’ foo.tex

-pv Run file previewer. If **-pdf** is specified, then *latexmk* will run the pdf previewer. If **-ps** is specified, then *latexmk* will run the postscript previewer. Otherwise it will run the dvi previewer.

-pvc Run a file previewer and continually update the .dvi, .ps, and/or .pdf files whenever changes are made to source files (see the Description above). If **-pdf** is specified, the .pdf file will be updated and a pdf previewer will be used. If **-ps** is specified, the .dvi and .ps files will be updated and a postscript previewer will be used. Otherwise the .dvi file will be updated and a dvi previewer will be used.

With a good previewer the display will be automatically updated. (Under UNIX "gv -watch" does this for postscript files; it would also do it for pdf files except for an apparent bug in gv that causes an error when the newly updated pdf file is read.)

Note that if *latexmk* dies because it encounters an error, the forked previewer will continue to run. Successive invocations with the **-pvc** option will not fork new previewers, but will use the existing previewer. (At least this will happen when *latexmk* is running under an operating system where it knows how to determine whether an existing previewer is running.)

-r <rcfile>
Read specified initialization file ("RC file") before processing. Since it may be used override options specified before the **-r** option, therefore it is a good idea have the habit of specifying the -r option first. See below for more details about RC files.

--silent Run commands silently, i.e., with options that reduce the amount of diagnostics generated. For example, with the default settings for commands under UNIX, the command "latex -interaction=batchmode" is used for latex.

-v, --version
Print version number of *Latexmk*.

--verbose
Opposite of **--silent**. This is the default setting.

--view=default, --view=dvi, --view=ps, --view=pdf
Set the kind of file used when previewing is requested (e.g., by the **-pv** or **-pvc** switches). The default is to view the "highest" kind of requested file (in the order dvi, ps, pdf).

If **-ps** are specified when the dependency file is created or updated, then these options are stored in the dependency file. This means that these options will automatically be activated without having to be explicitly specified on the command line in future runs. Of coarse, if the dependency file is updated or deleted, these automatic settings will be lost.

Options **-pv** and **-pvc** require one and only one filename specified on the command line.

Options **-p**, **-pv** and **-pvc** are mutually exclusive.

EXAMPLES

% **latexmk thesis**	*# run latex enough times to resolve cross-references*
% **latexmk -pvc -ps -f thesis**	*# run latex enough times to resolve cross-references, make a postscript file, start a previewer. Then watch for changes in of the source file thesis.tex and any files it uses. After any changes rerun latex the appropriate number of times and remake the postscript file. If latex encounters an error, latexmk will keep running (as indicated by the* **-f** switch).
% **latexmk -c**	*# remove .aux, .log, .bbl, .blg, .dep, .dvi, .pdf, .ps & .bbe files*

INITIALIZATION (RC) FILES

There are four initialization files ("RC files") that *latexmk* can read at startup:

1) The system RC file. On a UNIX system, it may be any one of the following "/opt/local/share/latexmk/LatexMk", "/usr/local/share/latexmk/LatexMk", "/usr/local/lib/latexmk/LatexMk". On a MS-WINDOWS system it is "C:\latex\LatexMk".

2) The user's RC file in "$HOME/.latexmkrc".

3) The RC file in the current working directory called "latexmkrc".

4) The RC file specified on the command line with the **-r** option.

Each RC file is a sequence of Perl commands. Usually it will be just a sequence of assignment statements that override the built-in settings of *Latexmk*. Comment lines are introduced by the "#" character.

RC VARIABLES

Many of the available variables that can be set are shown below. Syntax for the statements in an initialization file is of the form:

$bibtex = 'bibtex';

If a command is to be run detached this is indicated by preceding it with "start", as in

$dvi_previewer = 'start xdvi';

If a command is not to be run, the command name NONE is used, as in

$lpr = 'NONE lpr';

Default values are indicated in brackets.

