MiKTeX Manual

Revision 2.9.6809

Christian Schenk
# Table of Contents

About this Document ................................................................................................. v  
I. User Guide .................................................................................................. 1  
  1. Introduction ............................................................................................................ 3  
     About this Manual ................................................................................................. 3  
     About MiKTeX ........................................................................................................ 3  
     How to Get MiKTeX ............................................................................................... 3  
     Give Back ............................................................................................................... 3  
     The MiKTeX Project Page ...................................................................................... 3  
     Documentation ...................................................................................................... 4  
  2. Installing MiKTeX .................................................................................................. 5  
     Windows .............................................................................................................. 5  
     macOS ............................................................................................................... 5  
     Linux .................................................................................................................. 5  
  3. Using MiKTeX ...................................................................................................... 6  
     Getting Started ................................................................................................... 6  
     Unique TeX features ............................................................................................ 6  
        Automatic Package Installation ....................................................................... 6  
        Finding out Package Usages ........................................................................... 6  
        Specifying Additional Input Directories ....................................................... 7  
        Specifying the Directory for Auxiliary Files .............................................. 7  
     texify: The MiKTeX Compiler Driver .................................................................. 7  
     Printing .............................................................................................................. 8  
        Using a Viewer to Print DVI/PDF Files ......................................................... 8  
        Using mtprint to Print DVI Files ................................................................... 8  
  4. Maintenance ....................................................................................................... 9  
     Refreshing the File Name Database ................................................................... 9  
     Setting the Preferred Paper Format .................................................................. 9  
     Installing Updates .............................................................................................. 9  
     Automatic Package Installation ....................................................................... 9  
     Integrating Local Additions ............................................................................. 9  
  5. Advanced Topics ............................................................................................... 11  
     Managing Font Map Files ............................................................................... 11  
     Changing TEXMF run-time parameters ........................................................ 11  
  6. Programs ........................................................................................................... 14  
     miktex-bibtex ..................................................................................................... 15  
     miktex-dvicopy .................................................................................................... 17  
     miktex-dvips ...................................................................................................... 19  
     findtexmf .......................................................................................................... 25  
     miktex-gftodvi ................................................................................................... 27  
     initexmf ............................................................................................................ 29  
     miktex-luatex .................................................................................................... 32  
     miktex-mf .......................................................................................................... 36  
     miktexsetup ........................................................................................................ 40  
     mpm .................................................................................................................... 43  
     miktex-npost ....................................................................................................... 47  
     mthelp ................................................................................................................ 49  
     mtprint ............................................................................................................... 51  
     miktex-pdfTeX .................................................................................................... 52  
     setupwiz ............................................................................................................. 57  
     miktex-tex ......................................................................................................... 59
About this Document

This is version 2.9.6809 of the MiKTeX manual. It corresponds to MiKTeX 2.9.7300 as of February 13, 2020.
Part I. User Guide
Table of Contents

1. Introduction .................................................................................................................. 3
   About this Manual ........................................................................................................ 3
   About MiKTeX ............................................................................................................. 3
   How to Get MiKTeX ..................................................................................................... 3
   Give Back .................................................................................................................... 3
   The MiKTeX Project Page ......................................................................................... 3
   Documentation ............................................................................................................ 4
2. Installing MiKTeX .......................................................................................................... 5
   Windows ....................................................................................................................... 5
   macOS ......................................................................................................................... 5
   Linux .......................................................................................................................... 5
3. Using MiKTeX .............................................................................................................. 6
   Getting Started .......................................................................................................... 6
   Unique TeX features ..................................................................................................... 6
   Automatic Package Installation .................................................................................. 6
   Finding out Package Usages ....................................................................................... 6
   Specifying Additional Input Directories ..................................................................... 7
   Specifying the Directory for Auxiliary Files .............................................................. 7
   texify: The MiKTeX Compiler Driver ......................................................................... 7
   Printing ....................................................................................................................... 8
   Using a Viewer to Print DVI/PDF Files ..................................................................... 8
   Using mtprint to Print DVI Files ............................................................................... 8
4. Maintenance .................................................................................................................. 9
   Refreshing the File Name Database ............................................................................ 9
   Setting the Preferred Paper Format .......................................................................... 9
   Installing Updates ...................................................................................................... 9
   Automatic Package Installation .................................................................................. 9
   Integrating Local Additions ....................................................................................... 9
5. Advanced Topics .......................................................................................................... 11
   Managing Font Map Files ......................................................................................... 11
   Changing TEXMF run-time parameters ..................................................................... 11
Chapter 1. Introduction

About this Manual

This manual is about MiKTeX, a modern implementation of TeX & Friends.

If you are not yet familiar with using TeX (LaTeX), then please consider reading one of the tutorials available on the Internet.

About MiKTeX

TeX is a typesetting system invented by D. E. Knuth. MiKTeX (pronounced mik-tech) is an implementation of TeX and related programs.

MiKTeX is available for Windows, macOS and selected Linux distributions. Its main features include:

• Easy to install

• Integrated package management: MiKTeX's integrated package manager installs missing components from the Internet, if required. This allows users to keep the TeX installation as minimal as possible (“Just enough TeX”).

• Complete: the MiKTeX distribution contains almost all packages that are freely redistributable.

• Always up-to-date: the MiKTeX package repository is updated regularly and MiKTeX provides tools to easily install package updates.

• Open source

How to Get MiKTeX

The MiKTeX installer for Windows and macOS can be downloaded from the MiKTeX download page [https://miktex.org/download].

For selected Linux distributions, you can install MiKTeX via the system package manager frontend. The MiKTeX download page [https://miktex.org/download] has relevant information available.

Give Back

If you enjoy MiKTeX and want to support the project, then please become a known MiKTeX user by giving back something. It encourages me to continue, and is the perfect way to say thank you!

Visit the MiKTeX Give Back page [https://miktex.org/giveback], for more information.

The MiKTeX Project Page

The MiKTeX Project Page [https://miktex.org/] is the address to turn to for MiKTeX related news & information.
Documentation

Use the \texttt{mthelp} utility to quickly access general TeX related documentation. For example, run \texttt{mthelp memoir} to view documentation of the \texttt{memoir} package.
Chapter 2. Installing MiKTeX

Windows

You use the Basic MiKTeX Installer to install MiKTeX on your Windows computer. Please read the installation tutorial [https://miktex.org/howto/install-miktex], for step-by-step guidance.

The installer is available on the download page [https://miktex.org/download].

macOS

MiKTeX for macOS is distributed as a disk image (.dmg) file. To set up MiKTeX, download and open the disk image. Then drag the MiKTeX icon onto the Applications folder. Please read the installation tutorial [https://miktex.org/howto/install-miktex-mac], for step-by-step guidance.

The disk image is available on the download page [https://miktex.org/download].

Linux

MiKTeX is available for selected Linux distributions.

Installing MiKTeX on Linux involves these steps:

1. Register the GPG key with which MiKTeX installation packages and metadata is signed.

2. Register the installation source which contains the MiKTeX installation package. The installation source depends on the Linux distribution version.

3. Use the package management system to install MiKTeX.

Please read the installation tutorial [https://miktex.org/howto/install-miktex-unx], for step-by-step guidance.
Chapter 3. Using MiKTeX

Getting Started

If you have never used TeX before, then it is recommendable to work through one of the TeX/LaTeX tutorials. A good starting point is the TeX FAQ: https://texfaq.org/.

MiKTeX doesn't differ very much from any other TeX system you might have used before. Typesetting with MiKTeX involves these steps:

1. Start TeXworks (a sophisticated TeX frontend) and edit your LaTeX document.
2. Press Ctrl+T to create a typeset view of your document.

Unique TeX features

This section describes features that are unique to MiKTeX's TeX implementation.

Automatic Package Installation

MiKTeX can be configured in such a way that missing packages are automatically installed (see the section called "Automatic Package Installation").

It is possible to override the global configuration setting with these command line options:

--disable-installer Missing packages will not be installed.
--enable-installer Missing packages will be installed.

Finding out Package Usages

The command line option --record-package-usages can be used to find out which packages are used in a job.

For example, you would say

 latex --record-package-usages=packages.txt test

to create the file packages.txt, which contains the names of the packages used by test.tex.

If test.tex looks like this:

\documentclass{scrartcl}
\begin{document}
Hello, world!
\end{document}

Then the resulting packages.txt would contain these lines:

cm
koma-script
ltxbase
Using MiKTeX

The package list can be handed over to the package manager (see mpm(1)), e.g.

```
mpm --update-some=packages.txt
```

would ensure that you have the latest versions installed.

## Specifying Additional Input Directories

The command-line option `--include-directory=dir` causes the program to include `dir` into the list of input directories.

For example:

```
latex --include-directory="C:\My Styles" foo.tex
```

This prepends `C:\My Styles` to the input search path, i.e., `C:\My Styles` will be searched first, when TeX tries to find an input file.

## Specifying the Directory for Auxiliary Files

The option `--aux-directory=dir` causes TeX to create auxiliary files in another directory. For example:

```
> mkdir C:\texoutput
> mkdir C:\tobedeleted
> latex -output-directory=C:\texoutput -aux-directory=C:\tobedeleted foo.tex
...>
```

This ensures that 1) `foo.dvi` will be created in `C:\texoutput` and 2) all other files (`foo.log`, ...) will be created in `C:\tobedeleted`.

## texify: The MiKTeX Compiler Driver

`texify` is a command-line utility that simplifies the creation of DVI (PDF) documents: `texify` automatically runs LaTeX (pdfLaTeX), MakeIndex and BibTeX as many times as necessary to produce a DVI (PDF) file with sorted indices and all cross-references resolved.

To run `texify` on an input file `foo.tex`, do this:

```
texify foo.tex
```

As shown in the example above, the input file names to `texify` must include any extension (.tex, .ltx, ...).

There are several command line options you can use to control `texify` (see `texify(1)`). Here are some examples:

```
texify --clean foo.tex
```

All auxiliary files will be removed, i.e., only the output `foo.dvi` file will be left in the current folder.

```
texify --tex-option=--src foo.tex
```

Passes the option `--src` to the TeX compiler.
texify --run-viewer foo.tex

Opens the output file foo.dvi (unless there are compile errors).

```
texify --tex-option=--src --viewer-option="-1 -s"200 foo.tex"" --run-viewer foo.tex
```

Compiles foo.tex with source file information (--src) and then initiates forward DVI search to open foo.dvi at the source special location “200 foo.tex”. The previewer option –1 re-uses an existing previewer window.

See the Yap manual, for a complete list of previewer options.

**Printing**

**Using a Viewer to Print DVI/PDF Files**

TeX output files (*.dvi/*.pdf) can be printed from within the viewer.

**Using mtprint to Print DVI Files**

DVI files can also be printed with the help of the command-line utility mtprint (MiKTeX Print Utility).

For example, run `mtprint paper` to send the DVI file `paper.dvi` to the default Windows printer.

See mtprint(1), for more information about mtprint
Chapter 4. Maintenance

Refreshing the File Name Database

To speed up file search, MiKTeX makes use of a list of known file names. This list is called the *file name database* (FNDB).

It is necessary that you refresh the file name database whenever you manually install TeX/LaTeX-related files in a user-managed TEXMF directory.

You can update the file name database with the help of MiKTeX Console [https://miktex.org/howto/miktex-console]. If you prefer the command-line, you can use the MiKTeX Configuration Utility (see initexmf(1)).

Setting the Preferred Paper Format

You can set the preferred paper format with the help of MiKTeX Console [https://miktex.org/howto/miktex-console].

Installing Updates

You can use MiKTeX Console to install the latest MiKTeX updates.

To start MiKTeX Console, search and click the MiKTeX Console icon in the application launcher. Please read the MiKTeX Console tutorial [https://miktex.org/howto/miktex-console], for a step-by-step guide.

Automatic Package Installation

MiKTeX has the ability to automatically install missing packages.

MiKTeX asks your permission before installing a package.

Click Install to start the installation of the package. Click Cancel, to cancel the installation. If you do not want to see this dialog in the future, clear the mark from the check box Always show this dialog before installing packages. Your decision will be remembered.

Integrating Local Additions

If you have files that you want to integrate into the MiKTeX setup, you have several options:

**Use the command-line option --include-directory=dir**

For example:

```
latex --include-directory=C:\path\to\my\style\files thesis.tex
```

See the section called “Specifying Additional Input Directories”, for more information.
Set environment variables

For example:

```bash
set TEXINPUTS=C:\path\to\my\style\files
latex thesis.tex
```

See Chapter 8, *Environment variables*, to learn more about MiKTeX environment variables.

Register a user-managed TEXMF root directory

Register the root of the directory tree which contains your files. The directory tree must conform to the TDS standard, i.e., you must imitate the directory tree in the MiKTeX installation directory (usually `C:\Program Files\MiKTeX 2.9`).

**Tip**

This is the recommended method. You can register TEXMF root directories with the help of MiKTeX Console [https://miktex.org/howto/miktex-console].
Chapter 5. Advanced Topics

Managing Font Map Files

Information about outline fonts is stored in a file by the name of psfonts.map. This file is normally created automatically. It can be manually created by running initexmf --mkmaps (see initexmf(1)).

psfonts.map depends on the file updmap.cfg. This configuration file contains declarative instructions (see updmap.cfg(5)), which will be used to build psfonts.map.

Caution

The contents of psfonts.map should never be edited directly. Your modifications get lost when you install new packages.

For example, follow these steps if you want to add an entry for the font map file xyz.map:

1. Run initexmf --edit-config-file updmap.
2. Insert the following line at the end of the file:
   Map xyz.map
3. Save the file and close the editor.
4. Run initexmf --mkmaps to rebuild the font map files.

Changing TEXMF run-time parameters

You can control a number of run-time parameters (in particular, array sizes) on the command-line or in a configuration file.

Some of the more interesting parameters:

main_memory    Total words of memory available, for TeX, METAFONT, and MetaPost.
extra_mem_bot   Extra space for large TeX data structures: boxes, glue, breakpoints, et al.
font_mem_size   Words of font info available for TeX.

See Chapter 10, Run-Time Defaults, for a complete list of the TEXMF run-time parameters.

Here is a typical example of a configuration file:

main_memory=2000000
extra_mem_bot=2000000
font_mem_size=2000000

The name of the configuration file is that of the engine (e.g., miktex-pdftex) or format (e.g. miktex-pdflatex). You use the --edit-config-file option of initexmf to edit the configuration file, e.g.:

> initexmf --edit-config-file=pdflatex

>
Part II. Reference
# Table of Contents

6. Programs .......................................................... 14  
   miktex-bibtex .................................................. 15  
   miktex-dviciope .............................................. 17  
   miktex-dvips .................................................. 19  
   findtexmf .................................................... 25  
   miktex-gftodvi ............................................... 27  
   initexmf ..................................................... 29  
   miktex-luatex ................................................ 32  
   miktex-mf .................................................... 36  
   miktexsetup .................................................. 40  
   mpm .......................................................... 43  
   miktex-mpost ................................................. 47  
   ntheip ....................................................... 49  
   mtprint ....................................................... 51  
   miktex-pdftex .............................................. 52  
   setupwiz .................................................... 57  
   miktex-tex ................................................... 59  
   texify ....................................................... 64  
   miktex-xetex ................................................. 66  
7. Files .......................................................... 70  
   miktex.ini .................................................... 71  
   pdftex.cfg ................................................... 87  
   updmap.cfg .................................................. 89  
8. Environment variables ........................................ 91  
9. Trace Streams .................................................. 92  
10. Run-Time Defaults ........................................... 94  
    All MiKTeX Programs ......................................... 94  
    All TeXMF Programs ....................................... 109  
    All TeX Programs .......................................... 110  
    Omega ..................................................... 111  
    pdfTeX ..................................................... 111  
    METAFONT & MetaPost ...................................... 111  
    METAFONT .................................................. 111  
    MetaPost .................................................. 112
Chapter 6. Programs
Name
miktex-bibtex — make a bibliography for LaTeX

Synopsis
miktex-bibtex [option...] auxfile

Description
This man page is an adaption of the corresponding TeX Live man page.

BibTeX reads the top-level auxiliary (.aux) file that was output during the running of miktex-latex(1) miktex-tex(1) and creates a bibliography (.bbl) file that will be incorporated into the document on subsequent runs of LaTeX or TeX.

BibTeX looks up, in bibliographic database (.bib) files specified by the \bibliography command, the entries specified by the \cite and \nocite commands in the LaTeX or TeX source file. It formats the information from those entries according to instructions in a bibliography style (.bst) file (specified by the \bibliographystyle command, and it outputs the results to the .bbl file.

The LaTeX reference manual explains what a LaTeX source file must contain to work with BibTeX. Appendix B of the manual describes the format of the .bib files. The BibTeXing document describes extensions and details of this format, and it gives other useful hints for using BibTeX.

Options

--alias=name Pretend to be program name, i.e., set program (and memory dump) name to name. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to name and invoking name.

--disable-installer Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--enable-installer Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--help Give help and exit.

--hhelp This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.

--include-directory=dir Add the directory dir to the head of the list of directories to be searched for input files.

--min-crossrefs=n Defines the minimum number of crossrefs required for automatic inclusion of the crossref'd entry on the citation list; the default is two.

--quiet Suppress all output, except errors.
Programs

`--record-package-usage=file` Record all package usages and write them into `file`.

`--trace=[trace streams]` Enable trace messages. The `trace streams` argument, if specified, is a comma-separated list of trace stream names (Chapter 9, Trace Streams).

`--version` Show version information and exit.

Environment

`BIBINPUTS` Extra paths to locate `.bib` files.

`BSTINPUTS` Extra paths to locate `.bst` files.

`MIKTEX_TRACE` Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

See Also

miktex-latex(1), miktex-tex(1)

Name
miktex-dvicopy — produce modified copy of DVI file

Synopsis
miktex-dvicopy [option...] indvi outdvi

Description
This man page is an adaption of the corresponding TeX Live man page.

miktex-dvicopy reads a DVI file, expands any references to virtual fonts to base fonts, and writes the resulting DVI file. Thus you can use virtual fonts even if your DVI processor does not support them, by passing the documents through miktex-dvicopy first.

Options

--alias=name  
Pretend to be program name, i.e., set program (and memory dump) name to name. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to name and invoking name.

--disable-installer  
Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--enable-installer  
Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--help  
Give help and exit.

--hhelp  
This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.

--include-directory=dir  
Add the directory dir to the head of the list of directories to be searched for input files.

--mag=mag  
Override existing magnification with mag.

--max-pages=n  
Process n pages; default one million. This option cannot be used together with --select.

--page-start=pagespec  
Start at page-spec, for example 2 or 5.*.-2. This option cannot be used together with --select.

--record-package-usages=file  
Record all package usages and write them into file.

--select=sel  
Select pages to be copied.
The syntax for \textit{sel} is: \texttt{start \[ n \]}, where \texttt{start} is the starting page specification (for example 2 or 5.*.-2) and \texttt{n} (optional) is the maximum number of pages to be copied.

You can use up to 10 --select options. This option cannot be used together with --max-pages or --page-start.

\textbf{--trace[=tracestreams]} Enable trace messages. The \textit{tracestreams} argument, if specified, is a comma-separated list of trace stream names (Chapter 9, \textit{Trace Streams}).

\textbf{--version} Show version information and exit.

\section*{Environment}

\textbf{MIKTEX_TRACE} Comma-separated list of trace stream names (see Chapter 9, \textit{Trace Streams}). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.
**Name**

miktex-dvips — convert a DVI file to PostScript

**Synopsis**

miktex-dvips [option...] dvifile

**Description**

This man page is an adaption of the corresponding TeX Live man page.

This man page is obsolete! See the Texinfo documentation instead.

Dvips takes a DVI file produced by TeX (or by some other processor such as miktex-gftodvi) and converts it to PostScript. The DVI file may be specified without the .dvi extension.

