MiKTeX Manual

Christian Schenk <cschenk@snafu.de>

1 What is MiKTEX?

MiKTEX is a free TEX distribution for Windows (9x/NT).

MiKTEX Features

- Native Windows implementation with support for long filenames.
- On-the-fly generation of missing font files.
- TDS (TEX directory structure) compliant.
- Free distribution, i.e. full source code is available.
- TEX compiler features:
    - The compiler is able to insert source file information into the DVI file. This feature can improve Editor/Previewer interaction.
    - The compiler can read compressed input files.
    - The input encoding can be changed via TCX tables.
- Previewer features:
    - Supports graphics (PostScript, BMP, WMF, . . .)
    - Supports PostScript fonts
    - Anti-aliasing of text
    - Understands HyperTEX (`html:`) specials
    - Understands source (`src:`) specials
    - Customizable magnifying glasses
- MiKTEX is network friendly:
    - integrates well into a heterogeneous TEX environment
    - supports UNC filenames
    - supports multiple TEXMF directory trees
    - uses a filename database for efficient file access

Components

The MiKTEX consists of the following applications:

TEX 3.14159
  The classic TEX compiler.

e-TEX 2.1   A feature-extended version of TEX.

Yap 0.95d   The MiKTEX DVI previewer.

pdfTEX 0.13d
  Creates PDF files from TEX documents.

dvipdfm 0.10.5
  Converts DVI files into PDF documents.

Omega 1.8
> An enhanced version of TeX with support for 16-bit character sets.

METAFONT 2.718
> Converts font specifications into raster fonts.

MetaPost 0.641
> Converts picture specifications into PostScript commands.

dvips 5.83   Converts DVI files into PostScript.

MakeIndex 2.12
> Composes indexes.

BibTeX 0.99c
> Composes bibliographies.

Standard LaTeX Packages
> AMS-LaTeX, Babel, PSNFSS, ...

TeXware, METAFONTware, PSutils, . . .
> Lots of utilities.

1.1 How to get MiKTeX

MiKTeX Distribution

You can download the MiKTeX distribution from the CTAN[1] directory

    systems/win32/miktex/

Other Packages

Here is a list of other packages you should take into consideration:

Aladdin GhostScript 5.50 (`http://www.cs.wisc.edu/~ghost/aladdin/index.html`).
> GhostScript is an interpreter for the PostScript language. The DVI previewer Yap uses GhostScript to display EPS inclusions.

Adobe Acrobat Reader (`http://www.adobe.com/prodindex/acrobat/readstep.html`).
> You can use it two view PDF files.

WinEdt (`http://home.istar.ca/~winedt`)
> WinEdt is a shareware TeX editor/shell. It coorperates with MiKTeX.

ActivePerl (`http://www.activestate.com`)
> ActivePerl is an implementation of Perl for the Win32 platform. A few MiKTeX utilities (e.g. `psmerge`) are Perl scripts. Therefore you will need Perl if you want to use these utilities.

---

[1] CTAN: Comprehensive TeX Archive Network

1.2 The MiKT_EX Project Page

Visit the MiKTeX Project Page (`http://www.snafu.de/~cschenk/miktex`) for information about new releases, patches and so on.

1.3 The MiKT_EX Mailing List

MiKT_EX Mailing List

Claus Ekstroem from Denmark has created a discussion list for MiKT_EX. To join this list, send an e-mail to <miktex-request@dsts.dk> which contains the word **subscribe** as the first line in the message body.

This list is archived at www.egroups.com (`http://www.egroups.com/list/miktex`).

1.4 Documentation

The MiKT_EX Manual (which you are reading right now) concentrates on documenting MiKT_EX specific details.

The 'doc' directory hierarchy contains user manuals for various applications.

2 What's new in MiKTEX 1.20a?

The source archives (`source*.zip`) were removed from the normal distribution. They are now in the CTAN directory `systems/win32/miktex/source`.

Updated Apps

- pdfTEX 0.13d
- Dvipdfm 0.10.5
- Babel 3.6m

New Apps

- URW PostScript fonts; these are free replacements for the standard Adobe fonts.

Resolved Problems

- Hyphenation bug

3 Installing MiKTEX

1. If you still have MiKTEX 1.11 installed on you computer, then you should use the uninstall option of MiKTEX 1.11.

2. Make sure that you have enough disk space. A complete MiKTEX installation consumes approximately 40MB of disk space.

3. It is highly recommended that you login as Administrator, if you're installing MiKTEX on a Windows NT computer.

4. Choose a location for the installation directory, say `c:\texmf`. This directory receives the files of the MiKTEX distribution.

5. You can cause MiKTEX to deposit newly created files (fonts, memory dumps, filename databases) in a separate directory tree. This directory tree is called the *Local TEXMF Tree*. If you decide to create such a tree, then you must choose a name for its root directory, say `c:\localtexmf`.

   Benefits that a local tree provides include the following:

   - Fast file search: MiKTEX assumes that only the local tree can receive new fonts and the like, i.e. MiKTEX can trust in the filename database when the remaining (non-local) trees are searched for a file.

   - You can use the local tree for your own additions (macros, fonts).

   - Easier updates: You don't have to worry about future MiKTEX updates, since the local tree will not be overwritten by the setup program.