$banner [0]
If nonzero, the banner message is printed across each page when converting the dvi file to postscript. Without modifying $banner_message, this is equivalent to specifying the **-d** option.

Note that if **$banner** is nonzero, the **$postscript_mode** is assumed and the postscript file is always generated, even if it is newer than the dvi file.

$banner_intensity [0.95]
Equivalent to the **-bi** option, this is a decimal number between 0 and 1 that specifies how dark to print the banner message. 0 is black, 1 is white. The default is just right if your toner cartridge isn't running too low.

$banner_message ["DRAFT"]
The banner message to print across each page when converting the dvi file to postscript. This is equivalent to the **-bm** option.

$banner_scale [220.0]
A decimal number that specifies how large the banner message will be printed. Experimentation is necessary to get the right scale for your message, as a rule of thumb the scale should be about equal to 1100 divided by the number of characters in the message. The Default is just right for 5 character messages. This is equivalent to the **-bs** option.

$bibtex ["bibtex"]
The BibTeX processing program.

$bibtex_silent ["-terse"]
Switches for the BibTeX processing program when silent mode is on.

$bibtex_mode [0]
If nonzero, the document has a bibtex bibliography. Set by the **-i** and **-I** options and saved in the dependency file. Should not need to set this in an RC file.

$cleanup_mode [0]
If nonzero, specifies cleanup mode. All other options ignored. Equivalent to specifying the **-c** option. Recommend that this is not set from an RC file.

$clean_ext [""]
Extra extensions of files for *latexmk* to remove when the **-c** option is selected.

$clean_full_ext [""]
Extra extensions of files for *latexmk* to remove when the **-C** option is selected.

$cus_dep_list [empty]
Custom dependency list (see below).

$dvi_filter [empty]
The dvi file filter to be run on the newly produced dvi file before other processing. Equivalent to specifying the **-dF** option.

$dvi_previewer ["start xdvi"]
[Default is "start yap" under MS-WINDOWS.] The dvi-previewer to run with the **-pv** option.

$dvi_previewer_landscape ["start xdvi"]
[Default is "start yap" under MS-WINDOWS.] The dvi-previewer to run with the **-pv** option in landscape mode.

$dvips ["dvips"]
The program to used as a filter to convert a .dvi file to a .ps file.

$dvips_landscape ["dvips -tlandscape"]
The program to used as a filter to convert a .dvi file to a .ps file in landscape mode.

$dvips_silent ["-q"]
Switches for dvips program when silent mode is on.

$dvi_update_method [2 under UNIX, 1 under MSWindows]
How the dvi viewer updates its display when the dvi file has changed. 0 => update is automatic, 1=> manual update, by user, which may only mean a mouse click on the
viewer's window or may mean serious action. 2 => Send the signal SIGUSR1 (as for xdvi under UNIX), 3 => Viewer cannot do an update, because it locks the file.

$force_generate_and_save_includes [0]
If nonzero, specifies that the dependency file should always be generated. Equivalent to specifying the **-I** option.

$force_mode [0]
If nonzero, continue processing past minor *latex* errors including unrecognized cross references. Equivalent to specifying the **-f** option.

$force_include_mode [0]
If nonzero, force *latexmk* to include files that don't exist when generating dependency files. A warning is produced instead of an error message and the program terminating. If the file name is not an absolute path, it is assumed to be relative to the current working directory. Equivalent to specifying the **-F** option.

$generate_and_save_includes [0]
If nonzero, generates dependency file if it does not exist or is older than the root file (the base .tex file). Equivalent to specifying the **-i** option.

$go_mode [0]
If nonzero, process files regardless of timestamps. Equivalent to the **-g** option.

$includes_from_log [1]
If nonzero, extract dependency information from the log file, rather than the source TeX file(s); this is equivalent to the **-il** option. If zero, extract dependency information by scanning the source TeX file(s); this is equivalent to the **-it** option.