**Options**

Many of the parameterless options listed here can be turned off by suffixing the option with a zero (0); for instance, to turn off page reversal, use \texttt{-r0}. Such options are marked with a trailing *.

- \texttt{-} Read additional options from standard input after processing the command line.

- \texttt{--help} Print a usage message and exit.

- \texttt{--version} Print the version number and exit.

- \texttt{-a*} Conserve memory by making three passes over the DVI file instead of two and only loading those characters actually used. Generally only useful on machines with a very limited amount of memory.

- \texttt{-A} Print only the odd pages. This option uses TeX page numbers, not physical page numbers.

- \texttt{-B} Print only the even pages. This option uses TeX page numbers, not physical page numbers.

- \texttt{-b num} Generate \textit{num} copies of each page, but duplicating the page body rather than using the \texttt{/#copies} PostScript variable. This can be useful in conjunction with a header file setting \texttt{bop-hook} to do color separations or other neat tricks.

- \texttt{-C num} Generate \textit{num} copies, but collated (by replicating the data in the PostScript file). Slower than the \texttt{-c} option, but easier on the hands, and faster than resubmitting the same PostScript file multiple times.

- \texttt{-D num} Set both the horizontal and vertical resolution to \textit{num}, given in dpi (dots per inch). This affects the choice of bitmap fonts that are loaded and also
the positioning of letters in resident PostScript fonts. Must be between 10 and 10000. This affects both the horizontal and vertical resolution. If a high resolution (something greater than 400 dpi, say) is selected, the -Z flag should probably also be used. If you are using fonts made with METAfont, such as Computer Modern, makepk needs to know about the value for num that you use or METAfont will fail. See the file modes.mf for a list of resolutions and mode names for most devices.

-e num
Maximum drift in pixels of each character from its “true” resolution-independent position on the page. The default value of this parameter is resolution dependent (it is the number of entries in the list [100, 200, 300, 400, 500, 600, 800, 1000, 1200, 1600, 2000, 2400, 2800, 3200, …] that are less than or equal to the resolution in dots per inch). Allowing individual characters to “drift” from their correctly rounded positions by a few pixels, while regaining the true position at the beginning of each new word, improves the spacing of letters in words.

-E*
Generate an EPSF file with a tight bounding box. This only looks at marks made by characters and rules, not by any included graphics. In addition, it gets the glyph metrics from the TFM file, so characters that print outside their enclosing TFM box may confuse it. In addition, the bounding box might be a bit too loose if the character glyph has significant left or right side bearings. Nonetheless, this option works well enough for creating small EPSF files for equations or tables or the like. (Of course, Dvips output, especially when using bitmap fonts, is resolution-dependent and thus does not make very good EPSF files, especially if the images are to be scaled; use these EPSF files with care.) For multiple page input files, also specify -i to get each page as a separate EPSF file; otherwise, all the pages are overlaid in the single output file.

-f*
Read the DVI file from standard input and write the PostScript to standard output. The standard input must be seekable, so it cannot be a pipe. If your input must be a pipe, write a shell script that copies the pipe output to a temporary file and then points Dvips at this file. It turns off the automatic sending of control-D if it was turned on with the -F option or in the configuration file; use -F after the -f to send it anyway.

-F*
Write control-D (ASCII code 4) as the very last character of the PostScript file. This is useful when Dvips is driving the printer directly instead of working through a spooler, as is common on personal systems. On systems shared by more than one person, this is not recommended.

-G
Shift non-printing characters (ASCII 0-32, 127) to higher-numbered positions. This was useful to work around bugs in old versions of Adobe’s PDF reader. It’s more likely to cause problems nowadays.

-h name
Prepend name as an additional header file, or, if name is -, suppress all header files. Any definitions in the header file get added to the PostScript userdict.

-i*
Make each section be a separate file; a section is a part of the document processed independently, most often created to avoid memory overflow. The filenames are created replacing the suffix of the supplied output file name by a three-digit sequence number. This option is most often used in conjunction with the -S option which sets the maximum section length
in pages; if \texttt{-l} is specified and \texttt{-S} is not, each page is output as a separate file. For instance, some phototypesetters cannot print more than ten or so consecutive pages before running out of steam; these options can be used to automatically split a book into ten-page sections, each to its own file.

On the other hand, if your document uses very large fonts or very large included figures, Dvips might take it upon itself to split the output into unwanted sections, to try to avoid overflowing printer memory.

\texttt{-j*}  Download only needed characters from Type 1 fonts. This is the default. Some debugging flags trace this operation. You can also control partial downloading on a per-font basis (see \texttt{updmap.cfg(5)}).

\texttt{-k*}  Print crop marks. This option increases the paper size (which should be specified, either with a paper size special or with the \texttt{-T} option) by a half inch in each dimension. It translates each page by a quarter inch and draws cross-style crop marks. It is mostly useful with typesetters that can set the page size automatically. This works by downloading \texttt{crop.pro}.

\texttt{-K*}  Remove comments in included PostScript graphics, font files, and headers; only necessary to get around bugs in spoolers or PostScript post-processing programs. Specifically, the \texttt{%%Page} comments, when left in, often cause difficulties. Use of this flag can cause other graphics to fail, however, since the PostScript header macros from some software packages read portion the input stream line by line, searching for a particular comment.

\texttt{-l \[=\] num}  The last page printed will be the first one numbered \texttt{num}. Default is the last page in the document. If \texttt{num} is prefixed by an equals sign, then it (and the argument to the \texttt{-p} option, if specified) is treated as a physical (absolute) page number, rather than a value to compare with the \TeX\ \texttt{\count0} values stored in the DVI file. Thus, using \texttt{-l =9} will end with the ninth page of the document, no matter what the pages are actually numbered.

\texttt{-m*}  Specify manual feed, if supported by the output device.

\texttt{-mode mode}  Use \texttt{mode} as the METAFONT device name for path searching and font generation. This overrides any value from configuration files. With the default paths, explicitly specifying the mode also makes the program assume the fonts are in a subdirectory named mode.

\texttt{-M*}  Turns off automatic font generation.

\texttt{-n num}  Print at most \texttt{num} pages. Default is 100000.

\texttt{-N*}  Turns off generation of structured comments such as \texttt{%%Page}; this may be necessary on some systems that try to interpret PostScript comments in weird ways, or on some PostScript printers. Beware: This also disables page movement, etc., in PostScript viewers such as GSview.

\texttt{-noomega}  Disable the use of Omega extensions when interpreting DVI files. By default, the additional opcodes 129 and 134 are recognized by Dvips as Omega or pTeX extensions and interpreted as requests to set 2-byte characters.
-noptex

Disable the use of pTeX extensions when interpreting DVI files. By default, the additional opcodes 130 and 135 are recognized by Dvips as Omega extensions and interpreted as requests to set 3-byte characters, and 255 as request to change the typesetting direction.

The only drawback is that the virtual font array will (at least temporarily) require 65536 or more positions instead of the default 256 positions, i.e., the memory requirements of Dvips will be somewhat larger. If you find this unacceptable or encounter another problem with the Omega or pTeX extensions, you can switch off the pTeX extension by -noptex, or both by -noomega.

-o name

Send output to the file name. If -o is specified without name, the default is file.ps where the input DVI file was file.dvi. If -o isn't given at all, the configuration file default is used.

If name is -, output goes to standard output. If the first character of name is ! or |, then the remainder will be used as an argument to popen; thus, specifying |lpr as the output file will automatically queue the file for printing as usual. Dvips will print to the local printer device PRN when name is |lpr and a program by that name cannot be found.

-o turns off the automatic sending of control-D. See the -f option for how to override this.

-O x-offset,y-offset

Move the origin by x-offset,y-offset, a comma-separated pair of dimensions such as .1in, -.3cm. The origin of the page is shifted from the default position (of one inch down, one inch to the right from the upper left corner of the paper) by this amount. This is usually best specified in the printer-specific configuration file.

This is useful for a printer that consistently offsets output pages by a certain amount. You can use the file testpage.tex to determine the correct value for your printer. Be sure to do several runs with the same O value-some printers vary widely from run to run.

If your printer offsets every other page consistently, instead of every page, your best recourse is to use bop-hook (see the Dvips manual for more information).

-p [=] num

The first page printed will be the first one numbered num. Default is the first page in the document. If num is prefixed by an equals sign, then it (and the argument to the -l option, if specified) is treated as a physical (absolute) page number, rather than a value to compare with the TeX \count0 values stored in the DVI file. Thus, using -p =3 will start with the third page of the document, no matter what the pages are actually numbered.

-pp first-last

Print pages first through last; equivalent to -p first -l last, except that multiple -pp options accumulate, unlike -p and -l. The - separator can also be :.

-P printer

Read the configuration file config.printer, which can set the output name (most likely o |lpr -Pprinter), resolution, METAFONT mode, and perhaps font paths and other printer-specific defaults. It works best to put sitewide defaults in the one master config.ps file and
only things that vary printer to printer in the config.printer files; config.ps is read before config.printer.

A configuration file for eventual creation of Adobe PDF files is provided in config.pdf and thus can be loaded with -Ppdf. It will try to include Type 1 outline fonts into the PostScript file.

-q* Run quietly. Don't chatter about pages converted, etc. to standard output; report no warnings (only errors) to standard error.

-r* Output pages in reverse order. By default, page 1 is output first.

-R Run securely. -R2 disables both shell command execution in \special (via `) and config files (via the E), pipes as output files, and opening of any absolute or ..-relative filenames. -R1, the default, forbids shell escapes but allows absolute filenames. -R0 allows both.

-s* Enclose the output in a global save/restore pair. This causes the file to not be truly conformant, and is thus not recommended, but is useful if you are driving a deficient printer directly and thus don’t care too much about the portability of the output to other environments.

-S num Set the maximum number of pages in each “section”. This option is most commonly used with the -i option; see its description above for more information.

-t papertype Set the paper type to papertype, usually defined in one of the configuration files, along with the appropriate PostScript code to select it. You can also specify a papertype of landscape, which rotates a document by 90 degrees. To rotate a document whose paper type is not the default, you can use the -t option twice, once for the paper type, and once for landscape.

In general, you should not use any -t option when using a papaersize special, which some LaTeX packages (e.g., hyperref) insert

One exception is when using a nonstandard paper size that is not already defined in config.ps; in this case, you need to specify -t unknown.

Another exception is when producing multi-page files for further processing; use -t nopaper to omit any paper size information in the output. (If you just have a single page document, you can use -E to get pure EPSF output.)

-T hsize,vsize Set the paper size to (hsize,vsize), a comma-separated pair of dimensions such as .1in,-.3cm. It overrides any paper size special in the DVI file. Be careful, as the paper size will stick to a predefined size if there is one close enough. To disable this behavior, use -Tunknown

-u psmapfile Set psmapfile to be the file that Dvips uses for looking up PostScript font aliases. If psmapfile starts with a + character, then the rest of the name is used as the name of the map file, and the map file is appended to the list of map files (instead of replacing the list). In either case, if the name has no extension, .map is added at the end.

-U* Disable a PostScript virtual memory-saving optimization that stores the character metric information in the same string that is used to store the
bitmap information. This is only necessary when driving the Xerox 4045 PostScript interpreter, which has a bug that puts garbage on the bottom of each character. Not recommended unless you must drive this printer.

\(-v\)
Print the Dvips version number and exit.

\(-v*\)
Download non-resident PostScript fonts as bitmaps. This requires use of `makepk` to generate the required bitmap fonts. The bitmap must be put into `psfonts.map` as the downloadable file for that font. This is useful only for those fonts for which you do not have real outlines, being downloaded to printers that have no resident fonts, i.e., very rarely.

\(-x \ num\)
Set the x magnification ratio to \num/1000. Overrides the magnification specified in the DVI file. Must be between 10 and 100000. It is recommended that you use standard magstep values (1095, 1200, 1440, 1728, 2074, 2488, 2986, and so on) to help reduce the total number of PK files generated. \num may be a real number, not an integer, for increased precision.

\(-X \ num\)
Set the horizontal resolution in dots per inch to \num.

\(-y \ num\)
Set the y magnification ratio to \num/1000. See \(-x\) above.

\(-Y \ num\)
Set the vertical resolution in dots per inch to \num.

\(-z*\)
Pass html hyperdvi specials through to the output for eventual distillation into PDF. This is not enabled by default to avoid including the header files unnecessarily, and use of temporary files in creating the output.

\(-Z*\)
Compress bitmap fonts in the output file, thereby reducing the size of what gets downloaded. Especially useful at high resolutions or when very large fonts are used. May slow down printing, especially on early 68000-based PostScript printers. Generally recommend today, and can be enabled in the configuration file.

**Environment**

`MIKTEX_TRACE`
Comma-separated list of trace stream names (see Chapter 9, *Trace Streams*). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

**See Also**

`updmap.cfg(5)`
Name
findtexmf — search files in MiKTeX directories

Synopsis
findtexmf [option...] file...

Description
findtexmf can be used to find files in the MiKTeX directories. When the -file-type option is not given, the search path used when looking for a file is inferred from the name given, by looking for a known extension. If no known extension is found, the search path for TeX source files is used.

Options

--alias=name
Pretend to be name when finding files.

--help
Give help and exit.

--file-type=filetype
Use the specified file type (see below).

--list-file-types
Print known file types.

--must-exist
Allow installation of a packacke, if a file should exist because it is a part of the package.

--show-path=filetype
Print the search path for the specified file type (see below).

--start
Start the associated program, if the file was found.

--the-name-of-the-game=name
Set the name of the engine. Relevant when searching for format files.

--version
Show version information and exit.

File Types

afm (.afm)
base (.base)
bib (.bib)
bst (.bst)
cid maps (.cid;cidmap)
clua (.dll; .so)
cweb (.w)
dvi (.dvi)
enc (.enc)
executables (.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC)
font feature files (.fea)
fmt (.fmt)
hbf (.hbf)
graphic/figure (.eps;.epsi;.png)
gf (.gf)
ist (.ist)
lig files (.lig)
lua (.lua;.luatex;.luc;.luctex;.texlua;.texluc;.tlu)
map (.map)
mem (.mem)
mf (.mf)
mfpool (.pool)
mft (.mft)
mlbib (.mlbib;.bib)
mlbst (.bst)
mp (.mp)
mppool (.pool)
ocp (.ocp)
ofm (.ofm;.tfm)
opl (.opl)
opentype fonts (.otf)
otp (.otp)
ovf (.ovf)
ovp (.ovp)
pk (.pk)
PostScript header (.pro;.enc)
subfont definition files (.sfd)
tcx (.tcx)
tex (.tex)
texpool (.pool)
TeX system documentation (.pdf;.html;.md;.txt;.ps;.dvi)
tfm (.tfm)
trueType fonts (.ttf;.ttc)
type1 fonts (.pfb;.pfa)
type42 fonts (.t42;.T42)
vf (.vf)
web (.web)
Name

miktex-gftodvi — make proof sheets from generic font files

Synopsis

miktex-gftodvi [option...] [gffile]

Description

This man page is an adaptation of the corresponding TeX Live man page.

The miktex-gftodvi program converts a generic font (GF) file output by, for example, METAFONT, to a device independent (DVI) file (that can then be typeset using the same software that has already been written for). The characters in the GF file will appear one per page, with labels, titles, and annotations as specified in Appendix H (Hardcopy Proofs) of The METAFONTbook.

miktex-gftodvi uses other fonts in addition to the main GF file. A “gray” font is used to typeset the pixels that actually make up the character. (We wouldn't want all the pixels to be simply black, since then labels, key points, and other information would be lost.) A “title” font is used for the information at the top of the page. A “label” font is used for the labels on key points of the figure. A “slant” font is used to typeset diagonal lines, which otherwise have to be simulated using horizontal and vertical rules. The default gray, title, and label fonts are gray, cmr8, and cmtt10, respectively; there is no default slant font.

To change the default fonts, you can give special commands in your source file, or you can change the fonts on the command-line.

The GF file name on the command-line must be complete. Because the resolution is part of the extension, it would not make sense to append a default extension as is done with other DVI-reading software. The output file name defaults to the same root as the GF file, with the .dvi extension added. For example, the input file cmr10.2602gf would become cmr10.dvi.

Options

--alias=name Pretend to be program name, i.e., set program (and memory dump) name to name. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to name and invoking name.

--disable-installer Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--enable-installer Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--gray-font=font Sets the “gray” font. Default is gray.

--help Give help and exit.

--hhelp This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.
--include-directory=dir
Add the directory dir to the head of the list of directories to be searched for input files.

--label-font=font
Sets the “label” font. Default is cmtt10.

--logo-font=font
Sets the “logo” font. Default is logo8.

--overflow-label-offset=real
Specifies the distance from the right edge of the character bounding box at which the overflow equations (if any) are typeset. The value is given in points. The default is a little over two inches.

--record-package- usages=file
Record all package usages and write them into file.

-slant-font=font
Sets the “slant” font. There is no default.

title-font=font
Sets the “title” font. Default is cmr8.

--trace[]=tracestreams
Enable trace messages. The tracestreams argument, if specified, is a comma-separated list of trace stream names (Chapter 9, Trace Streams).

--version
Show version information and exit.

Environment

MIKTEX_TRACE
Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

See also
miktex-mf(1)

Name
initexmf — MiKTeX configuration utility

Synopsis

initexmf [option...]

Description

initexmf can be used to configure MiKTeX.

User mode vs. administrator mode

This utility can be run in two modes:

User mode (default) MiKTeX operates on user-scoped configuration and data files.

Administrator mode Only system-wide MiKTeX configuration and data files are modified, assuming that the MiKTeX setup is shared by all users. The utility must be run with administrator privileges.

By default, the utility runs in user mode. You can turn on administrator mode with the --admin option. For example, if you want to update the system-wide file name database, you invoke the utility as follows:

> initexmf --admin --update-fndb

Options

--admin Run in administrator mode:
- Operate on the system-wide MiKTeX configuration data store.
- Install packages for all users.

Using this option requires a shared MiKTeX setup, i.e., MiKTeX must have been set up for all users. The program must be run with administrator privileges.

--default-paper-size=paper

--disable-installer Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--dump

--dump=name Dump the specified memory dump file.

--edit-config-file=configfile Open the specified config file in a text editor.

--enable-installer Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--engine=engine Engine to be used when generating format files.

--force

29
Programs

--list-formats

List all known formats.

--list-modes

--mklangs

Create language.dat, language.dat.lua and language.def.

--mklinks

Create all possible links.

--mklinks=category

Create links for the specified category, which must be one of:

- formats Create links from format names to TeX engines. For example, latex.exe will become a link to miktex-tex.exe and invoking latex.exe will cause TeX to load the format file latex.fmt.

- miktex Create links from standard program names to miktex-executables. For example, tex.exe will become a link to miktex-tex.exe.

- scripts Create executable links from script names to script wrappers. For example, latexmk.exe will become a link to the internal Perl wrapper runperl.exe and invoking latexmk.exe will eventually execute the Perl script latexmk.pl.

--mkmaps

--print-only

Print what would be done. Nothing is changed.

--quiet

Suppress screen output.

--register-root=dir

--remove-links

Remove the links which were created by --mklinks.

--report

Write a MiKTeX configuration report.

--set-config-value=[section]valuename=value

Set a value to be stored in the MiKTeX configuration data store (see miktex.ini(5)).

--show-config-value=[section]valuename

Print a value from the MiKTeX configuration data store (see miktex.ini(5)).

--unregister-root=dir

--update-fndb

Refresh the file name database for a specific TEXMF tree.

--update-fndb=dir

Register user root directories.
Programs

--verbose
Print information on what is being done.

--version
Print the version number and exit.

See also
miktex.ini(5)
Name

miktex-luatex — an extended version of pdfTeX using Lua as an embedded scripting language

Synopsis

miktex-luatex [option...] [[command...]] [file]

Description

This man page is an adaption of the corresponding TeX Live man page.

Run the LuaTeX typesetter on file, usually creating file.pdf. Any remaining commands are processed as LuaTeX input, after file 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 format 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 miktex-texlua it acts as Lua interpreter. If called as miktex-texluac it acts as Lua bytecode compiler.

LuaTeX is an extended version of pdfTeX with Unicode and OpenType font support, embedded Lua scripting language, the eTeX 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 the LuaTeX manual using the MiKTeX Help Utility (mthelp 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 miktex-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. But if the option is present, LuaTeX will enter an alternative mode of commandline parsing in comparison to the standard MiKTeX programs. The presence of --lua makes most of other options unreliable, because the Lua initialization file can disable path searching and/or hook functions into various callbacks.

--lua=file  Load and execute a Lua initialization script.

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.
--luaonly  Start LuaTeX as a Lua byte compiler. In this mode, LuaTeX is exactly like \texttt{luac} from the standalone Lua distribution, except that it does not have the \texttt{#1} switch, and that it accepts (but ignores) the \texttt{#luaonly} switch.

Then the regular options:

--alias=name  Pretend to be program \texttt{name}, i.e., set program (and memory dump) name to \texttt{name}. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to \texttt{name} and invoking \texttt{name}.

--aux-directory=dir  Set \texttt{dir} as the directory to which auxiliary files are written. Also look for input files in \texttt{dir} first, before along the normal search path.

--c-style-errors  Change the way, error messages are printed. The alternate style looks like error messages from many compilers and is easier to parse for some editors.

--credits  Display credits and exit.

--debug-format  Enable format debugging.

--disable-installer  Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--disable-write18

--draftmode  Switch on draft mode. LuaTeX; doesn't write a PDF and doesn't read any included images, thus speeding up execution.

--enable-installer  Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--enable-write18

--halt-on-error  Quit after the first error.

--help  Give help and exit.

--include-directory=dir  Add the directory \texttt{dir} to the head of the list of directories to be searched for input files.

--initialize  Become the \texttt{INI} variant of the program.

--interaction=mode  Set the interaction mode. Must be one of \texttt{batchmode}, \texttt{nonstopmode}, \texttt{scrollmode} and \texttt{errorstopmode}. The meaning of these modes is the same as the corresponding commands.

--job-name=name  Set the name of the job (\jobname). This has an affect on the output file names.

--mktex=fmt  Enable \texttt{fmt} generation, where \texttt{fmt} must be either \texttt{tex} or \texttt{tfm}.

--no-c-style-errors  Don't change the way, error messages are printed.

--no-mktex=fmt  Disable \texttt{fmt} generation, where \texttt{fmt} must be either \texttt{tex} or \texttt{tfm}.
Programs

--nosocket
Disable the Lua socket library.

--output-comment=string
Use string for DVI file comment instead of date.

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

--output-format=format
Use format for job output (one of: dvi, pdf).

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

--restrict-write18

--safer
Disable easily exploitable Lua commands.

--synctex=n
Generate SyncTeX data for previewers. If n is zero, no .synctex file is created. If n is negative, the .synctex file is a text file. If n is positive, the .synctex file is compressed with gzip and the .gz file name extension is added.

Furthermore, n is interpreted as a bit field:

Bit 1 (n AND 2)  Don't add the .gz file name extension.

Bit 2 (n AND 4)  Activate form support.

Bit 3 (n AND 8)  Activate better compression.

--undump=name
Use name as the name of the format to be used, instead of the name by which the program was called or a

%&

line.

--utc
Init time to UTC.

--version
Show version information and exit.

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.

Environment

MIKTEX_EDITOR
The editor to use when selecting ● in the error prompt menu.

The value can contain these placeholders:

%f  The name of the file, which contains the erroneous line of TeX code.
The line number.

Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

See Also
miktex-epstopdf(1), miktex-pdftex(1)
Name
miktex-mf — METAFONT, a language for font and logo design

Synopsis
miktex-mf [option...][command...] [file]

Description
This man page is an adaption of the corresponding TeX Live man page.

METAFONT reads the program in the specified files and outputs font rasters (in GF format) and font metrics (in TFM format). The METAFONT language is described in The METAFONTbook.

Like TeX, METAFONT is normally used with a large body of precompiled macros, and font generation in particular requires the support of several macro files. This version of METAFONT looks at its command line to see what name it was called under. Both inimf and virmf are links to the miktex-mf executable. When called as inimf (or when the --initialize option is given) it can be used to precompile macros into a .base file. When called as virmf it will use the plain base. When called under any other name, METAFONT will use that name as the name of the base to use. For example, when called as miktex-mf the mf base is used, which is identical to the plain base. Other bases than plain are rarely used.

The commands given on the command line to the METAFONT program are passed to it as the first input line. (But it is often easier to type extended arguments as the first input line, since shells tend to gobble up or misinterpret METAFONT's favorite symbols, like semicolons, unless you quote them.) As described in The METAFONTbook, that first line should begin with a filename, a \controlsequence, or a &basename.

The normal usage is to say miktex-mf \mode=printengine; input font to start processing font.mf. (Or you can just say miktex-mf and give the other stuff on the next line.) Other control sequences, such as batchmode (for silent operation) can also appear. The name font will be the “jobname”, and is used in forming output file names. If METAFONT doesn't get a file name in the first line, the job name is mfput. The default extension, .mf, can be overridden by specifying an extension explicitly.

A log of error messages goes into the file jobname.log. The output files are jobname.tfm and jobname.numbergf, where number depends on the resolution and magnification of the font. The mode in this example is shown generically as printengine, a symbolic term for which the name of an actual device or, most commonly, the name localfont (see below) must be substituted. If the mode is not specified or is not valid, METAFONT will default to proof mode which produces large character images for use in font design and refinement. Proof mode can be recognized by the suffix .2602gf after the job name. Examples of proof mode output can be found in Computer Modern Typefaces (Volume E of Computers and Typesetting). The system of magsteps is identical to the system used by TeX, with values generally in the range 0.5, 1.0, 2.0, 3.0, 4.0 and 5.0.

Magnification can also be specified not as a magstep but as an arbitrary value, such as 1.315, to create special character sizes.

Before font production can begin, it is necessary to set up the appropriate base files. The minimum set of components for font production for a given printengine is the plain.mf macro file and the local mode_def file. The macros in plain.mf can be studied in an appendix to The METAFONTbook; they were developed by Donald E. Knuth, and this file should never be altered except when it is officially upgraded. Each mode_def specification helps adapt fonts to a particular printengine. The local ones in use on this computer should be in modes.mf.
The response to METAFONT’s error-recovery prompt causes the default editor to start up at the current line of the current file. The configuration value `[Core]Editor` can be used to change the editor used. It may contain a string with `%f` indicating where the file name goes and `%l` indicating where the decimal line number (if any) goes. For example, an `[Core]Editor` string for `emacs` can be set with the command

```
> initexmf --set-config-value="[Core]Editor=emacs +%l%f"
```

A convenient file is `null.mf`, containing nothing. When METAFONT can't find the file it thinks you want to input, it keeps asking you for another file name; responding `null` gets you out of the loop if you don't want to input anything.

## Online Graphics Output

You can see METAFONT’s output without printing. Chapter 23 of *The METAFONTbook* describes what you can do. You enable screen output by giving `--screen` on the command-line.

## Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--alias=name</code></td>
<td>Pretend to be program <code>name</code>, i.e., set program (and memory dump) name to <code>name</code>. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to <code>name</code> and invoking <code>name</code>.</td>
</tr>
<tr>
<td><code>--aux-directory=dir</code></td>
<td>Set <code>dir</code> as the directory to which auxiliary files are written. Also look for input files in <code>dir</code> first, before along the normal search path.</td>
</tr>
<tr>
<td><code>--bistack-size=n</code></td>
<td>Set the size of the stack for bisection algorithms.</td>
</tr>
<tr>
<td><code>--buf-size=n</code></td>
<td>Set the maximum number of characters simultaneously present in current lines of open files and in control sequences between <code>\csname</code> and <code>\endcsname</code>. TeX uses the buffer to contain input lines, but macro expansion works by writing material into the buffer and reparsing the line. As a consequence, certain constructs require the buffer to be very large, even though most documents can be handled with a small value.</td>
</tr>
<tr>
<td><code>--c-style-errors</code></td>
<td>Change the way, error messages are printed. The alternate style looks like error messages from many compilers and is easier to parse for some editors.</td>
</tr>
<tr>
<td><code>--disable-installer</code></td>
<td>Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.</td>
</tr>
<tr>
<td><code>--dont-parse-first-line</code></td>
<td>Disable checking whether the first line of the main input file starts with <code>%&amp;</code>.</td>
</tr>
<tr>
<td><code>--enable-installer</code></td>
<td>Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.</td>
</tr>
<tr>
<td><code>--error-line=n</code></td>
<td>Set the width of context lines on terminal error messages.</td>
</tr>
<tr>
<td><code>--half-error-line=n</code></td>
<td>Set the width of first lines of contexts in terminal error messages.</td>
</tr>
<tr>
<td><code>--halt-on-error</code></td>
<td>Quit after the first error.</td>
</tr>
<tr>
<td><code>--help</code></td>
<td></td>
</tr>
</tbody>
</table>
--hhelp

Give help and exit.

This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.

--include-directory=dir

Add the directory dir to the head of the list of directories to be searched for input files.

--initialize

Become the INI variant of the program.

--interaction=mode

Set the interaction mode. Must be one of batchmode, nonstopmode, scrollmode and errorstopmode. The meaning of these modes is the same as the corresponding commands.

--job-name=name

Set the name of the job (jobname). This has an affect on the output file names.

--job-time=file

Set the time-stamp of all output files equal to file's time-stamp.

--lig-table-size=n

Set the maximum number of ligature/kern steps. Must be at least 255 and at most 32510.

--main-memory=n

Change the total size (in memory words) of the main memory array. Relevant only while creating memory dump files.

--max-print-line=n

Set the width of longest text lines output; should be at least 60.

--max-strings=n

Set the maximum number of strings.

--max-wiggle=n

Set the number of autorounded points per cycle.

--move-size=n

Set the space for storing moves in a single octant.

--no-c-style-errors

Don't change the way, error messages are printed.

--output-directory=dir

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

--param-size=n

Set the maximum number of simultaneous macro parameters.

--parse-first-line

Check whether the first line of the main input file starts with %&, and parse if it does. This can be used to specify extra command-line options.

--path-size=n

Set the maximum number of knots between breakpoints of a path.

--pool-size=n

Set the maximum number of characters in strings, including all error messages and help texts, and the names of all fonts and control sequences.

--quiet

Suppress all output, except errors.

--record-package-usages=file

Record all package usages and write them into file.
Programs

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

--screen
Enable screen output.

--stack-size=n
Set the maximum number of simultaneous input sources.

--string-vacancies=n
Set the minimum number of characters that should be available for the user's control sequences and font names, after the compiler's own error messages are stored. Must be at least 25000 less than pool_size, but doesn't need to be nearly that large.

--tcx=tcxname

--time-statistics
Show processing time statistics.

--trace[=tracestreams]
Enable trace messages. The tracestreams argument, if specified, is a comma-separated list of trace stream names (Chapter 9, Trace Streams).

--undump=name
Use name as the name of the format to be used, instead of the name by which the program was called or a

%&
line.

--version
Show version information and exit.

Environment

MFINPUTS
Extra paths to locate METAFONT input and openin files.

MIKTEX_EDITOR
The editor to use when selecting in the error prompt menu.

The value can contain these placeholders:

%f The name of the file, which contains the erroneous line of TeX code.
%l The line number.

MIKTEX_TRACE
Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

See also

Name
miktexsetup — MiKTeX setup utility

Synopsis
miktexsetup [options] download
miktexsetup [options] install
miktexsetup [options] uninstall

Description
The MiKTeX Setup Utility is used to download, install and uninstall MiKTeX. This utility is the command line counterpart of the MiKTeX Setup Wizard. It is suitable for unattended setup tasks.

The download task creates a local package repository which is a mirror of the remote package repository. It is possible to run the task on a regular basis in order to maintain an up-to-date package repository.

The install task installs MiKTeX from the local package repository.

The uninstall task removes MiKTeX.

Options
--common-config=dir Set the location of the common configuration directory. This option requires administrator privileges.
--common-data=dir Set the location of the common data directory. This option requires administrator privileges.
--common-install=dir Set the common installation directory. This option requires administrator privileges.
--common-link-target-directory=dir Set the system-wide directory in which to create symbolic links to MiKTeX executables.
--common-roots=dirs Register additional directories for all users. dirs must be a semicolon-separated list of fully qualified path names. This option requires administrator privileges.

Environment variables (<VARNAME>) can be used.
--list-repositories Download the list of known package repository URLs, then print the list.
--local-package-repository=dir Download into (install from) the specified directory.
--modify-path Add MiKTeX to PATH.
--modify-path=no Don't add MiKTeX to PATH.
<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--modify-path</td>
<td>Add MiKTeX to PATH.</td>
</tr>
<tr>
<td>--package-set=set</td>
<td>Download/Install the specified package set. This must be one of essential, basic, complete.</td>
</tr>
<tr>
<td>--portable=dir</td>
<td>Setup MiKTeX for use on a portable device.</td>
</tr>
<tr>
<td>--print-info-only</td>
<td>Print information about what would be done then exit.</td>
</tr>
<tr>
<td>--program-folder=name</td>
<td>Add shortcuts to the specified program folder.</td>
</tr>
<tr>
<td>--quiet</td>
<td>Suppress all output, except errors.</td>
</tr>
<tr>
<td>--remote-package-repository=url</td>
<td>Download from the specified URL. Use --list-repositories to download an up-to-date list of possible repositories.</td>
</tr>
<tr>
<td>--shared</td>
<td>Run the task for all users. This option requires administrator privileges.</td>
</tr>
<tr>
<td>--shared=no</td>
<td>Run the task for current user only.</td>
</tr>
<tr>
<td>--trace=[], tracestreams</td>
<td>Enable trace messages. The tracestreams argument, if specified, is a comma-separated list of trace stream names (Chapter 9, Trace Streams).</td>
</tr>
<tr>
<td>--use-registry</td>
<td>Write configuration settings into the Windows registry.</td>
</tr>
<tr>
<td>--use-registry=no</td>
<td>Don't write configuration settings into the Windows registry. Use configuration files instead.</td>
</tr>
<tr>
<td>--user-config=dir</td>
<td>Set the location of the configuration directory for the current user.</td>
</tr>
<tr>
<td></td>
<td>Environment variables (&lt;VARNAME&gt;) can be used. See the example below.</td>
</tr>
<tr>
<td>--user-data=dir</td>
<td>Set the location of the data directory for the current user.</td>
</tr>
<tr>
<td></td>
<td>Environment variables (&lt;VARNAME&gt;) can be used. See the example below.</td>
</tr>
<tr>
<td>--user-install=dir</td>
<td>Set the user installation directory.</td>
</tr>
<tr>
<td></td>
<td>Environment variables (&lt;VARNAME&gt;) can be used. See the example below.</td>
</tr>
<tr>
<td>--user-link-target-directory=dir</td>
<td>Set the per-user directory in which to create symbolic links to MiKTeX executables.</td>
</tr>
<tr>
<td>--user-roots=dirs</td>
<td>Register additional directories for the current user. dirs must be a semicolon-separated list of fully qualified path names.</td>
</tr>
<tr>
<td></td>
<td>Environment variables (&lt;VARNAME&gt;) can be used.</td>
</tr>
<tr>
<td>--verbose</td>
<td>Turn on verbose output mode.</td>
</tr>
<tr>
<td>--version</td>
<td>Show version information and exit.</td>
</tr>
</tbody>
</table>
Examples

Downloading

The first task is to download MiKTeX into a local package repository:

```
> miktexsetup
   --verbose
   --local-package-repository=C:\miktex-repository
   --package-set=complete
   download
```

This command will create a local package repository in `C:\miktex-repository`.

It is possible to interrupt (Control+C) this operation at anytime and resume it later by running the same command again.

Installing for all users

In this example, MiKTeX is installed for all users from the local package repository `C:\miktex-repository`. User directories are specified by using environment variables (VARNAME) which are expanded at run-time (delayed expansion).

You can first specify `--print-info-only` in order to perform a dry run:

```
> miktexsetup
   --verbose
   --local-package-repository=C:\miktex-repository
   --shared
   --user-config="<APPDATA>\MiKTeX\2.9"
   --user-data="<LOCALAPPDATA>\MiKTeX\2.9"
   --user-install=<APPDATA>\MiKTeX\2.9
   --print-info-only
   install
```

Uninstalling

MiKTeX can be removed by selecting the `uninstall` task. `--shared` should be specified, if MiKTeX is installed for all users.

```
> miktexsetup --verbose --shared uninstall
```
Name

mpm — MiKTeX package manager

Synopsis

mpm [option...]

Description

MPM (MiKTeX Package Manager) is used to install packages from a MiKTeX package repository.

MPM starts in windowed mode, if you do not specify any command-line options.

User mode vs. administrator mode

This utility can be run in two modes:

User mode (default) MPM operates on the user installation directory (usually %LOCALAPPDATA%\Programs\MiKTeX 2.9).

Administrator mode MPM operates on the system-wide installation directory (usually C:\Program Files\MiKTeX 2.9), assuming that the MiKTeX setup is shared by all users. MPM must be run with administrator privileges.

By default, MPM runs in user mode. You can turn on administrator mode with the --admin option. For example, if you want to install a package for all users, you invoke MPM as follows:

> mpm --admin --install=a0poster

Options

--admin Run in administrator mode:

• Operate on the system-wide MiKTeX configuration data store.

• Install packages for all users.

Using this option requires a shared MiKTeX setup, i.e., MiKTeX must have been set up for all users. The program must be run with administrator privileges.

--find-updates Check the package repository for updates, then print the list of updateable packages.

--find-upgrades Search for packages that must be installed in order to complete the MiKTeX setup (can be used in conjunction with --package-level). Then print the package list.

--help Give help and exit.

--hhelp
This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.

--import=package  
Import the specified package from another MiKTeX installation. The root directory must be specified via --repository=dir.

--import-all  
Import all packages from another MiKTeX installation. The root directory must be specified via --repository=dir.

--install=packagelist  
Install the specified packages.

--install=@listfile  
Install packages. Package names are read from listfile.

--list  
List the contents of the package database: for each package, print the installation status, the number of files, the size, and the name.

--list-package-names  
List the package names.

--list-repositories  
Download the list of known package repository URLs, then print the list.

--max-count=num  
Stop after num packages.

--package-level=level  
Use the specified package level (to be used in conjunction with --find-upgrades and --upgrade).

level must be one of:

- essential  Includes mandatory packages.
- basic  Includes popular packages.
- complete  Includes all available packages.

--pick-repository-url  
Pick up a suitable URL from the package repository list and print it.

--print-package-info=package  
Print detailed information about the specified package.

--quiet  
Suppress all output, except errors.

--require=packagelist  
Install packages which are not already installed.

--require=@listfile  
Install packages which are not already installed. Package names are read from listfile.

--repository=location  
Use the specified location as the package repository. The location can be either a fully qualified path name (a local package
repository) or an URL (a remote package repository). You can use the `--list-repositories` to retrieve a list of working package repository URLs.

`--repository-release-state=state` Select the release state of the remote package repository. The release state is relevant for finding appropriate package repositories (`--list-repositories`, `--pick-repository-url`). The release state must be one of `stable` or `next`.

`--reverse` Reverse the result of comparisons (when listing packages).

`--set-repository=location` Store the location of the default package repository in the MiKTeX configuration data store. The location can be either a fully qualified path name (a local package repository) or an URL (a remote package repository).

`--trace[=tracestreams]` Enable trace messages. The `tracestreams` argument, if specified, is a comma-separated list of trace stream names (Chapter 9, Trace Streams).

`--uninstall=package` Uninstall the specified package.

`--update` Update all installed packages.

`--update=packagelist` Update the specified packages.