$index_mode [0]
If nonzero, run *makeindex* to produce index of document. Set by the **-i** and **-I** options and saved in the dependency file. Should not need to set this in an RC file.

$landscape_mode [0]
If nonzero, run in landscape mode, using the landscape mode previewers and dvi to postscript converters. Equivalent to the **-l** option.

$latex ["latex"]
The LaTeX processing program.

$latex_silent ["latex -interaction=batchmode"]
Command to invoke the LaTeX processing program when silent mode is on. Under MSWindows, the default command is changed to "latex -interaction=batchmode -c-style-errors", as used by MikTeX.

$lpr ["lpr"]
[Default is "NONE lpr" under MS-WINDOWS.] The printing program.

$makeindex ["makeindex"]
The index processing program.

$pdf_mode [0]
If nonzero, generate a postscript version of the document. Set by the **-i** and **-I** options if $slide_mode is set. Equivalent to the **-pdf** option and is stored in the dependency file if generated.

$pdflatex ["pdflatex"]
The LaTeX processing program in the version that makes a pdf file instead of a dvi file.

$pdflatex_silent ["pdflatex -interaction=batchmode"]
Command to invoke the pdflatex program (for directly generating a pdf file) when silent mode is on. Under MSWindows, the default command is changed to "pdflatex -interaction=batchmode -c-style-errors", as used by MikTeX.

$pdf_previewer ["start acroread"]
[Default is changed to "start" on MS-WINDOWS, which launches the program associated with the .pdf file extension, which is commonly acroread.] The pdf previewer to run with the **-pv** option. Potential problem: if acroread is used as the pdf previewer, and it is actually viewing a pdf file that pdflatex wants to write, pdflatex will fail. This problem does not occur if ghostview, gv or psview is used to view pdf files.

$pdf_update_method [1 under UNIX, 3 under MSWindow]
How the pdf viewer updates its display when the pdf file has changed. See $dvi_update_method for the codes. Note that acroread under MSWindows (but not UNIX) locks the pdf file, so the default value is then 3.

$pid_position = [1 under UNIX, -1 under MSWindows]
Command used to get all the processes currently run by the user. The -pvc option uses the command specified by the variable $pscmd to determine if there is an already running previewer, and to find the process ID (needed if *latexmk* needs to signal the previewer about file changes). The variable $pid_position is used to specify which word in lines of the output from $pscmd corresponds to the process ID. The first word in the line is numbered 0. The default value of 1 (2nd word in line) is correct for Solaris 2.6 and Linux. Setting the variable to -1 is used to indicate that $pscmd is not to be used.

$postscript_mode [0]
If nonzero, generate a postscript version of the document. Set by the **-i** and **-I** options if $slide_mode is set. Equivalent to the **-ps** option and is stored in the dependency file if generated.

$preview_continuous_mode [0]
If nonzero, run the **$dvi_previewer** to preview the document, and continue running *latexmk* to keep .dvi up-to-date. Equivalent to the **-pvc** option.

$preview_mode [0]
If nonzero, run the **$ps_previewer** to preview the document. Equivalent to the **-pv** option.

$printout_mode [0]
If nonzero, print the document using *lpr*. Equivalent to the **-p** option. Recommend that this is not set from an RC file or you could waste lots of paper.

$pscmd
[On UNIX, the default is "ps -f -u $ENV{USER}". On MS-WINDOWS the default in "NONE pscmd".] Command used to get all the processes currently run by the user. This is used by the

-pvc option to determine if there is an already running previewer. The command line options for this command under the different flavors of UNIX are quite variable. The command given is suitable for Solaris 2.6 and for Linux. The variable $pid_position must also be set: see its description.

$ps_filter [empty]
The postscript file filter to be run on the newly produced postscript file before other processing. Equivalent to specifying the **-pF** option.