`--update=@listfile` Update packages. Package names are read from `listfile`.

`--update-db` Synchronize the local package database with the package repository.

`--upgrade` Upgrade the MiKTeX setup to a package level (can be used in conjunction with `--package-level`). This will install all the missing packages.

`--verify` Verify the integrity of all installed packages.

`--verify=packagelist` Verify the integrity of the specified packages.

`--verify=@listfile` Verify the integrity packages. Package names are read from `listfile`.

`--verbose` Turn on verbose output mode.

`--version` Show version information and exit.
Package Database

All package information is retrieved from the package database, which must have been properly installed by running MPM with the `--update-db` option.

Examples

Print the list of known package repository URLs:

```bash
> mpm --list-repositories
```

Retrieve the package database files from the server `some.server` (assuming this a registered URL):

```bash
> mpm --repository=ftp://some.server/miktex/packages/ \
   --verbose --update-db
```

Print information about package `a0poster`:

```bash
> mpm --print-package-info a0poster
```

Install package `a0poster`:

```bash
> mpm --verbose --install a0poster
```

Update all installed packages:

```bash
> mpm --verbose --update
```

Print the list of installed packages:

```bash
> mpm --list | grep ^i
```

Upgrade the MiKTeX setup to the highest level:

```bash
> mpm --verbose --package-level=complete --upgrade
```

Environment

**MIKTEX_REPOSITORY** Location of the default package repository. This can be either a fully qualified path name (a local package repository) or an URL (a remote package repository).

**MIKTEX_TRACE** Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

**http_proxy** The proxy server to be used for HTTP.

**FTP_PROXY** The proxy server to be used for .

**ALL_PROXY** The proxy server to be used, if no protocol-specific proxy is set.

**NO_PROXY** Comma-separated list of host names that should not go through any proxy.
Name

miktex-mpost — MetaPost, a system for creating graphics

Synopsis

miktex-mpost [option...][[command...][file]]

Description

This man page is an adaption of the corresponding TeX Live man page.

MetaPost interprets the MetaPost language and produces PostScript (EPS) or Scalable Vector Graphics (SVG) pictures. The MetaPost language is similar to Knuth's METAFONT with additional features for including TeX commands and accessing features of PostScript not found in METAFONT.

MetaPost is normally used with some preloaded macros, and it will use its executable name as the name of the preload file to use. For example, when called as miktex-mpost the mpost.mp file is used, which is identical to plain.mp. When the --initialize option is given, preloading does not happen.

The commands given on the command line to the MetaPost program are passed to it as the first input line. (But it is often easier to type extended arguments as the first input line, since shells tend to gobble up or misinterpret MetaPost's favorite symbols, like semicolons, unless you quote them.) The normal usage is to say miktex-mpost figs to process the file figs.mp. The basename of figs becomes the "jobname", and is used in forming output file names. If no file is named, the jobname becomes mpout. The default extension, .mp, can be overridden by specifying an extension explicitly.

When the --dvitomp option is given, MetaPost acts as DVI-to-MPX converter only. Run miktex-mpost --dvitomp --help for option explanation.

Options

--alias=name Pretend to be program name, i.e., set program (and memory dump) name to name. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to name and invoking name.

--c-style-errors Change the way, error messages are printed. The alternate style looks like error messages from many compilers and is easier to parse for some editors.

--debug Print debugging info and leave temporary files in place.

--halt-on-error Quit after the first error.

--help Give help and exit.

--initialize Become the INI variant of the program.

--interaction=mode Set the interaction mode. Must be one of batchmode, nonstopmode, scrollmode and errorstopmode. The meaning of these modes is the same as the corresponding commands.

--job-name=name Set the name of the job (jobname). This has an affect on the output file names.
Programs

--no-c-style-errors  Don't change the way, error messages are printed.
--numbersystem=string  Set number system mode (string one of: scaled, double, binary, decimal).
--output-directory=dir  Write output files in dir. instead of the current directory. Look up input files in dir first, then along the normal search path.
--recorder  Enable the file name recorder. This leaves a trace of the files opened for input and output in a file with the extension .fls.
--restricted  Be secure: disable tex, makempx and editor commands.
-s internal="string"  Set internal to the string value.
-s internal=number  Set internal to the number value.
--tex=texprogram  Use texprogram instead of miktex-tex when compiling text labels. This flag overrides the environment variable TEX.
--troff  Set prologues:=1 and assume --tex=troff.
--undump=name  Use name as the name of the format to be used, instead of the name by which the program was called or a %& line.
--version  Show version information and exit.

Environment

MIKTEX_TRACE  Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

MPINPUTS  Extra paths to locate MetaPost input files.
Name

mthelp — MiKTeX help utility

Synopsis

mthelp [option...][name...]

Description

mthelp is a utility to look up MiKTeX related documentation.

mthelp creates an HTML page which contains a short description of the package together with links to all documentation files. An HTML viewer is started to view the page.

You can use the --view to bypass the intermediate HTML file.

name should be the name of a package in the TeX distribution.

Options

--list-only  List documentation files, but do not start a viewer.

--print-only  Print the command that would be executed to view the documentation, but do not start the command.

--quiet  Suppress all output, except errors.

--version  Show version information and exit.

--view  Open the main documentation file in a viewer.

Environment

MIKTEX_TRACE  Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

MIKTEX_VIEW_dvi  DVI viewer.

MIKTEX_VIEW_pdf  PDF viewer.

MIKTEX_VIEW_ps  PostScript viewer.

MIKTEX_VIEW_html  HTML viewer.

MIKTEX_VIEW_txt  Text viewer.

The environment variables should be set with a “%f” as a placeholder for the name of the file. For example:

> MIKTEX_VIEW_pdf="gv %f"
Files

The intermediate HTML file *(package.html)* is stored in the directory `miktex/mthelp` relative to the data TEXMF data root (usually `%LOCALAPPDATA%\MiKTeX\2.9`).

See also

MiKTeX Project Page [https://miktex.org/]

Environment

*MIKTEX_TRACE*  
Comma-separated list of trace stream names (see Chapter 9, *Trace Streams*). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.
Name

mtprint — MiKTeX print utility

Synopsis

mtprint [option...] file...

Description

mtprint sends TeX output files to a printing device.

Options

--even-only
Prints only even TeX pages.

--landscape
Selects landscape output format.

--odd-only
Prints only odd TeX pages.

--page-range=range
Selects a TeX page range (e.g., 20–21). Multiple --page-range options accumulate

--print-method=method
Selects a print method. One of

    psbmp  This method uses Dvips and Ghostscript to produce the print output.

    ps     This method uses Dvips to produce an intermediate PostScript file which will be sent to the printer. This only works for PostScript printers.

--print-nothing
Simulates printing.

--printer=printer
Selects a printing device. The default printer is used, if this option is omitted.
**Name**

miktex-pdftex — DVI/PDF output from TeX

**Synopsis**

miktex-pdftex [option...] [file] [command...]

**Description**

This man page is an adaption of the corresponding TeX Live man page.

Run the pdfTeX typesetter on `file`, usually creating `file.pdf`. If the file argument has no extension, `.tex` will be appended to it. Instead of a file name, a set of pdfTeX commands can be given, the first of which must start with a backslash. With a `&format` argument pdfTeX uses a different set of precompiled commands, contained in `format.fmt`; it is usually better to use the `--undump=format` option instead.

pdfTeX is a version of TeX, with the eTeX extensions, that can create PDF files as well as DVI files.

In DVI mode, pdfTeX can be used as a complete replacement for the TeX engine.

The typical use of pdfTeX is with a pregenerated formats for which PDF output has been enabled. The `miktex-pdftex` command uses the equivalent of the plain TeX format, and the `miktex-pdflatex` command uses the equivalent of the LaTeX format. To generate formats, use the `--initialize` switch.

In PDF mode, pdfTeX can natively handle the PDF, JPG, JBIG2 and PNG graphics formats. pdfTeX cannot include PostScript or Encapsulated PostScript (EPS) graphics files; first convert them to PDF using `miktex-epstopdf(1)`. pdfTeX’s handling of its command-line arguments is similar to that of the other TeX programs in the MiKTeX implementation.

**Options**

- `--alias=name` Pretend to be program `name`, i.e., set program (and memory dump) name to `name`. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to `name` and invoking `name`.

- `--aux-directory=dir` Set `dir` as the directory to which auxiliary files are written. Also look for input files in `dir` first, before along the normal search path.

- `--buf-size=n` Set the the maximum number of characters simultaneously present in current lines of open files and in control sequences between \csname and \endcsname. TeX uses the buffer to contain input lines, but macro expansion works by writing material into the buffer and reparsing the line. As a consequence, certain constructs require the buffer to be very large, even though most documents can be handled with a small value.

- `--c-style-errors` Change the way, error messages are printed. The alternate style looks like error messages from many compilers and is easier to parse for some editors.

- `--disable-8bit-chars` Make only 7-bit characters printable.
Programs

--disable-installer  Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--disable-pipes  Disable input (output) from (to) child processes.

--disable-write18

--dont-parse-first-line  Disable checking whether the first line of the main input file starts with %&.

--draftmode  Sets \pdftexdraftmode so pdfTeX doesn't write a PDF and doesn't read any included images, thus speeding up execution.

--enable-8bit-chars  Make all characters printable.

--enable-enctex  Enable encTeX extensions such as \texttt{\textbackslash mubyte}.

--enable-etex  Enable eTeX extensions.

--enable-installer  Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--enable-mltex  Enable MLTeX extensions such as \texttt{\textbackslash charsudedef}.

--enable-pipes  Enable input (output) from (to) child processes.

--enable-write18

--error-line=n  Set the width of context lines on terminal error messages.

--extra-mem-bot=n  Set the extra size (in memory words) for large data structures like boxes, glue, breakpoints, et al. Relevant only after the memory dump file has been read.

--extra-mem-top=n  Set the extra size (in memory words) for chars, tokens, et al. Relevant only after the memory dump file has been read.

--font-max=n  Set the maximum internal font number.

--font-mem-size=n  Set the size, in TeX memory words, of the font memory.

--half-error-line=n  Set the width of first lines of contexts in terminal error messages.

--halt-on-error  Quit after the first error.

--hash-extra=n  Set the extra space for the hash table of control sequences (which allows 10K names as distributed).

--help  Give help and exit.

--hhelp  This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.

--include-directory=dir  Add the directory \texttt{dir} to the head of the list of directories to be searched for input files.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--initialize</td>
<td>Become the INI variant of the program.</td>
</tr>
<tr>
<td>--interaction=mode</td>
<td>Set the interaction mode. Must be one of batchmode, nonstopmode, scrollmode and errorstopmode. The meaning of these modes is the same as the corresponding commands.</td>
</tr>
<tr>
<td>--job-name=name</td>
<td>Set the name of the job (\jobname). This has an effect on the output file names.</td>
</tr>
<tr>
<td>--job-time=file</td>
<td>Set the time-stamp of all output files equal to file's time-stamp.</td>
</tr>
<tr>
<td>--main-memory=n</td>
<td>Change the total size (in memory words) of the main memory array. Relevant only while creating memory dump files.</td>
</tr>
<tr>
<td>--max-in-open=n</td>
<td>Set the maximum number of input files and error insertions that can be going on simultaneously.</td>
</tr>
<tr>
<td>--max-print-line=n</td>
<td>Set the width of longest text lines output; should be at least 60.</td>
</tr>
<tr>
<td>--max-strings=n</td>
<td>Set the maximum number of strings.</td>
</tr>
<tr>
<td>--nest-size=n</td>
<td>Set the maximum number of semantic levels simultaneously active.</td>
</tr>
<tr>
<td>--no-c-style-errors</td>
<td>Don't change the way, error messages are printed.</td>
</tr>
<tr>
<td>--output-directory=dir</td>
<td>Write output files in dir instead of the current directory. Look up input files in dir first, then along the normal search path.</td>
</tr>
<tr>
<td>--output-format=format</td>
<td>Set the output format mode, where format must be either dvi or pdf. This also influences the set of graphics formats understood by pdfTeX.</td>
</tr>
<tr>
<td>--param-size=n</td>
<td>Set the maximum number of simultaneous macro parameters.</td>
</tr>
<tr>
<td>--parse-first-line</td>
<td>Check whether the first line of the main input file starts with %&amp;, and parse if it does. This can be used to specify extra command-line options.</td>
</tr>
<tr>
<td>--pool-free=n</td>
<td>Set the minimum pool space left after loading the format.</td>
</tr>
<tr>
<td>--pool-size=n</td>
<td>Set the maximum number of characters in strings, including all error messages and help texts, and the names of all fonts and control sequences.</td>
</tr>
<tr>
<td>--quiet</td>
<td>Suppress all output, except errors.</td>
</tr>
<tr>
<td>--record-package-usages=file</td>
<td>Record all package usages and write them into file.</td>
</tr>
<tr>
<td>--recorder</td>
<td>Enable the file name recorder. This leaves a trace of the files opened for input and output in a file with the extension .fls.</td>
</tr>
<tr>
<td>--restrict-write18</td>
<td></td>
</tr>
<tr>
<td>--save-size=n</td>
<td>Set the amount of space for saving values outside of current group.</td>
</tr>
<tr>
<td>--src-specials</td>
<td>Embed source file information (source specials) in the DVI file.</td>
</tr>
</tbody>
</table>
--stack-size=n
Set the maximum number of simultaneous input sources.

--string-vacancies=n
Set the minimum number of characters that should be available for the user's control sequences and font names, after the compiler's own error messages are stored. Must be at least 25000 less than pool_size, but doesn't need to be nearly that large.

--synctex=n
Generate SyncTeX data for previewers. If n is zero, no .synctex file is created. If n is negative, the .synctex file is a text file. If n is positive, the .synctex file is compressed with gzip and the .gz file name extension is added.

Furthermore, n is interpreted as a bit field:

Bit 1 (n AND 2) Don't add the .gz file name extension.
Bit 2 (n AND 4) Activate form support.
Bit 3 (n AND 8) Activate better compression.

--tcx=tcxname

--time-statistics
Show processing time statistics.

--trace[=tracestreams]
Enable trace messages. The tracestreams argument, if specified, is a comma-separated list of trace stream names (Chapter 9, Trace Streams).

--trie-size=n
Set the amount of space for hyphenation patterns.

--undump=name
Use name as the name of the format to be used, instead of the name by which the program was called or a

%&
line.

--version
Show version information and exit.

Files
pdftex.cfg(5)

Environment

MIKTEX_EDITOR
The editor to use when selecting e in the error prompt menu.

The value can contain these placeholders:

%f The name of the file, which contains the erroneous line of TeX code.
%l The line number.

MIKTEX_TRACE
Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.
TEXINPUTS  Extra paths to locate TeX \input and \openin files.

TEXINPUTS  Extra paths to locate TeX \input and \openin files.

TFMFONTS  Extra paths to locate TeX font metric files

See also
miktex-epstopdf(1)

The full pdfTeX manual can be accessed from the home page or CTAN page.
Name

setupwiz — MiKTeX setup wizard

Synopsis

basic-miktex-2.9.xxxx.exe[options]

setup-2.9.xxxx.exe[options]

Description

MiKTeX Setup Wizard is used to install MiKTeX.

There are two instances of the installer:

Basic MiKTeX Installer (basic-miktex-2.9.xxxx.exe)  

Basic MiKTeX Installer is used to set up a basic MiKTeX system. All required resources are embedded in the installer, i.e., nothing else needs to be downloaded from the Internet.

MiKTeX Net Installer (setup-2.9.xxxx.exe)  

MiKTeX Net Installer is used to set up a complete MiKTeX system. In a first step, all required resources will be downloaded from the Internet. In a second step, a complete MiKTeX system is installed.

Both installers read command-line options from the file setupwiz.opt, if it exists.

Options

--allow-unattended-reboot  

Restart the system, if necessary.

--common-config=dir  

Set the location of the common configuration directory. This option requires administrator privileges.

--common-data=dir  

Set the location of the common data directory. This option requires administrator privileges.

--common-install=dir  

Set the common installation directory. This option requires administrator privileges.

--common-roots=dirs  

Register additional directories for all users. dirs must be a semicolon-separated list of fully qualified path names. This option requires administrator privileges.

Environment variables (<VARNAME>) can be used.

--download-only  

Download all required packages, but do not otherwise install MiKTeX.

--dry-run  

Simulate. No files shall be downloaded and/or installed.

--install-from-local-repository  

Install MiKTeX from a directory (to be specified with the --local-package-repository option).

--local-package-repository=dir  

Download into (Install from) the specified directory.
--no-additional-roots  Do not integrate additional TEXMF root directories into the MiKTeX setup.

--no-registry  Do not store path information in the Windows Registry but write the startup configuration file (miktexstartup.ini).

--package-set=set  Download/Install the specified package set. This must be one of basic, complete.

--portable  Setup MiKTeX Portable.

--private  Install MiKTeX for the current user only.

--program-folder=name  Add shortcuts to the specified program folder.

--remote-package-repository=url  Download from the specified the URL.

--shared  Install MiKTeX for everyone using this computer. This option requires administrator privileges.

--unattended  Run in unattended mode.

--user-config=dir  Set the location of the configuration directory for the current user.

Environment variables (<VARNAME>) can be used. See the example below.

--user-data=dir  Set the location of the data directory for the current user.

Environment variables (<VARNAME>) can be used. See the example below.

--user-install=dir  Set the user installation directory.

Environment variables (<VARNAME>) can be used. See the example below.

--user-roots=dirs  Register additional directories for the current user. dirs must be a semicolon-separated list of fully qualified path names.

Environment variables (<VARNAME>) can be used.

**Examples**

In this example, MiKTeX is installed from a network share (\server\miktex\repository). User directories are specified by using environment variables (<VARNAME>) which are expanded at run-time.

> setupwiz --install-from-local-repository ^
  --local-package-repository=\server\miktex\repository ^
  --package-set=complete ^
  --shared ^
  --user-config=\APPDATA\MiKTeX\2.9 ^
  --user-data=\LOCALAPPDATA\MiKTeX\2.9 ^
  --user-install=\APPDATA\MiKTeX\2.9
Programs

Name

miktex-tex — text formatting and typesetting

Synopsis

miktex-tex [option...] [[file] | [\command...]]

Description

This man page is an adaption of the corresponding TeX Live man page.

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

TeX formats the interspersed text and commands contained in the named files and outputs a typesetter independent file (called DVI, which is short for DeVice Independent). TeX's capabilities and language are described in The TeXbook. TeX is normally used with a large body of precompiled macros, and there are several specific formatting systems, such as LaTeX, which require the support of several macro files.

This version of TeX looks at its command-line to see what name it was called under. Both initex and virtex are links to the miktex-tex executable. When called as initex (or when the -initialize option is given) it can be used to precompile macros into a .fmt file. When called as virtex it will use the plain format. When called under any other name, TeX will use that name as the name of the format to use. For example, when called as miktex-tex the tex format is used, which is identical to the plain format. The commands defined by the plain format are documented in The TeXbook. Other formats that are often available include latex and amstex.

The non-option command line arguments to the TeX program are passed to it as the first input line. (But it is often easier to type extended arguments as the first input line, since shells tend to gobble up or misinterpret TeX's favorite symbols, like backslashes, unless you quote them.) As described in The TeXbook, that first line should begin with a file name, a \controlsequence, or a &formatname.

The normal usage is to say miktex-tex paper to start processing paper.tex. The name paper will be the “jobname”, and is used in forming output file names. If TeX doesn't get a file name in the first line, the job name is texp. When looking for a file, TeX looks for the name with and without the default extension (.tex) appended, unless the name already contains that extension. If paper is the “jobname”, a log of error messages, with rather more detail than normally appears on the screen, will appear in paper.log, and the output file will be in paper.dvi.

This version of TeX will look in the first line of the file paper.tex to see if it begins with the magic sequence %. If the first line begins with %&format --translate-file tcxname, then TeX will use the named format and translation table tcxname to process the source file. Either the format name or the --translate-file specification may be omitted, but not both. This overrides the format selection based on the name by which the program is invoked. The -parse-first-line option or the parse_first_line configuration value controls whether this behaviour is enabled.

The e response to TeX's error-recovery prompt causes the default editor to start up at the current line of the current file. The configuration value [Core]Editor can be used to change the editor used. It may contain a string with %f indicating where the file name goes and %l indicating where the decimal line number (if any) goes. For example, an [Core]Editor string for emacs can be set with the command

> initexmf --set-config-value="[Core]Editor=emacs +%l %f"
A convenient file is `null.tex`, containing nothing. When TeX can't find a file it thinks you want to input, it keeps asking you for another file name; responding `null` gets you out of the loop if you don't want to input anything. You can also type your EOF character (usually Control+Z).

### Options

`--alias=name` Pretend to be program `name`, i.e., set program (and memory dump) name to `name`. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to `name` and invoking `name`.

`--aux-directory=dir` Set `dir` as the directory to which auxiliary files are written. Also look for input files in `dir` first, before along the normal search path.

`--buf-size=n` Set the maximum number of characters simultaneously present in current lines of open files and in control sequences between \csname and \endcsname. TeX uses the buffer to contain input lines, but macro expansion works by writing material into the buffer and reparsing the line. As a consequence, certain constructs require the buffer to be very large, even though most documents can be handled with a small value.

`--c-style-errors` Change the way, error messages are printed. The alternate style looks like error messages from many compilers and is easier to parse for some editors.

`--disable-8bit-chars` Make only 7-bit characters printable.

`--disable-installer` Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

`--disable-pipes` Disable input (output) from (to) child processes.

`--disable-write18` Disable checking whether the first line of the main input file starts with %&.

`--enable-8bit-chars` Make all characters printable.

`--enable-enctex` Enable encTeX extensions such as \unbyte.

`--enable-installer` Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

`--enable-mltex` Enable MLTeX extensions such as \charsubdef.

`--enable-pipes` Enable input (output) from (to) child processes.

`--enable-write18` Set the width of context lines on terminal error messages.

`--extra-mem-bot=n` Set the extra size (in memory words) for large data structures like boxes, glue, breakpoints, et al. Relevant only after the memory dump file has been read.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--extra-mem-top=n</td>
<td>Set the extra size (in memory words) for chars, tokens, et al. Relevant only after the memory dump file has been read.</td>
</tr>
<tr>
<td>--font-max=n</td>
<td>Set the maximum internal font number.</td>
</tr>
<tr>
<td>--font-mem-size=n</td>
<td>Set the size, in TeX memory words, of the font memory.</td>
</tr>
<tr>
<td>--half-error-line=n</td>
<td>Set the width of first lines of contexts in terminal error messages.</td>
</tr>
<tr>
<td>--halt-on-error</td>
<td>Quit after the first error.</td>
</tr>
<tr>
<td>--hash-extra=n</td>
<td>Set the extra space for the hash table of control sequences (which allows 10K names as distributed).</td>
</tr>
<tr>
<td>--help</td>
<td>Give help and exit.</td>
</tr>
<tr>
<td>--hhelp</td>
<td>This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.</td>
</tr>
<tr>
<td>--include-directory=dir</td>
<td>Add the directory dir to the head of the list of directories to be searched for input files.</td>
</tr>
<tr>
<td>--initialize</td>
<td>Become the INI variant of the program.</td>
</tr>
<tr>
<td>--interaction=mode</td>
<td>Set the interaction mode. Must be one of batchmode, nonstopmode, scrollmode and errorstopmode. The meaning of these modes is the same as the corresponding commands.</td>
</tr>
<tr>
<td>--job-name=name</td>
<td>Set the name of the job (\jobname). This has an affect on the output file names.</td>
</tr>
<tr>
<td>--job-time=file</td>
<td>Set the time-stamp of all output files equal to file's time-stamp.</td>
</tr>
<tr>
<td>--main-memory=n</td>
<td>Change the total size (in memory words) of the main memory array. Relevant only while creating memory dump files.</td>
</tr>
<tr>
<td>--max-in-open=n</td>
<td>Set the maximum number of input files and error insertions that can be going on simultaneously.</td>
</tr>
<tr>
<td>--max-print-line=n</td>
<td>Set the width of longest text lines output; should be at least 60.</td>
</tr>
<tr>
<td>--max-strings=n</td>
<td>Set the maximum number of strings.</td>
</tr>
<tr>
<td>--nest-size=n</td>
<td>Set the maximum number of semantic levels simultaneously active.</td>
</tr>
<tr>
<td>--no-c-style-errors</td>
<td>Don't change the way, error messages are printed.</td>
</tr>
<tr>
<td>--output-directory=dir</td>
<td>Write output files in dir instead of the current directory. Look up input files in dir first, then along the normal search path.</td>
</tr>
<tr>
<td>--param-size=n</td>
<td>Set the maximum number of simultaneous macro parameters.</td>
</tr>
<tr>
<td>--parse-first-line</td>
<td>Check whether the first line of the main input file starts with %&amp;, and parse if it does. This can be used to specify extra command-line options.</td>
</tr>
</tbody>
</table>
Programs

--pool-free=n  Set the minimum pool space left after loading the format.

--pool-size=n  Set the maximum number of characters in strings, including all error messages and help texts, and the names of all fonts and control sequences.

--quiet  Suppress all output, except errors.

--record-package-usages=file  Record all package usages and write them into file.

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

--restrict-write18

--save-size=n  Set the amount of space for saving values outside of current group.

--src-specials  Embed source file information (source specials) in the DVI file.

--stack-size=n  Set the maximum number of simultaneous input sources.

--string-vacancies=n  Set the minimum number of characters that should be available for the user's control sequences and font names, after the compiler's own error messages are stored. Must be at least 25000 less than pool_size, but doesn't need to be nearly that large.

--tcx=tcxname

--time-statistics  Show processing time statistics.

--trace[=tracestreams]  Enable trace messages. The tracestreams argument, if specified, is a comma-separated list of trace stream names (Chapter 9, Trace Streams).

--trie-size=n  Set the amount of space for hyphenation patterns.

--undump=name  Use name as the name of the format to be used, instead of the name by which the program was called or a %& line.

--version  Show version information and exit.

Environment

MIKTEX_EDITOR  The editor to use when selecting e in the error prompt menu.

The value can contain these placeholders:

%f  The name of the file, which contains the erroneous line of TeX code.

%l  The line number.

MIKTEX_TRACE
Comma-separated list of trace stream names (see Chapter 9, Trace Streams). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

**TEXINPUTS**
Extra paths to locate TeX \input and \openin files.

**TFMFonts**
Extra paths to locate TeX font metric files

**Bugs**

This version of TeX implements a number of optional extensions. In fact, many of these extensions conflict to a greater or lesser extent with the definition of TeX. This version of TeX fails to trap arithmetic overflow when dimensions are added or subtracted. Cases where this occurs are rare, but when it does the generated DVI file will be invalid.

**See Also**

texify(1)

Name
texify — MiKTeX compiler driver

Synopsis
texify [option...] file...

Description
texify runs Texinfo or LaTeX input files through miktex-tex (miktex-pdftex) in turn until all cross-references are resolved, building all indices.

The directory containing each file is searched for included files. The suffix of file is used to determine its language (LaTeX or Texinfo).

makeinfo is used to perform Texinfo macro expansion before running miktex-tex when needed.

Options
-@ Use @input (instead of \input); for preloaded Texinfo.
--batch, -b No interaction.
--clean, -c Remove all auxiliary files.
--expand, -e Force macro expansion using makeinfo.
-I dir Search dir for input files.
--help, -h Display help and exit successfully.
--language=lang, -l lang Specify the language of input files: either latex or texinfo.
--max-iterations=n Limits the number of iterations to prevent endless processing. The default for n is 5.
--mkidx-option=option Pass option to the index generator.
--pdf, -p Use miktex-pdftex (or miktex-pdflatex) for processing.
--quiet, -q, --silent, -s No screen output unless errors plies --batch).
--run-viewer Run a viewer on the resulting DVI (PDF) file.
--src Pass --src-specials to the TeX compiler.
Programs

--texinfo=cmd, --t=cmd
Insert cmd after @setfilename in copy of input file. Multiple
values accumulate.

--tex-option=option
Pass option to the compiler.

--verbose, -V
Print information on what is being done.

--version, -v
Display version information and exit successfully.

--viewer-option=option
Pass option to the viewer.

Environment Variables

The values of the BIBTEX, LATEX (or PDFLATEX), MAKEINDEX, MAKEINFO, TEX (or PDFTEX), and
TEXINDEX environment variables are used to run those commands, if they are set.
Name
miktex-xetex — Unicode-based TeX engine

Synopsis
miktex-xetex [option...] [[file] | [\command...]]

Description
This man page is an adaption of the corresponding TeX Live man page.

Run the XeTeX typesetter on file, usually creating file.pdf. If the file argument has no extension, .tex will be appended to it. Instead of a file name, a set of XeTeX commands can be given, the first of which must start with a backslash.

XeTeX has simple font installation; it can use any installed fonts in the operating system without configuring TeX font metric. As a result, XeTeX can access font features such as special ligatures and variable font weights.

Options

--alias=name Pretend to be program name, i.e., set program (and memory dump) name to name. This may affect the search paths and other values used. Using this option is equivalent to copying the program file to name and invoking name.

--aux-directory=dir Set dir as the directory to which auxiliary files are written. Also look for input files in dir first, before along the normal search path.

--buf-size=n Set the maximum number of characters simultaneously present in current lines of open files and in control sequences between \csname and \endcsname. TeX uses the buffer to contain input lines, but macro expansion works by writing material into the buffer and reparsing the line. As a consequence, certain constructs require the buffer to be very large, even though most documents can be handled with a small value.

--c-style-errors Change the way, error messages are printed. The alternate style looks like error messages from many compilers and is easier to parse for some editors.

--disable-8bit-chars Make only 7-bit characters printable.

--disable-installer Disable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--disable-pipes Disable input (output) from (to) child processes.

--disable-write18

--dont-parse-first-line Disable checking whether the first line of the main input file starts with %&.

--enable-8bit-chars Make all characters printable.

--enable-etex
Enable eTeX extensions.

--enable-installer
Enable automatic installation of packages. Specifying this option overrules settings in the MiKTeX configuration data store.

--enable-mltex
Enable MLTeX extensions such as \charsubdef.

--enable-pipes
Enable input (output) from (to) child processes.

--enable-write18

--error-line=n
Set the width of context lines on terminal error messages.

--extra-mem-bot=n
Set the extra size (in memory words) for large data structures like boxes, glue, breakpoints, et al. Relevant only after the memory dump file has been read.

--extra-mem-top=n
Set the extra size (in memory words) for chars, tokens, et al. Relevant only after the memory dump file has been read.

--font-max=n
Set the maximum internal font number.

--font-mem-size=n
Set the size, in TeX memory words, of the font memory.

--half-error-line=n
Set the width of first lines of contexts in terminal error messages.

--halt-on-error
Quit after the first error.

--hash-extra=n
Set the extra space for the hash table of control sequences (which allows 10K names as distributed).

--help
Give help and exit.

--hhelp
This option is only available on Windows systems: show the manual page in an HTML Help window and exit when the window is closed.

--include-directory=dir
Add the directory dir to the head of the list of directories to be searched for input files.

--initialize
Become the \texttt{INI} variant of the program.

--interaction=mode
Set the interaction mode. Must be one of \texttt{batchmode}, \texttt{nonstopmode}, \texttt{scrollmode} and \texttt{errorstopmode}. The meaning of these modes is the same as the corresponding commands.

--job-name=name
Set the name of the job (\jobname). This has an affect on the output file names.

--job-time=file
Set the time-stamp of all output files equal to file's time-stamp.

--main-memory=n
Change the total size (in memory words) of the main memory array. Relevant only while creating memory dump files.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--max-in-open=n</code></td>
<td>Set the maximum number of input files and error insertions that can be going on simultaneously.</td>
</tr>
<tr>
<td><code>--max-print-line=n</code></td>
<td>Set the width of longest text lines output; should be at least 60.</td>
</tr>
<tr>
<td><code>--max-strings=n</code></td>
<td>Set the maximum number of strings.</td>
</tr>
<tr>
<td><code>--nest-size=n</code></td>
<td>Set the maximum number of semantic levels simultaneously active.</td>
</tr>
<tr>
<td><code>--no-c-style-errors</code></td>
<td>Don't change the way, error messages are printed.</td>
</tr>
<tr>
<td><code>--no-pdf</code></td>
<td>Generate XDV (extended DVI) output rather than PDF.</td>
</tr>
<tr>
<td><code>--output-directory=dir</code></td>
<td>Write output files in <code>dir</code> instead of the current directory. Look up input files in <code>dir</code> first, then along the normal search path.</td>
</tr>
<tr>
<td><code>--output-driver=cmd</code></td>
<td>Use <code>cmd</code> as the XDV-to-PDF driver instead of <code>xdvipdfmx</code>.</td>
</tr>
<tr>
<td><code>--papersize=string</code></td>
<td>Set PDF media size to <code>string</code>.</td>
</tr>
<tr>
<td><code>--param-size=n</code></td>
<td>Set the maximum number of simultaneous macro parameters.</td>
</tr>
<tr>
<td><code>--parse-first-line</code></td>
<td>Check whether the first line of the main input file starts with <code>%&amp;</code>, and parse if it does. This can be used to specify extra command-line options.</td>
</tr>
<tr>
<td><code>--pool-free=n</code></td>
<td>Set the minimum pool space left after loading the format.</td>
</tr>
<tr>
<td><code>--pool-size=n</code></td>
<td>Set the maximum number of characters in strings, including all error messages and help texts, and the names of all fonts and control sequences.</td>
</tr>
<tr>
<td><code>--quiet</code></td>
<td>Suppress all output, except errors.</td>
</tr>
<tr>
<td><code>--record-package-usages=file</code></td>
<td>Record all package usages and write them into <code>file</code>.</td>
</tr>
<tr>
<td><code>--recorder</code></td>
<td>Enable the file name recorder. This leaves a trace of the files opened for input and output in a file with the extension <code>.fls</code>.</td>
</tr>
<tr>
<td><code>--restrict-write18</code></td>
<td></td>
</tr>
<tr>
<td><code>--save-size=n</code></td>
<td>Set the amount of space for saving values outside of current group.</td>
</tr>
<tr>
<td><code>--src-specials</code></td>
<td>Embed source file information (source specials) in the DVI file.</td>
</tr>
<tr>
<td><code>--stack-size=n</code></td>
<td>Set the maximum number of simultaneous input sources.</td>
</tr>
<tr>
<td><code>--string-vacancies=n</code></td>
<td>Set the minimum number of characters that should be available for the user's control sequences and font names, after the compiler's own error messages are stored. Must be at least 25000 less than <code>pool_size</code>, but doesn't need to be nearly that large.</td>
</tr>
<tr>
<td><code>--synctex=n</code></td>
<td>Generate SyncTeX data for previewers. If <code>n</code> is zero, no <code>.synctex</code> file is created. If <code>n</code> is negative, the <code>.synctex</code> file is a text file. If <code>n</code> is positive, the <code>.synctex</code> file is compressed with <code>gzip</code> and the <code>.gz</code> file name extension is added.</td>
</tr>
</tbody>
</table>
Furthermore, $n$ is interpreted as a bit field:

- **Bit 1** ($n$ AND 2) Don't add the `.gz` file name extension.
- **Bit 2** ($n$ AND 4) Activate form support.
- **Bit 3** ($n$ AND 8) Activate better compression.

---

**--time-statistics**

Show processing time statistics.

**--trace [=tracestreams]**

Enable trace messages. The `tracestreams` argument, if specified, is a comma-separated list of trace stream names (Chapter 9, *Trace Streams*).

**--trie-size=n**

Set the amount of space for hyphenation patterns.

**--undump=name**

Use `name` as the name of the format to be used, instead of the name by which the program was called or a `%` line.

**--version**

Show version information and exit.

---

### Environment

**MIKTEX_EDITOR**

The editor to use when selecting `e` in the error prompt menu.

The value can contain these placeholders:

- `%f` The name of the file, which contains the erroneous line of TeX code.
- `%l` The line number.

**MIKTEX_TRACE**

Comma-separated list of trace stream names (see Chapter 9, *Trace Streams*). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.

**TEXINPUTS**

Extra paths to locate TeX `\input` and `\openin` files.

**TFMFONTS**

Extra paths to locate TeX font metric files.
Chapter 7. Files
Name
miktex.ini — MiKTeX configuration data store

Description
MiKTeX configurations settings are read from the file miktex.ini.

Syntax
The syntax follows the informal conventions of a traditional INI file [https://en.wikipedia.org/wiki/INI_file].

Querying and Modifying
It is recommended that you use the MiKTeX Configuration Utility (see initexmf(1)) to query and modify configuration settings.

Settings Reference

[miktex]

;; This variable specifies the external program called for
;; TeX's interactive 'e' option. %l is replaced by the line
;; number and %f by the current file name.
;; Editor = miktex-texworks -p=%l "%f"

[Core]

;; Shell command mode.
;; Forbidden: don't allow any shell commands
;; Restricted: allow the commands listed in AllowedShellCommands[]
;; Unrestricted: allow all shell commands
ShellCommandMode = Restricted

;; The programs listed here are probably safe: they either do
;; not write any output files or implement restrictions
;; similar to or higher than
;; [Core]AllowUnsafeOutputFiles=true.
;; They also have no features to invoke arbitrary other
;; programs, and no known exploitable bugs. All to the best
;; of our knowledge. They also have practical use for being
;; called from TeX.
AllowedShellCommands[] = miktex-bibtex
AllowedShellCommands[] = miktex-bibtex8
AllowedShellCommands[] = miktex-epstopdf
AllowedShellCommands[] = miktex-gregorio
AllowedShellCommands[] = miktex-kpsewhich
AllowedShellCommands[] = miktex-makeindex
AllowedShellCommands[] = bibtex
AllowedShellCommands[] = bibtex8
AllowedShellCommands[] = epstopdf
AllowedShellCommands[] = extractbb
AllowedShellCommands[] = findtexmf
AllowedShellCommands[] = gregorio
AllowedShellCommands[] = kpsewhich
AllowedShellCommands[] = makeindex
AllowedShellCommands[] = texosquery-jre8

;;; Do we allow TeX \input or \openin on file names starting
;;; with `.' (e.g., .rhosts) or outside the current tree (e.g.,
;;; /etc/passwd)?
AllowUnsafeInputFiles = true

;;; Do we allow TeX \openout on file names starting with `.'
;;; (e.g., .rhosts) or outside the current tree (e.g.,
;;; /etc/passwd)?
AllowUnsafeOutputFiles = false

;;; Automatically turn on administrator mode for elevated MiKTeX programs
;;; in a shared setup.
AutoAdmin = ?