$ps_previewer ["start gv -watch"]
[Default is "start c:/gstools/gsview/gsview32" on MS-WINDOWS.] The postscript previewer to run with the **-pv** option. Note that gv with the -watch option updates its display whenever the postscript file changes, whereas ghostview does not.

$ps_previewer_landscape ["start gv -swap -watch"]
[Default is "start c:/gstools/gsview/gsview32" on MS-WINDOWS.] The postscript previewer to run with the **-pv** option in landscape mode.

$ps_update_method [0 under UNIX, 1 under MSWindows]
How the postscript viewer updates its display when the ps file has changed. See $dvi_update_method for the codes.

$sleep_time [2]
The time to sleep (in seconds) between checking for source file changes when running the **-pvc** option.

$slide_mode [0]
If nonzero, document is a Slitex document. Set by the **-i** and **-I** options or the **-s** option. Saved in the dependency file if generated.

$texfile_search [""]
If no files were specified on the command line, then *latexmk* will search for and files in the current working directory that end with a ".tex" extension or match the file mask specified in this variable.

An example is:

```
$texfile_search = "$texfile_search *.ftex *.c.tex";
```

$tmpdir ["/tmp"]
[Default is the value of the environment variable TEMP under MS-WINDOWS.] Directory to store very small temporary files that *latexmk* generates while running.

CUSTOM DEPENDENCIES

In any RC file a set of custom dependencies can be set up to convert a file with one extension to a file with another. An example use of this would be to allow *latexmk* to convert a *.fig* file to dependencies are set up by using the **@cus_dep_list** array. Each string in the array has four arguments, separated by a space:

from extension:
The extension of the file we are converting from (e.g. "fig").

to extension:
The extension of the file we are converting to (e.g. "eps").

must: If non-zero, the file we are converting from **must** exist, if it doesn't exist *latexmk* will give an error message and exit unless the **-f** option is specified. If *must* is zero and the file we are converting from doesn't exist, then no action is taken.

function:
The name of the subroutine that *latexmk* should call to perform the file conversion. The first argument to the subroutine is the base name of the file to be converted without any extension. The subroutines are declared in the syntax of *perl*.

Example in an RC file to convert a *.fig* file to a *.eps* file:

```
@cus_dep_list = (@cus_dep_list, "fig eps 0 fig2eps");

sub fig2eps {
  system("fig2dev -Lps $_[0].fig $_[0].eps"); }
```

The subroutine *fig2eps* will only be called if the *.fig* file was modified more recently then the *.eps* file, or the *.eps* file does not exist.

If the return value of the subroutine is non-zero, then *latexmk* will assume an error occurred during the execution of the subroutine.

SEE ALSO

latex(1), bibtex(1).

BUGS

Search for .bib files is not correct if they are not in the current directory; the problem is that the log file generated by bibtex does not give the full path to the .bib files. The easiest fix at the moment is to set the BIBINPUTS environment variable to include explicitly the path containing your .bib files. Or you can set the $BIBINPUTS variable explicitly in one of latexmk's startup files.

Preview continuous mode only works perfectly with certain previewers: Xdvi on UNIX/LINUX works for dvi files. Gv on UNIX/LINUX works for both postscript and pdf. Ghostview on UNIX/LINUX needs a manual update (reopen); it views postscript and pdf. Gsview under MS-Windows works for both postscript and pdf, but only reads the updated file when its screen is refreshed. Acroread under UNIX/LINUX views pdf, but the file needs to be closed and reopened to view an updated version. Under MS-Windows, acroread locks its input file and so the pdf file cannot be update. (Remedy: configure *latexmk* use gsview instead.)

THANKS TO

David Coppit (david@coppit.org) made many useful suggestions that contributed to version 3.

IDENTIFICATION

Current version, with substantial modifications, enhancements and bug fixes by John Collins (collins@phys.psu.edu). (Version 3.04).
Modifications and enhancements by Evan McLean (Version 2.0)
Original script called "go" by David J. Musliner (RCS Version 3.2)