;;; System-wide directory in which to create symbolic links to
;;; MiKTeX executables.
CommonLinkTargetDirectory =

;;; Per-user directory in which to create symbolic links to
;;; MiKTeX executables.
UserLinkTargetDirectory =

[Core.FileTypes.afm]

;;; Search path for Adobe font metric (AFM) files.
Paths[] = .
Paths[] = %R/fonts/afm//

;;; Environment variables to be used for searching AFM files.
EnvVars[] = AFMFONTS
EnvVars[] = TEXFONTS

;;; AFM file name extensions.
Extensions[] = .afm

[Core.FileTypes.base]

;;; Search path for METAPOST memory dump files.
Paths[] = .
Paths[] = %r/miktex/data/le

;;; METAPOST memory dump file name extensions.
Extensions[] = .base

[Core.FileTypes.bib]

;;; Search path for BibTeX database files.
Paths[] = .
Files

Paths[] = %R/bibtex/bib//

;; Environment variables to be used for searching BibTeX
;; database files.
EnvVars[] = BIBINPUTS
EnvVars[] = TEXBIB

;; BibTeX database file name extensions.
Extensions[] = .bib

[Core.FileTypes.bst]

;; Search path for BibTeX style files.
Paths[] = .
Paths[] = %R/bibtex/{bst,csf}//

;; Environment variables to be used for searching BibTeX
;; style files.
EnvVars[] = BSTINPUTS

;; BibTeX style file name extensions.
Extensions[] = .bst

[Core.FileTypes.cid maps]

;; Search path for CID map files.
Paths[] = .
Paths[] = %R/fonts/cid//

;; Environment variables to be used for searching CID map
;; files.
EnvVars[] = FONTCIDMAPS

;; CID map file name extensions.
Extensions[] = .cid
Extensions[] = .cidmap

[Core.FileTypes.clua]

;; Search path for dynamic libraries for Lua.
Paths[] = .
Paths[] = %R/scripts/{$progname,$engine,}/lua//

;; Environment variables to be used for searching dynamic
;; libraries for Lua.
EnvVars[] = CLUAINPUTS

;; File name extensions for Lua dynamic libraries.
Extensions[] = .dll
Extensions[] = .so

[Core.FileTypes.cmap files]

;; Search path for character map files.
Files

Paths[] = .
Paths[] = %R/fonts/cmap//

;; Environment variables to be used for character map files.
EnvVars[] = CMAPFONTS
EnvVars[] = TEXFONTS

[Core.FileTypes.cweb]

;; Search path for CWeb input files.
Paths[] = .
Paths[] = %R/cweb//

;; Environment variables to be used for searching Cweb input files.
EnvVars[] = CWEBINPUTS

;; CWeb file name extensions.
Extensions[] = .w

[Core.FileTypes.dvi]

;; Search path for DVI files.
Paths[] = .
Paths[] = %R/doc//

;; DVI file name extensions.
Extensions[] = .dvi

[Core.FileTypes.dvips config]

;; Search path for Dvips configuration files.
Paths[] = .
Paths[] = %R/dvips//

;; Environment variables to be used for searching Dvips configuration files.
EnvVars[] = TEXCONFIG

[Core.FileTypes.enc]

;; Search path for encoding vector files.
Paths[] = .
Paths[] = %R/fonts/enc//
Paths[] = %R/miktex/config//
Paths[] = %R/dvips//
Paths[] = %R/pdftex//
Paths[] = %R/dvipdfm//

;; Environment variables to be used for searching encoding vector files.
EnvVars[] = ENCFONTS
EnvVars[] = TEXFONTS
Files

;;;; Encoding vector file name extensions.
Extensions[] = .enc

[Core.FileTypes.font feature files]

;;;; Search path for font feature files.
Paths[] = .
Paths[] = %R/fonts/fea/

;;;; Environment variables to be used for searching font feature
;;;; files.
EnvVars[] = FONTFEATURES

;;;; Font feature file name extensions.
Extensions[] = .fea

[Core.FileTypes.fmt]

;;;; Search path for TeX memory dump files.
Paths[] = .
Paths[] = %r/miktex/data/le/{$engine,}

;;;; TeX memory dump file name extensions.
Extensions[] = .fmt

[Core.FileTypes.gf]

;;;; Search path for generic font bitmap files.
Paths[] = .
Paths[] = %R/fonts/

;;;; Environment variables to be used for searching generic font
;;;; bitmap files.
EnvVars[] = GFFONTS
EnvVars[] = GLYPHFONTS
EnvVars[] = TEXFONTS

;;;; Generic font bitmap file name extensions.
Extensions[] = .gf

[Core.FileTypes.bitmap font]

;;;; Search path for bitmap font files.
Paths[] = .
Paths[] = %R/fonts/

;;;; Environment variables to be used for searching bitmap font
;;;; files.
EnvVars[] = GLYPHFONTS
EnvVars[] = TEXFONTS

[Core.FileTypes.graphic/figure]

;;;; Search path for figure files.
Files

```
Paths[] = .
Paths[] = %R/dvips//
Paths[] = %R/pdftex//
Paths[] = %R/tex//

;; Environment variables to be used for searching figure files.
EnvVars[] = TEXPICTS
EnvVars[] = TEXINPUTS

;; Figure file name extensions.
Extensions[] = .eps
Extensions[] = .epsi
Extensions[] = .png

[Core.FileTypes.hbf]

;; Search path for HBF files.
Paths[] = .
Paths[] = %R/fonts/misc/hbf//

;; HBF file name extensions.
Extensions[] = .hbf

[Core.FileTypes.ist]

;; Search path for MakeIndex style files.
Paths[] = .
Paths[] = %R/makeindex//

;; Environment variables to be used for searching MakeIndex style files.
EnvVars[] = TEXINDEXSTYLE
EnvVars[] = INDEXSTYLE

;; MakeIndex style file name extensions.
Extensions[] = .ist

[Core.FileTypes.lig files]

;; Search path for ligature definition files.
Paths[] = .
Paths[] = %R/fonts/lig//

;; Environment variables to be used for searching ligature definition files.
EnvVars[] = TEXFONTS

;; Ligature definition file name extensions.
Extensions[] = .lig

[Core.FileTypes.ls-R]

;; Search path for Web2c file name database files.
```
Files

Paths[] = %R

;; Environment variables to be used for searching Web2C file
;; name database files.
EnvVars[] = TEXMFDBS

[Core.FileTypes.lua]

;; Search path for Lua files.
Paths[] = .
Paths[] = %R/scripts/($proname,$engine)/{lua,}://
Paths[] = %R/tex/($proname,generic,)//

;; Environment variables to be used for searching Lua files.
EnvVars[] = LUAINPUTS

;; File name extensions for Lua files.
Extensions[] = .lua
Extensions[] = .luatex
Extensions[] = .luc
Extensions[] = .luctex
Extensions[] = .texlua
Extensions[] = .texluc
Extensions[] = .tlu

;; Environment variables to be used for searching Lua files.
EnvVars[] = LUAINPUTS

[Core.FileTypes.map]

;; Search path for font map files.
Paths[] = .
Paths[] = %R/fonts/map/($proname,pdftex,dvips,)//
Paths[] = %R/miktex/config//
Paths[] = %R/dvips//
Paths[] = %R/pdftex//
Paths[] = %R/dvipdfm//

;; Environment variables to be used for searching font map
;; files.
EnvVars[] = TEXFONTMAPS
EnvVars[] = TEXFONTS

;; Font map file name extensions.
Extensions[] = .map

[Core.FileTypes.mem]

;; Search path for MetaPost memory dump files.
Paths[] = .

;; MetaPost memory dump file name extensions.
Extensions[] = .mem
[Core.FileTypes.mf]

;; Search path for METAFONT input files.
Paths[] = .
Paths[] = %R/metafont//
Paths[] = %R/fonts/source//

;; Environment variables to be used for searching METAFONT
;; input files.
EnvVars[] = MFINPUTS

;; METAFONT file name extensions.
Extensions[] = .mf

[Core.FileTypes.mfpool]

;; Search path for METAFONT program string files.
Paths[] = .

;; Environment variables to be used for searching METAFONT
;; program string files.
EnvVars[] = MFPOOL
EnvVars[] = TEXMFINI

;; METAFONT program string file name extensions.
Extensions[] = .pool

[Core.FileTypes.mft]

;; Search path for MFT style files.
Paths[] = .
Paths[] = %R/mft//

;; Environment variables to be used for searching MFT style
;; files.
EnvVars[] = MFTINPUTS

;; MFT style file name extensions.
Extensions[] = .mft

[Core.FileTypes.misc fonts]

;; Search path for font related files.
Paths[] = .
Paths[] = %R/fonts/misc//

;; Environment variables to be used for font related
;; files.
EnvVars[] = MISCFONTS
EnvVars[] = TEXFONTS

[Core.FileTypes.mlbib]

;; Search path for MlBibTeX database files.
Files

```
Paths[] = .
Paths[] = %R/bibtex/bib/{mlbib,}///

;; Environment variables to be used for searching MlBibTeX
;; databases files.
EnvVars[] = MLBIBINPUTS
EnvVars[] = BIBINPUTS
EnvVars[] = TEXBIB

;; MlBibTeX database file name extensions.
Extensions[] = .mlbib
Extensions[] = .bib

[Core.FileTypes.mlbst]

;; Search path for MlBibTeX style files.
Paths[] = .
Paths[] = %R/bibtex/{mlbst,bst}//

;; Environment variables to be used for searching MlBibTeX
;; style files.
EnvVars[] = MLBSTINPUTS
EnvVars[] = BSTINPUTS

;; MlBibTeX style file name extensions.
Extensions[] = .bst

[Core.FileTypes.mp]

;; Search path for MetaPost input files.
Paths[] = .
Paths[] = %R/metapost///

;; Environment variables to be used for searching MetaPost
;; input files.
EnvVars[] = MPINPUTS

;; MetaPost file name extensions.
Extensions[] = .mp

[Core.FileTypes.mppool]

;; Search path for MetaPost program string files.
Paths[] = .

;; Environment variables to be used for searching MetaPost
;; program string files.
EnvVars[] = MPPOOL
EnvVars[] = TEXMFINI

;; MetaPost program string file name extensions.
Extensions[] = .pool

[Core.FileTypes.MetaPost support]
```
Files

;;; Search path for MetaPost support files.
Paths[] = .
Paths[] = $R/metapost/support/

;;; Environment variables to be used for searching MetaPost
;;; support files.
EnvVars[] = MPSUPPORT

[Core.FileTypes.ocp]

;;; Search path for Omega compiled process files.
Paths[] = .
Paths[] = $R/omega/ocp/

;;; Environment variables to be used for searching Omega
;;; compiled process files.
EnvVars[] = OCPINPUTS

;;; Omega compiled process file name extensions.
Extensions[] = .ocp

[Core.FileTypes.ofm]

;;; Search path for Omega font metric files.
Paths[] = .
Paths[] = $R/fonts/ofm/
Paths[] = $R/fonts/tfm/

;;; Environment variables to be used for searching Omega
;;; font metric files.
EnvVars[] = OFMFPONTS
EnvVars[] = TEXFONTS

;;; Omega font metric file name extensions.
Extensions[] = .ofm
Extensions[] = .tfm

[Core.FileTypes.opl]

;;; Search path for Omega property list files.
Paths[] = .
Paths[] = $R/fonts/opl/

;;; Environment variables to be used for searching Omega
;;; property list files.
EnvVars[] = OPLFPONTS
EnvVars[] = TEXFONTS

;;; Omega property list file name extensions.
Extensions[] = .opl

[Core.FileTypes.otp]
Files

;;; Search path for Omega translation process files.
Paths[] = .
Paths[] = %R/fonts/otp/

;;; Environment variables to be used for searching Omega
;;; translation process files.
EnvVars[] = OTPINPUTS

;;; Omega translation process file name extensions.
Extensions[] = .otp

[Core.FileTypes.opentype fonts]

;;; Search path for OpenType font files.
Paths[] = .
Paths[] = %R/fonts/opentype/

;;; Environment variables to be used for searching OpenType
;;; font files.
EnvVars[] = OPENTYPEFONTS
EnvVars[] = TEXFONTS

;;; OpenType font file name extensions.
Extensions[] = .otf

[Core.FileTypes.ovf]

;;; Search path for Omega virtual font files.
Paths[] = .
Paths[] = %R/fonts/ovf/
Paths[] = %R/fonts/vf/

;;; Environment variables to be used for searching Omega
;;; virtual font files.
EnvVars[] = OVFFONTS
EnvVars[] = TEXFONTS

;;; Omega virtual font file name extensions.
Extensions[] = .ovf

[Core.FileTypes.ovp]

;;; Search path for Omega virtual property list files.
Paths[] = .
Paths[] = %R/fonts/ovp/

;;; Environment variables to be used for searching Omega
;;; virtual property list files.
EnvVars[] = OVFPPONTS
EnvVars[] = TEXFONTS

;;; Omega virtual property list file name extensions.
Extensions[] = .ovp
[Core.FileTypes.pdftex config]

;; Search path for pdfTeX configuration files.
Paths[] = .
Paths[] = %R/pdftex/($proname,)//

;; Environment variables to be used for searching pdfTeX configuration files.
EnvVars[] = PDFTEXCONFIG

[Core.FileTypes.pk]

;; Search path for packed bitmap font files.
Paths[] = .
Paths[] = %R/fonts/

;; Packed bitmap font file name extensions.
Extensions[] = .pk

[Core.FileTypes.other binary files]

;; Search path for program binary files.
Paths[] = .
Paths[] = %R/$proname/

[Core.FileTypes.other text files]

;; Search path for program text files.
Paths[] = .
Paths[] = %R/$proname/

[Core.FileTypes.PostScript header]

;; Search path for downloadable PostScript files.
Paths[] = .
Paths[] = %R/miktex/config/
Paths[] = %R/dvips/
Paths[] = %R/pdftex/
Paths[] = %R/dvipdfm/
Paths[] = %R/fonts/enc/
Paths[] = %R/fonts/type1/
Paths[] = %R/fonts/type42/
Paths[] = %R/fonts/type3/
Paths[] = $psfontdirs

;; Environment variables to be used for searching downloadable PostScript files.
EnvVars[] = TEXPSHEADERS
EnvVars[] = PSHEADERS

;; Downloadable PostScript file name extensions.
Extensions[] = .pro
Extensions[] = .enc
Files

[Core.FileTypes.texmfscrips]

;;;; Search path for architecture-independent executables.
Paths = .
Paths = %R/scripts/($proname,$engine)//

;;;; Environment variables to be used for searching
;;;; architecture-independent executables.
EnvVars = TEXMFSCRIPTS

[Core.FileTypes.subfont definition files]

;;;; Search path for subfont definition files.
Paths = .
Paths = %R/fonts/sfd//

;;;; Environment variables to be used for searching subfont
;;;; definition files.
EnvVars = SFDFONTS
EnvVars = TEXFONTS

;;;; Subfont definition file name extensions.
Extensions = .sfd

[Core.FileTypes.tcx]

;;;; Search path for TCX files.
Paths = .
Paths = %R/miktex/config
Paths = %R/miktex/web2c

;;;; TCX file name extensions.
Extensions = .tcx

[Core.FileTypes.tex]

;;;; Search path for TeX input files.
Paths = .
Paths = %R/tex/($proname,generic,//

;;;; Environment variables to be used for searching TeX input
;;;; files.
EnvVars = TEXINPUTS

;;;; TeX input file name extensions.
Extensions = .tex

[Core.FileTypes.texpool]

;;;; Search path for TeX program string files.
Paths = .

;;;; Environment variables to be used for searching METAFONT
;;;; program string files.
EnvVars[] = TEXPOOL
EnvVars[] = TEXMFFINI

;; TeX program string file name extensions.
Extensions[] = .pool

[Core.FileTypes.TeX system sources]

;; Search path for source files.
Paths[] = .
Paths[] = %R/source/

;; Environment variables to be used for searching source
;; files.
EnvVars[] = TEXSOURCES

[Core.FileTypes.TeX system documentation]

;; Search path for documentation files.
Paths[] = .
Paths[] = %R/doc/miktex/
Paths[] = %R/doc/

;; Environment variables to be used for searching
;; documentation files.
EnvVars[] = TEXDOCS

;; Documentation file name extensions.
Extensions[] = .pdf
Extensions[] = .html
Extensions[] = .md
Extensions[] = .txt
Extensions[] = .ps
Extensions[] = .dvi

[Core.FileTypes.tfm]

;; Search path for TeX font metric files.
Paths[] = .
Paths[] = %R/fonts/tfm/

;; Environment variables to be used for searching TeX font
;; metric files.
EnvVars[] = TFMFONTS
EnvVars[] = TEXFONTS

;; TeX font metric file name extensions.
Extensions[] = .tfm

[Core.FileTypes.troff fonts]

;; Environment variables to be used for searching Troff font
;; files.
EnvVars[] = TRFONTS
[Core.FileTypes.truetype fonts]

;; Search path for TrueType font files.
Paths[] = .
Paths[] = %R/fonts/truetype/

;; Environment variables to be used for searching TrueType
;; font files.
EnvVars[] = TTFONTS
EnvVars[] = TEXFONTS

;; TrueType font file name extensions.
Extensions[] = .ttf
Extensions[] = .ttc

[Core.FileTypes.type1 fonts]

;; Search path for Type1 font files.
Paths[] = .
Paths[] = %R/fonts/type1/

;; Environment variables to be used for searching Type1 font
;; files.
EnvVars[] = T1FONTS
EnvVars[] = T1INPUTS
EnvVars[] = TEXFONTS
EnvVars[] = TEXPSHEADERS
EnvVars[] = PSHEADERS

;; Type1 font file name extensions.
Extensions[] = .pfb
Extensions[] = .pfa

[Core.FileTypes.type42 fonts]

;; Search path for Type42 font files.
Paths[] = .
Paths[] = %R/fonts/type42/

;; Environment variables to be used for searching Type42 font
;; files.
EnvVars[] = T42FONTS
EnvVars[] = TEXFONTS

;; Type42 font file name extensions.
Extensions[] = .t42
Extensions[] = .T42

[Core.FileTypes.vf]

;; Search path for TeX virtual font files.
Paths[] = .
Paths[] = %R/fonts/vf/
Files

;; Environment variables to be used for searching TeX virtual
;; font files.
EnvVars[] = VFFONTS
EnvVars[] = TEXFONTS

;; TeX virtual font file name extensions.
Extensions[] = .vf

[Core.FileTypes.web2c files]

;; Search path for Web2c files.
Paths[] = .
Paths[] = %R/web2c/

[Core.FileTypes.web]

;; Search path for WEB input files.
Paths[] = .
Paths[] = %R/web/

;; Environment variables to be used for searching WEB input
;; files.
EnvVars[] = WEBINPUTS

;; CWeb file name extensions.
Extensions[] = .web

[MPM]

;; Install packages for all users.
AutoAdmin = ?

;; Install missing packages automatically (on-the-fly).
AutoInstall = ?

[TeXandFriends]

;; Create auxiliary directory if '--aux-directory=DIR' refers
;; to a non-existing directory.
CreateAuxDirectory = t

;; Create the output directory if '--output-directory=DIR'
;; refers to a non-existing directory.
CreateOutputDirectory = t

;; Enable file:line:error style messages.
CStyleErrors = f

See also
initexmf(1)
Files

Name

pdftex.cfg — configuration settings for MiKTeX-pdfTeX

Description

MiKTeX-pdfTeX configurations settings are read from the file pdftex.cfg when a format file is being created by MiKTeX-pdfTeX.

Caution

Do not edit this file directly. Run initexmf --edit-config-file pdftex.cfg to edit configuration settings for MiKTeX-pdfTeX.

Instructions

A typical pdftex.cfg file looks like this, setting up output for A4 paper size and the standard TeX offset of 1 inch:

compress_level 9
decimal_digits 3
horigin 1 true in
vorigin 1 true in
image_resolution 300
move_chars 1
output_format 1
page_width 210 true mm
page_height 297 true mm
pdf_minorversion 4
pk_resolution 600

The configuration file sets default values for these parameters, and they all can be overridden in the TeX source file. Dimensions can be specified as true, which makes them immune for magnification (when set).

compress_level This integer parameter specifies the level of text and inline graphics compression. MiKTeX-pdfTeX uses Zip compression. A value of 0 means no compression, 1 means fastest, 9 means best, 2..8 means something in between. Just set this value to 9, unless there is a good reason to do otherwise; 0 is great for testing macros that use \pdfliteral.

decimal_digits This integer specifies the preciseness of real numbers in PDF page descriptions. It gives the maximal number of decimal digits after the decimal point of real numbers. Valid values are in range 0..5. A higher value means more precise output, but also results in a much larger file size and more time to display or print. In most cases the optimal value is 2. This parameter does not influence the precision of numbers used in raw PDF code, like that used in \pdfliteral and annotation action specifications.

horigin & vorigin These dimension parameters can be used to set the offset of the TeX output box from the top left corner of the “paper”.
image_resolution  When MiKTeX-pdfTeX is not able to determine the natural dimensions of an image, it assumes a resolution of type 72 dots per inch. Use this variable to change this default value.

move_chars  Although PDF output is claimed to be portable, especially when all font information is included in the file, problems with printing and viewing have a persistent nature. Moving the characters in range 0–31 sometimes helps a lot. When set to 1, characters are only moved when a font has less than 128 glyphs, when set to 2 higher slots are used too.

output_format  This integer parameter specifies whether the output format should be DVI or PDF. A positive value means PDF output, otherwise we get DVI output.

date_width & date_height  These two dimensional parameters specify the output medium dimensions (the paper, screen or whatever the page is put on). If they are not specified, these values are calculated.

pdf_minorversion  Sets the PDF version of the generated file and the latest allowed PDF version of included PDFs. The value 3 tells MiKTeX-pdfTeX to set the PDF version to 1.3 and allows only included PDFs with versions less than 1.3. A suitable default value is 4.

pk_resolution  One can use this entry to specify the resolution for bitmap fonts. Nowadays most printers are capable to print at least 600 dots per inch, so this is a reasonable default.
Files

**Name**

updmap.cfg — configuration Settings for outline fonts

**Description**

The configuration file updmap.cfg contains declarative instructions, which will be used to build font map files.

**Caution**

Do not edit this file directly. Run `initexmf --edit-config-file updmap` to edit configuration settings for outline fonts.

**Instructions**

updmap.cfg can contain the following instructions:

- **dvipsPreferOutline value**
  Specifies whether Dvips prefers bitmap fonts or outline fonts if both are available. Valid values are `true` (default) and `false`.
  
  Independend of this setting, outlines can be forced by putting
  
  `p psfonts_t1.map`
  
  into a configuration file that Dvips reads. Bitmaps (for the fonts in question) can be forced by putting
  
  `p psfonts_pk.map`
  
  into a configuration file. Such configuration files are provided, which can be enabled via
  
  `dvips -Poutline ...`
  
  resp.
  
  `dvips -Ppk ...`

- **LW35 value**
  Specifies which fonts for the “Basic 35 LaserWriter Fonts” will be used and how their file names are chosen. Valid values:
  
<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URW</td>
<td>URW fonts with “vendor” file names (e.g., n019064l.pfb).</td>
</tr>
<tr>
<td>URWkb</td>
<td>URW fonts with “berry” file names (e.g., uhvbo8ac.pfb). URWkb is the default value.</td>
</tr>
<tr>
<td>ADOBE</td>
<td>Adobe fonts with “vendor” file names (e.g., hvnbo___.pfb).</td>
</tr>
<tr>
<td>ADOBEkb</td>
<td>Adobe fonts with “berry” file names (e.g., phvbo8an.pfb).</td>
</tr>
</tbody>
</table>

- **dvipsDownloadBase35 value**
  Specifies whether Dvips downloads the standard 35 LaserWriter fonts with the document. If these fonts are not downloaded, then
they must be available in the PostScript printer (interpreter). Valid values are \texttt{true} and \texttt{false} (default).

Whatever is specified here, the user can override it by specifying

dvips -Pdownload35 ...

resp.

dvips -Pbuiltin35 ...

to either download the LW35 fonts resp. use the build-in fonts.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{pdftexDownloadBase14} \texttt{value}</td>
<td>Specifies whether pdfTeX downloads the base 14 PDF fonts. Valid values are \texttt{true} (default) and \texttt{false}. Since some configurations (PostScript / PDF tools / printers) use bad default fonts, it is safer to download the fonts. The PDF files will get bigger, though.</td>
</tr>
<tr>
<td>\texttt{dvipdfmDownloadBase14} \texttt{value}</td>
<td>Specifies whether Dvipdfm downloads the base 14 PDF fonts. Valid values are \texttt{true} (default) and \texttt{false}. Since some configurations (PostScript / PDF tools / printers) use bad default fonts, it is safer to download the fonts. The PDF files will get bigger, though.</td>
</tr>
<tr>
<td>\texttt{Map filename}</td>
<td>Arranges that the contents of \textit{filename} will be included in \texttt{psfonts.map}.</td>
</tr>
<tr>
<td>\texttt{MixedMap filename}</td>
<td>Arranges that the contents of \textit{filename} will be included in \texttt{psfonts.map}, unless \texttt{dvipsPreferOutline} is set to \texttt{false}. “Mixed” means that the fonts referenced in the file are available as bitmap and as outline.</td>
</tr>
</tbody>
</table>
### Chapter 8. Environment variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BIBINPUTS</strong></td>
<td>Extra paths to locate <code>.bib</code> files.</td>
</tr>
<tr>
<td><strong>BSTINPUTS</strong></td>
<td>Extra paths to locate <code>.bst</code> files.</td>
</tr>
<tr>
<td><strong>MFINPUTS</strong></td>
<td>Extra paths to locate METAFONT input and openin files.</td>
</tr>
<tr>
<td><strong>MIKTEX_REPOSITORY</strong></td>
<td>Location of the default package repository. This can be either a fully qualified path name (a local package repository) or an URL (a remote package repository).</td>
</tr>
<tr>
<td><strong>MIKTEX_TRACE</strong></td>
<td>Comma-separated list of trace stream names (see Chapter 9, <em>Trace Streams</em>). If this variable is set, then MiKTeX programs will write trace messages into the configured log sink.</td>
</tr>
<tr>
<td><strong>MFINPUTS</strong></td>
<td>Extra paths to locate METAFONT input and openin files.</td>
</tr>
<tr>
<td><strong>TEXINPUTS</strong></td>
<td>Extra paths to locate TeX <code>\input</code> and <code>\openin</code> files.</td>
</tr>
<tr>
<td><strong>TFMFONTS</strong></td>
<td>Extra paths to locate TeX font metric files</td>
</tr>
</tbody>
</table>
Chapter 9. Trace Streams

access file tests (access(), stat())
config MiKTeX configuration settings
core MiKTeX core library
curl cURL library
dib device independant bitmaps
dvibitmap DVI bitmaps
dvicolor DVI color
dvifile DVI files
dvige DVI garbage collector
dvi hypertex DVI hypertex specials
dvipage DVI page builder
dvipkbitmap DVI PK raster operations
dvipkchar DVI PK characters
dvipkfont DVI PK fonts
dvisearch DVI source specials
dvifm DVI font metrics
dvivfchar DVI virtual font characters
dvivfont DVI virtual fonts
e env environment variables
error error conditions
extractor MiKTeX package archive file extractor
files file operations
filesearch file searching
fndb file name database operations
fontinfo font information retrieval
mem TeX & Friends memory allocation
mmap memory mapped files
mpm package manager
| mtprint | MiKTeX print utility |
| packages | packages |
| process | execution of secondary processes |
| tempfile | temporary files |
| time | execution time |
| values | configuration values |
| yap | Yap |
Chapter 10. Run-Time Defaults

MiKTeX configuration settings are initialized with default values which are described in this chapter.

All MiKTeX Programs

[miktex]

;; This variable specifies the external program called for
;; TeX's interactive `e' option. %l is replaced by the line
;; number and %f by the current file name.
;;Editor = miktex-texworks -p=%l "%f"

[Core]

;; Shell command mode.
;; Forbidden: don't allow any shell commands
;; Restricted: allow the commands listed in AllowedShellCommands[]
;; Unrestricted: allow all shell commands
ShellCommandMode = Restricted

;; The programs listed here are probably safe: they either do
;; not write any output files or implement restrictions
;; similar to or higher than
;; [Core]AllowUnsafeOutputFiles=true.
;; They also have no features to invoke arbitrary other
;; programs, and no known exploitable bugs. All to the best
;; of our knowledge. They also have practical use for being
;; called from TeX.
AllowedShellCommands[] = miktex-bibtex
AllowedShellCommands[] = miktex-bibtex8
AllowedShellCommands[] = miktex-epstopdf
AllowedShellCommands[] = miktex-gregorio
AllowedShellCommands[] = miktex-kpsewhich
AllowedShellCommands[] = miktex-makeindex
AllowedShellCommands[] = bibtex
AllowedShellCommands[] = bibtex8
AllowedShellCommands[] = epstopdf
AllowedShellCommands[] = extractbb
AllowedShellCommands[] = findtexmf
AllowedShellCommands[] = gregorio
AllowedShellCommands[] = kpsewhich
AllowedShellCommands[] = makeindex
AllowedShellCommands[] = texosquery-jre8

;; Do we allow TeX \input or \openin on file names starting
;; with `.' (e.g., .rhosts) or outside the current tree (e.g.,
;; /etc/passwd)?
AllowUnsafeInputFiles = true

;; Do we allow TeX \openout on file names starting with `.'
;; (e.g., .rhosts) or outside the current tree (e.g.,
AllowUnsafeOutputFiles = false

AutoAdmin = ?

CommonLinkTargetDirectory =

UserLinkTargetDirectory =

[Core.FileTypes.afm]

Paths[] = .
Paths[] = %R/fonts/afm/

EnvVars[] = AFMFONTS
EnvVars[] = TEXFONTS

Extensions[] = .afm

[Core.FileTypes.base]

Paths[] = .
Paths[] = %r/miktex/data/le

Extensions[] = .base

[Core.FileTypes.bib]

Paths[] = .
Paths[] = %R/bibtex/bib/

EnvVars[] = BIBINPUTS
EnvVars[] = TEXBIB

Extensions[] = .bib

[Core.FileTypes.bst]

Paths[] = .

[Core.FileTypes]
Paths[] = .
Paths[] = %R/bibtex/{bst,csf}//

;; Environment variables to be used for searching BibTeX
;; style files.
EnvVars[] = BSTINPUTS

;; BibTeX style file name extensions.
Extensions[] = .bst

[Core.FileTypes.cid maps]

;; Search path for CID map files.
Paths[] = .
Paths[] = %R/fonts/cid//

;; Environment variables to be used for searching CID map
;; files.
EnvVars[] = FONTCIDMAPS

;; CID map file name extensions.
Extensions[] = .cid
Extensions[] = .cidmap

[Core.FileTypes.clua]

;; Search path for dynamic libraries for Lua.
Paths[] = .
Paths[] = %R/scripts/($progname,$engine,)/lua//

;; Environment variables to be used for searching dynamic
;; libraries for Lua.
EnvVars[] = CLUAINPUTS

;; File name extensions for Lua dynamic libraries.
Extensions[] = .dll
Extensions[] = .so

[Core.FileTypes.cmap files]

;; Search path for character map files.
Paths[] = .
Paths[] = %R/fonts/cmap//

;; Environment variables to be used for character map files.
EnvVars[] = CMAPFONTS
EnvVars[] = TEXFONTS

[Core.FileTypes.cweb]

;; Search path for CWeb input files.
Paths[] = .
Paths[] = %R/cweb//
Run-Time Defaults

;; Environment variables to be used for searching Cweb input files.
EnvVars[] = CWEBINPUTS

;; CWeb file name extensions.
Extensions[] = .w

[Core.FileTypes.dvi]

;; Search path for DVI files.
Paths[] = .
Paths[] = %R/doc/

;; DVI file name extensions.
Extensions[] = .dvi

[Core.FileTypes.dvips config]

;; Search path for Dvips configuration files.
Paths[] = .
Paths[] = %R/dvips/

;; Environment variables to be used for searching Dvips configuration files.
EnvVars[] = TEXCONFIG

[Core.FileTypes.enc]

;; Search path for encoding vector files.
Paths[] = .
Paths[] = %R/fonts/enc/
Paths[] = %R/miktex/config/
Paths[] = %R/dvips/
Paths[] = %R/pdftex/
Paths[] = %R/dvipdfm/

;; Environment variables to be used for searching encoding vector files.
EnvVars[] = ENCFONTS
EnvVars[] = TEXFONTS

;; Encoding vector file name extensions.
Extensions[] = .enc

[Core.FileTypes.font feature files]

;; Search path for font feature files.
Paths[] = .
Paths[] = %R/fonts/fea/

;; Environment variables to be used for searching font feature files.
EnvVars[] = FONTFEATURES
Run-Time Defaults

;;; Font feature file name extensions.
Extensions[] = .fea

[Core.FileTypes.fmt]

;;; Search path for TeX memory dump files.
Paths[] = .
Paths[] = %r/miktex/data/le/{$engine,}

;;; TeX memory dump file name extensions.
Extensions[] = .fmt

[Core.FileTypes.gf]

;;; Search path for generic font bitmap files.
Paths[] = .
Paths[] = %R/fonts//

;;; Environment variables to be used for searching generic font
;;; bitmap files.
EnvVars[] = GFFONTS
EnvVars[] = GLYPHFONTS
EnvVars[] = TEXFONTS

;;; Generic font bitmap file name extensions.
Extensions[] = .gf

[Core.FileTypes.bitmap font]

;;; Search path for bitmap font files.
Paths[] = .
Paths[] = %R/fonts//

;;; Environment variables to be used for searching bitmap font
;;; files.
EnvVars[] = GLYPHFONTS
EnvVars[] = TEXFONTS

[Core.FileTypes.graphic/figure]

;;; Search path for figure files.
Paths[] = .
Paths[] = %R/dvips//
Paths[] = %R/pdftex//
Paths[] = %R/tex//

;;; Environment variables to be used for searching figure
;;; files.
EnvVars[] = TEXPICTS
EnvVars[] = TEXINPUTS

;;; Figure file name extensions.
Extensions[] = .eps
Extensions[] = .epsi
Extensions[] = .png

[Core.FileTypes.hbf]

;; Search path for HBF files.
Paths[] = .
Paths[] = %R/fonts/misc/hbf//

;; HBF file name extensions.
Extensions[] = .hbf

[Core.FileTypes.ist]

;; Search path for MakeIndex style files.
Paths[] = .
Paths[] = %R/makeindex//

;; Environment variables to be used for searching MakeIndex
;; style files.
EnvVars[] = TEXINDEXSTYLE
EnvVars[] = INDEXSTYLE

;; MakeIndex style file name extensions.
Extensions[] = .ist

[Core.FileTypes.lig files]

;; Search path for ligature definition files.
Paths[] = .
Paths[] = %R/fonts/lig//

;; Environment variables to be used for searching ligature
;; definition files.
EnvVars[] = TEXFONTS

;; Ligature definition file name extensions.
Extensions[] = .lig

[Core.FileTypes.ls-R]

;; Search path for Web2c file name database files.
Paths[] = %R

;; Environment variables to be used for searching Web2C file
;; name database files.
EnvVars[] = TEXMFDBS

[Core.FileTypes.lua]

;; Search path for Lua files.
Paths[] = .
Paths[] = %R/scripts/($proname,$engine,){lua,} //
Paths[] = %R/tex/($proname,generic,) //
Run-Time Defaults

;; Environment variables to be used for searching Lua files.
EnvVars[] = LUAINPUTS

;; File name extensions for Lua files.
Extensions[] = .lua
Extensions[] = .luatex
Extensions[] = .luc
Extensions[] = .luctex
Extensions[] = .texlua
Extensions[] = .texluc
Extensions[] = .tlu

;; Environment variables to be used for searching Lua files.
EnvVars[] = LUAINPUTS

[Core.FileTypes.map]

;; Search path for font map files.
Paths[] = .
Paths[] = %R/fonts/map/{$progname,pdftex,dvips,}//
Paths[] = %R/miktex/config//
Paths[] = %R/dvips//
Paths[] = %R/pdftex//
Paths[] = %R/dvipdfm//

;; Environment variables to be used for searching font map files.
EnvVars[] = TEXFONTMAPS
EnvVars[] = TEXFONTS

;; Font map file name extensions.
Extensions[] = .map

[Core.FileTypes.mem]

;; Search path for MetaPost memory dump files.
Paths[] = .

;; MetaPost memory dump file name extensions.
Extensions[] = .mem

[Core.FileTypes.mf]

;; Search path for METAFONT input files.
Paths[] = .
Paths[] = %R/metafont//
Paths[] = %R/fonts/source//

;; Environment variables to be used for searching METAFONT input files.
EnvVars[] = MFINPUTS

;; METAFONT file name extensions.
Extensions[] = .mf
[Core.FileTypes.mfpool]

;; Search path for METAFONT program string files.
Paths[] = .

;; Environment variables to be used for searching METAFONT
;; program string files.
EnvVars[] = MFPOOL
EnvVars[] = TEXMFINI

;; METAFONT program string file name extensions.
Extensions[] = .pool

[Core.FileTypes.mft]

;; Search path for MFT style files.
Paths[] = .
Paths[] = %R/mft//

;; Environment variables to be used for searching MFT style
;; files.
EnvVars[] = MFTINPUTS

;; MFT style file name extensions.
Extensions[] = .mft

[Core.FileTypes.misc fonts]

;; Search path for font related files.
Paths[] = .
Paths[] = %R/fonts/misc//

;; Environment variables to be used for font related
;; files.
EnvVars[] = MISCFonts
EnvVars[] = TEXFONTS

[Core.FileTypes.mlbib]

;; Search path for MlBibTeX database files.
Paths[] = .
Paths[] = %R/bibtex/bib/{mlbib,}//

;; Environment variables to be used for searching MlBibTeX
;; databases files.
EnvVars[] = MLBIBINPUTS
EnvVars[] = BIBINPUTS
EnvVars[] = TEXBIB

;; MlBibTeX database file name extensions.
Extensions[] = .mlbib
Extensions[] = .bib
[Core.FileTypes.mlbst]

;; Search path for MlBibTeX style files.
Paths[] = .
Paths[] = %R/bibtex/{mlbst,bst}:/// 

;; Environment variables to be used for searching MlBibTeX
;; style files.
EnvVars[] = MLBSTINPUTS
EnvVars[] = BSTINPUTS

;; MlBibTeX style file name extensions.
Extensions[] = .bst

[Core.FileTypes.mp]

;; Search path for MetaPost input files.
Paths[] = .
Paths[] = %R/metapost:/// 

;; Environment variables to be used for searching MetaPost
;; input files.
EnvVars[] = MPINPUTS

;; MetaPost file name extensions.
Extensions[] = .mp

[Core.FileTypes.mppool]

;; Search path for MetaPost program string files.
Paths[] = .

;; Environment variables to be used for searching MetaPost
;; program string files.
EnvVars[] = MPPOOL
EnvVars[] = TEXMFINI

;; MetaPost program string file name extensions.
Extensions[] = .pool

[Core.FileTypes.MetaPost support]

;; Search path for MetaPost support files.
Paths[] = .
Paths[] = %R/metapost/support:/// 

;; Environment variables to be used for searching MetaPost
;; support files.
EnvVars[] = MPSUPPORT

[Core.FileTypes.ocp]

;; Search path for Omega compiled process files.
Paths[] = .
Run-Time Defaults

Paths[] = %R/omega/ocp//

;; Environment variables to be used for searching Omega
;; compiled process files.
EnvVars[] = OCPINPUTS

;; Omega compiled process file name extensions.
Extensions[] = .ocp

[Core.FileTypes.ofm]

;; Search path for Omega font metric files.
Paths[] = .
Paths[] = %R/fonts/ofm//
Paths[] = %R/fonts/tfm//

;; Environment variables to be used for searching Omega
;; font metric files.
EnvVars[] = OFMFONTS
EnvVars[] = TEXFONTS

;; Omega font metric file name extensions.
Extensions[] = .ofm
Extensions[] = .tfm

[Core.FileTypes.opl]

;; Search path for Omega property list files.
Paths[] = .
Paths[] = %R/fonts/opl//

;; Environment variables to be used for searching Omega
;; property list files.
EnvVars[] = OPLFONTS
EnvVars[] = TEXFONTS

;; Omega property list file name extensions.
Extensions[] = .opl

[Core.FileTypes.otp]

;; Search path for Omega translation process files.
Paths[] = .
Paths[] = %R/fonts/otp//

;; Environment variables to be used for searching Omega
;; translation process files.
EnvVars[] = OTPINPUTS

;; Omega translation process file name extensions.
Extensions[] = .otp

[Core.FileTypes.opentype fonts]
Run-Time Defaults

;; Search path for OpenType font files.
Paths[] = .
Paths[] = %R/fonts/opentype//

;; Environment variables to be used for searching OpenType
;; font files.
EnvVars[] = OPENTYPEFONTS
EnvVars[] = TEXFONTS

;; OpenType font file name extensions.
Extensions[] = .otf

[Core.FileTypes.ovf]

;; Search path for Omega virtual font files.
Paths[] = .
Paths[] = %R/fonts/ovf//
Paths[] = %R/fonts/vf//

;; Environment variables to be used for searching Omega
;; virtual font files.
EnvVars[] = OVFFONTS
EnvVars[] = TEXFONTS

;; Omega virtual font file name extensions.
Extensions[] = .ovf

[Core.FileTypes.ovp]

;; Search path for Omega virtual property list files.
Paths[] = .
Paths[] = %R/fonts/ovp//

;; Environment variables to be used for searching Omega
;; virtual property list files.
EnvVars[] = OVPFONTS
EnvVars[] = TEXFONTS

;; Omega virtual property list file name extensions.
Extensions[] = .ovp

[Core.FileTypes.pdftex config]

;; Search path for pdfTeX configuration files.
Paths[] = .
Paths[] = %R/pdftex/($proname,//

;; Environment variables to be used for searching pdfTeX
;; configuration files.
EnvVars[] = PDFTEXCONFIG

[Core.FileTypes.pk]

;; Search path for packed bitmap font files.

104
Run-Time Defaults

Paths[] = .
Paths[] = %R/fonts/

;; Packed bitmap font file name extensions.
Extensions[] = .pk

[Core.FileTypes.other binary files]

;; Search path for program binary files.
Paths[] = .
Paths[] = %R/$projname/

[Core.FileTypes.other text files]

;; Search path for program text files.
Paths[] = .
Paths[] = %R/$projname/

[Core.FileTypes.PostScript header]

;; Search path for downloadable PostScript files.
Paths[] = .
Paths[] = %R/miktex/config/
Paths[] = %R/dvips/
Paths[] = %R/pdftex/
Paths[] = %R/dvipdfm/
Paths[] = %R/fonts/enc/
Paths[] = %R/fonts/type1/
Paths[] = %R/fonts/type42/
Paths[] = %R/fonts/type3/
Paths[] = $psfontdirs

;; Environment variables to be used for searching downloadable
;; PostScript files.
EnvVars[] = TEXPSHEADERS
EnvVars[] = PSHEADERS

;; Downloadable PostScript file name extensions.
Extensions[] = .pro
Extensions[] = .enc

[Core.FileTypes.texmfscripts]

;; Search path for architecture-independent executables.
Paths[] = .
Paths[] = %R/scripts/($projname,$engine)00

;; Environment variables to be used for searching
;; architecture-independent executables.
EnvVars[] = TEXMFSCRIPTS

[Core.FileTypes.subfont definition files]

;; Search path for subfont definition files.
Paths[] = .
Paths[] = %R/fonts/sfd//

;; Environment variables to be used for searching subfont
;; definition files.
EnvVars[] = SFDFONTS
EnvVars[] = TEXFONTS

;; Subfont definition file name extensions.
Extensions[] = .sfd

[Core.FileTypes.tcx]

;; Search path for TCX files.
Paths[] = .
Paths[] = %R/miktex/config
Paths[] = %R/miktex/web2c

;; TCX file name extensions.
Extensions[] = .tcx

[Core.FileTypes.tex]

;; Search path for TeX input files.
Paths[] = .
Paths[] = %R/tex/{$progname,generic,} //

;; Environment variables to be used for searching TeX input
;; files.
EnvVars[] = TEXINPUTS

;; TeX input file name extensions.
Extensions[] = .tex

[Core.FileTypes.texpool]

;; Search path for TeX program string files.
Paths[] = .

;; Environment variables to be used for searching METAFONT
;; program string files.
EnvVars[] = TEXPOOL
EnvVars[] = TEXMFINI

;; TeX program string file name extensions.
Extensions[] = .pool

[Core.FileTypes.TeX system sources]

;; Search path for source files.
Paths[] = .
Paths[] = %R/source/

;; Environment variables to be used for searching source
Run-Time Defaults

;; files.
EnvVars[] = TEXSOURCES

[Core.FileTypes.TeX system documentation]

;; Search path for documentation files.
Paths[] = .
Paths[] = %R/doc/miktex/
Paths[] = %R/doc/

;; Environment variables to be used for searching ;; documentation files.
EnvVars[] = TEXDOCS

;; Documentation file name extensions.
Extensions[] = .pdf
Extensions[] = .html
Extensions[] = .md
Extensions[] = .txt
Extensions[] = .ps
Extensions[] = .dvi

[Core.FileTypes.tfm]

;; Search path for TeX font metric files.
Paths[] = .
Paths[] = %R/fonts/tfm/

;; Environment variables to be used for searching TeX font ;; metric files.
EnvVars[] = TFMFONTS
EnvVars[] = TEXFONTS

;; TeX font metric file name extensions.
Extensions[] = .tfm

[Core.FileTypes.troff fonts]

;; Environment variables to be used for searching Troff font ;; files.
EnvVars[] = TRFONTS

[Core.FileTypes.trueype fonts]

;; Search path for TrueType font files.
Paths[] = .
Paths[] = %R/fonts/truetype/

;; Environment variables to be used for searching TrueType ;; font files.
EnvVars[] = TTFONTS
EnvVars[] = TEXFONTS

;; TrueType font file name extensions.
Extensions[] = .ttf
Extensions[] = .ttc

[Core.FileTypes.type1 fonts]

;; Search path for Type1 font files.
Paths[] = .
Paths[] = %R/fonts/type1//

;; Environment variables to be used for searching Type1 font files.
EnvVars[] = T1FONTS
EnvVars[] = T1INPUTS
EnvVars[] = TEXFONTS
EnvVars[] = TEXPSHEADERS
EnvVars[] = PSHEADERS

;; Type1 font file name extensions.
Extensions[] = .pfb
Extensions[] = .pfa

[Core.FileTypes.type42 fonts]

;; Search path for Type42 font files.
Paths[] = .
Paths[] = %R/fonts/type42//

;; Environment variables to be used for searching Type42 font files.
EnvVars[] = T42FONTS
EnvVars[] = TEXFONTS

;; Type42 font file name extensions.
Extensions[] = .t42
Extensions[] = .T42

[Core.FileTypes.vf]

;; Search path for TeX virtual font files.
Paths[] = .
Paths[] = %R/fonts/vf//

;; Environment variables to be used for searching TeX virtual font files.
EnvVars[] = VFFONTS
EnvVars[] = TEXFONTS

;; TeX virtual font file name extensions.
Extensions[] = .vf

[Core.FileTypes.web2c files]

;; Search path for Web2c files.
Paths[] = .
Paths[] = %R/web2c/

[Core.FileTypes.web]

;; Search path for WEB input files.
Paths[] = .
Paths[] = %R/web/

;; Environment variables to be used for searching WEB input files.
EnvVars[] = WEBINPUTS

;; CWeb file name extensions.
Extensions[] = .web

[MPM]

;; Install packages for all users.
AutoAdmin = ?

;; Install missing packages automatically (on-the-fly).
AutoInstall = ?

[TeXandFriends]

;; Create auxiliary directory if '--aux-directory=DIR' refers to a non-existing directory.
CreateAuxDirectory = t

;; Create the output directory if '--output-directory=DIR' refers to a non-existing directory.
CreateOutputDirectory = t

;; Enable file:line:error style messages.
CStyleErrors = f

All TeXMF Programs

;; TeX uses the buffer to contain input lines, but macro expansion works by writing material into the buffer and reparsing the line.
;; As a consequence, certain constructs require the buffer to be very large, even though most documents can be handled with a small value.
buf_size = 200000

;; Width of context lines on terminal error messages.
error_line = 79

;; Extra low memory for boxes, glue, breakpoints, etc.
extra_mem_bot = 0

;; Extra high memory for chars, tokens, etc.
extra_mem_top = 0
Run-Time Defaults

;; Width of first lines of contexts in terminal error messages; should
;; be between 30 and (error_line - 15).
half_error_line = 50

;; Words of inimemory available.
main_memory = 3000000

;; Width of longest text lines output; should be at least 60.
max_print_line = 79

;; Maximum number of strings.
max_strings = 500000

;; Maximum number of simultaneous macro parameters.
param_size = 10000

;; Pool space free after format loaded.
pool_free = 47500

;; Max number of characters in all strings, including all error
;; messages, help texts, font names, control sequences. These values
;; apply to TeX and MP.
pool_size = 3250000

;; Maximum number of simultaneous input sources.
stack_size = 5000

;; Strings available after format loaded.
strings_free = 100

;; Minimum pool space after TeX/MP's own strings; must be at least
;; 25000 less than pool_size, but doesn't need to be nearly that
;; large.
string_vacancies = 90000

All TeX Programs

;; Maximum number of input files and error insertions that can be
;; going on simultaneously.
max_in_open = 50

;; Maximum number of semantic levels simultaneously active.
nest_size = 500

;; Space for saving values outside current group.
save_size = 50000

;; Space for hyphenation patterns.
trie_size = 700000

;; Total number of fonts.
font_max = 9000

110
;;; Words of font info for TeX (total size of all TFM files,    
;;; approximately).
font_mem_size = 3000000

;;; Extra space for the hash table of control sequences (which allows  
;;; 10K names as distributed).
hash_extra = 200000

;;; Prime number of hyphenation exceptions.
hyph_size = 8191

;;; Size of the output buffer; must be a multiple of 8.
dvi_buf_size = 8192

**Omega**

ocp_buf_size = 500000
ocp_listinfo_size = 1000
ocp_list_list_size = 1000
ocp_lstack_size = 1000
ocp_stack_size = 10000
trie_op_size = 35111

**pdfTeX**

pdf_mem_size = 10000
obj_tab_size = 1000
dest_names_size = 131072
pdf_os_buf_size = 1

**METAFONT & MetaPost**

;;; Size of stack for bisection algorithms; should probably be left at  
;;; this value.
bistack_size = 1500

;;; Maximum number of ligature/kern steps, must be at least 255 and at  
;;; most 32510.
lig_table_size = 15000

;;; Maximum number of knots between breakpoints of a path.
path_size = 10000

**METAFONT**

;;; Number of autorounded points per cycle.
max_wiggle = 1000

;;; Space for storing moves in a single octant.
move_size = 20000
MetaPost

;;; Number of words for TFM information for text fonts.
font_mem_size = 10000
Index

Symbols
--admin, 29, 43
--alias=name, 15, 17, 25, 27, 33, 37, 47, 52, 60, 66
--aux-directory=dir, 7, 33, 37, 52, 60, 66
--batch, 64
--bistack-size=n, 37
--buf-size=n, 37, 52, 60, 66
--c-style-errors, 33, 37, 47, 52, 60, 66
--clean, 64
--credits, 33
--debug, 47
--debug-format, 33
--default-paper-size=paper, 29
--disable-8bit-chars, , ,
--disable-installer, 6, 15, 17, 29, 33, 37, 53, 60, 66
--disable-pipes, 53, 60, 66
--disable-write18, 33, 53, 60, 66
--dont-parse-first-line, 37, 53, 60, 66
--draftmode, 33, 53
--dump, 29
--edit-config-file=file, 29
--enable-8bit-chars, , ,
--enable-enctex, 53, 60
--enable-etex, 53, 66
--enable-installer, 6, 15, 17, 29, 33, 37, 53, 60, 67
--enable-mltex, 53, 60, 67
--enable-pipes, 53, 60, 67
--enable-write18, 33, 53, 60, 67
--error-line=n, 37, 53, 60, 67
--even-only, 51
--expand, 64
--extra-mem-bot=n, 53, 60, 67
--extra-mem-top=n, 53, 61, 67
--file-line-error, 33, 37, 47, 52, 60, 66
--file-line-error-style, 33, 37, 47, 52, 60, 66
--file-type=filetype, 25
--find-updates, 43
--find-upgrades, 43
--font-max=n, 53, 61, 67
--font-mem-size=n, 53, 61, 67
--force, 29
--gray-font=font, 27
--half-error=n, 37, 53, 61, 67
--halt-on-error, 33, 37, 47, 53, 61, 67
--hash-extra=n, 53, 61, 67
--help, 15, 17, 25, 27, 33, 37, 43, 47, 53, 61, 64, 67
--hhelp, 15, 17, 27, 38, 43, 53, 61, 67
--import-all, 44
--import=package, 44
--include-directory=dir, 7, 15, 17, 28, 33, 38, 53, 61, 67
--initialize, 33, 38, 47, 54, 61, 67
--install=@listfile, 44
--install=packagelist, 44
--interaction=mode, 33, 38, 47, 54, 61, 67
--job-name=name, 33, 38, 47, 54, 61, 67
--job-time=file, 38, 54, 61, 67
--label-font=font, 28
--landscape, 51
--language=lang, 64
--lib-table-size=n, 38
--list, 44
--list-file-types, 25
--list-formats, 30
--list-modes, 30
--list-package-names, 44
--logo-font=font, 28
--lua=file, 32
--luaonly, 33
--luarend, 32
--mag=mag, 17
--main-memory=n, 38, 54, 61, 67
--max-in-open=n, 54, 61, 68
--max-pages=n, 17
--max-print-line=n, 38, 54, 61, 68
--max-strings=n, 38, 54, 61, 68
--max-size=n, 38
--min-crossrefs=n, 15
--mkidx-option=option, 64
--mklangs, 30
--mklinks, 30
--mklinks=category, 30
--mkmaps, 30
--mktex=fmt, 33
--move-size=n, 38
--must-exist, 25
--nest-size=n, 54, 61, 68
--no-c-style-errors, 33, 38, 48, 54, 61, 68
--no-file-line-error, 33, 38, 48, 54, 61, 68
--no-file-line-error-style, 33, 38, 48, 54, 61, 68
--no-mktex=fmt, 33
--no-pdf, ,
--no-registry, 58
--no-shell-escape, , ,
--nosocket, 34
--numbersystem=string, 48
--odd-only, 51
--output-comment=string, 34
--output-directory=dir, 34, 38, 48, 54, 61, 68
--output-driver=cmd, 34
--output-format=format, 34
--overflow-label-offset=real, 28
--package-level=level, 44
--page-range=range, 44
--page-start=pagespec, 17

113
--papersize=string,
--param-size=n, 38, 54, 61, 68
--parse-first-line, 38, 54, 61, 68
--path-size=n, 38
--pdf, 64
--pool-free=n, 54, 62, 68
--pool-size=n, 38, 54, 62, 68
--print-method=method, 51
--print-nothing, 51
--print-only, 30
--print-package-info=package, 44
--printer=printer, 51
--quiet, 30,
--record-package-usages=file, 6, 16, 17, 28, 38, 54, 62, 68
--recorder, 34, 39, 48, 54, 62, 68
--register-root, 30
--remove-links, 30
--report, 30
--require=@listfile, 44
--require=packagelist, 44
--restrict-write18, 34, 39, 54, 62, 68
--restricted, 48
--run-viewer, 64
--safer, 34
--save-size=n, 54, 62, 68
--screen, 39
--select=sel, 17
--set-config-value=[section]valuename=value, 30
--set-repository=location, 45
--shell-escape, 39, 55, 62, 68
--shell-restricted, 39, 55, 62, 68
--show-config-value=[section]valuename, 30
--show-path=filetype, 25
--src, 64
--src-specials, 39, 54, 62, 68
--start, 25
--string-vacancies=n, 39, 55, 62, 68
--syncTeX=n, 34, 55, 68
--tcx=tcxname, 39, 55, 62
--tex-option=option, 48
--tex=TEXprogram, 48
--texinfo=cmd, 65
--the-name-of-the-game=engine, 25
--time-statistics, 39, 55, 62, 69
--trace=tracestreams, 16, 18, 28, 39, 41, 45, 55, 62, 69
--translate-file=tcxname, 48
--trie-size=n, 55, 62, 69
--troc, 48
--undump=name, 34, 39, 48, 55, 62, 69
--uninstall=package, 45
--unregister-root, 30
--update, 45
--update-db, 45
--update-fndb, 30
--update=@listfile, 45
--update=packagelist, 45
--upgrade, 45
--utc, 34
--verbose, 31, 65
--verify, 45
--verify=@listfile, 45
--verify=packagelist, 45
--version, 16, 18, 25, 28, 31, 34, 39, 41, 45, 48, 49, 55, 62, 65, 69
--viewer-option=option, 65
--max-iterations=n, 64
-s internal=NUMBER, 48
-s internal=STRING, 48
-slant-font=font, 28
-src, 54, 62, 68
-title-font=font, 28
.bbl, 15
.bbl (bibliography file), 15
.bib (bibliographic database), 15
.bst (bibliography style file), 15
.dvi (DVI file), 19
\write18 (run a program from within TeX), 39

A
ADOBE fonts, 89
automatic package installation, 6
auxiliary files
removing, 7
specifying the directory for, 7

B
BIBINPUTS, 16, 91
bibliography
creation, 15
BibTeX
invoked by texify, 7
invoked by texify, 7

C
cross-references
minimum number required, 15
resolving, 7

D
DVI search, 8

E
error messages
Index

C/C++ style, 33, 37, 47, 52, 60, 66
line width, 37, 53, 60, 67
extra_mem_bot, 11

F
file name database
defined, 9
refreshing, 9
file name database refreshing,
font_mem_size, 11
format files
creating,

G
gftodvi, 19
give back, 3

K
Knuth
Donald E., 3

L
LaTeX
invoked by texify, 7
local additions
intergrating, 9
local guide, 6

M
main_memory, 11
MakeIndex
invoked by texify, 7
memory dump files
creating,
METAFONT
modes,
MPINPUTS, 39, 91, 91
MiKTeX
features, 3
going, 3
how to pronounce, 3
registering, 3
updating, 9
MiKTeX Console
running, 9
update wizard, 9
MiKTeX project
how to support the, 3
MiKTeX Project Page, 3
MICKTEX_EDITOR, 34, 39, 55, 62, 69
MICKTEX_REPOSITORY, 46, 91

MIKTEX_TRACE, 16, 18, 24, 28, 35, 39, 46, 48, 49, 50,
55, 62, 69, 91
modes.mf, 36
MPINPUTS, 48, 48
mthelp, 4

O
Outline fonts
psfonts.map, 11

P
package set, 41, 58
packages
automatic installation of, 9
finding out usages, 6
updating, 9
paper format
setting, 9
paper size
changing, 88
setting default,
PDF
distilling into, 24
pdfLaTeX
invoked by texify, 7
portable, 41
PostScript
converting TeX output into, 19
PostScript Type 1 fonts
psfonts.map, 11
private installation, 58
project page, 3
psfonts.map, 23
creating, 11,

R
registration, 3

S
security, 23
setup wizard
setupwiz.opt, 57
setupwiz.opt, 57
shared installation, 58
startup configuration file, 58

T
texify, 7
TEXINPUTS, 56, 56, 63, 69, 91
TEXMF root
registering,
unregistering,
TFMfonts, 56, 63, 69, 91
unattended setup, 40, 58
updates
installing, 9
updmap.cfg, 89
dvipdfmDownloadBase14, 90
dvipsDownloadBase35, 89
dvipsPreferOutline, 89
instructions, 89
LW35, 89
ADOBE, 89
ADOBEkb, 89
URW, 89
URWkb, 89
Map, 90
MixedMap, 90
pdftexDownloadBase14, 90
URW fonts, 89

virtual fonts
resolving, 17