   - You can install the MiKTEX distribution on a read-only media.

6. Decide whether you want to incorporate a preexisting TEXMF tree. For example, if you have a TEXLive CD in CDROM drive `e:`, then it is possible to include `e:\texmf` in the MiKTEX search path.

7. Start the MiKTEX Setup Wizard (`setupwiz.exe`).

   a. When prompted for the installation directory, enter the name chosen in step 4.

   b. When prompted for the local TEXMF directory, enter the name chosen in step 5 or check the button '`No local TEXMF tree`', if you don't need a local tree.

   c. When prompted for the list of additional TEXMF root directories, enter a semicolon-separated list of preexisting TEXMF root directories. Check the button '`No preexisting TEXMF directory trees`', if you just want to use the TEXMF tree that comes with MiKTEX.

The Setup Wizard does not change the `PATH` variable, i.e. you have to add something like

```
set PATH=%PATH%;c:\texmf\miktex\bin
```

to your `autoexec.bat`.

4 Uninstalling MiKTEX

There is no uninstall option (yet), so you must do it by hand if you decide to delete MiKTEX:

1. Delete the main TEXMF folder (usually `c:\texmf`) includaing sub folders.
2. Delete the local TEXMF folder (usually `c:\localtexmf`) including sub folders.
3. Delete Registry entries:
    1. Start `regedit`
    2. Open the key `HKEY_LOCAL_MACHINE\SOFTWARE` and delete the `MiK` subkey.
    3. Open the key `HKEY_CURRENT_USER\Software` and delete the `MiK` subkey.
4. Remove the MiKTEX item from the Start menu.
5. Remove the bin directory from the `PATH`.

5 The TEXMF Directory Hierarchy

The setup program creates a TDS-compliant directory hierarchy. Normally this hierarchy is distributed over two physical directory trees:

1. The *Installation Directory* (usually 'c:\texmf') contains all files from the MiKTEX distribution.

2. The *Local Directory* (usually 'c:\localtexmf') receives all files that are created on-the-fly (i.e. after the installation process has finished).

It's possible to redefine theses two directories anytime. See , for more information.

5.1 Installation Directory

The Installation Directory (usually c:\texmf) is the root of a TDS-compliant directory hierarchy. If you have installed the complete distribution, then the Installation Directory contains the following sub-directories:

bibtex, dvips, makeindex, . . .
        These directories contain application related input files.

doc   This directory contains all user documentation.

fonts   This directory contains fonts in various formats.

miktex   The miktex directory is reserved for MiKTEX related files:

    miktex\bin
        Contains all executable files.

    miktex\config
        Contains the global configuration file miktex.ini and the MiKTEX font mapping file miktex.map. The MiKTEX Setup Wizard installs its own log file here.

    miktex\base
        Contains the METAFONT string pool file mf.pool.

    miktex\fmt
        Contains TEX string pool files: etex.pool, pdftex.pool, omega.pool, tex.pool.

    miktex\mem
        Contains the MetaPost string pool file mp.pool.

5.2 Local Directory

The Local Directory (usually `c:\texmf` receives files that are generated on-the-fly. For example, if the TeX compiler needs a TeX Font Metric (TFM) file that is not available yet, then it creates that file (if possible) and installs it in an appropriate sub-directory of the Local Directory.

Typically the Local Directory contains the following sub-directories:

`fonts`   Contains font files that are not part of the MiKTeX distribution, but that were created on-the-fly.

`miktex\config`
      This directory contains the filename database files.

6 MiKTEX Configuration Files

6.1 The Two Kinds of Configuation Files

MiKTEX configuration parameters are stored in two configuration files:

- The *Global Configuration File* contains site-wide configuration settings (i.e. settings shared by all MiKTEX users). The name of this file is `miktex.ini`. It is located in the directory `miktex\config` relative to the installation root directory (usually `c:\texmf`).
- The *Personal Configuration File* contains per-user configuration settings. The location of the personal configuration file can be defined with the help of `initexmf` (see).

Personal configuration settings override global settings.

6.2 How to specify search paths

Search paths are used by MiKTEX to find special files (such as TEX input files) within a comprehensive directory hierarchy.

A search path is a semicolon-separated list of directory paths. This list is traversed from left to right, i.e. the first directory is searched first.

In a directory path, the following character seqeuences have a special meaning:

`%R`   A placeholder for the list of TEXMF root directories.

`//`   A flag, which auses MiKTEX to search recursively.

Example

Assuming that `c:\texmf;\\myserver\texmf` is the list of TEXMF root directories, the search path `.;%R\tex\latex//;%R\tex\generic//` causes LaTEX to search its input files in the following locations:

1. In the current directory (`.`).
2. In the directory `c:\texmf\tex\latex` and in all directories below it.
3. In the directory `\\myserver\texmf\tex\latex` and in all directories below it.
4. In the directory `c:\texmf\tex\generic` and in all directories below it
5. In the directory `\\myserver\texmf\tex\generic` and in all directories below it.

Testing a new search path

You can use the configuration utility `initexmf` to test whether an input file can be found via the current search path. For example, the command

    initexmf --find-latex-input a4.sty

searches for the LaTEX input file `a4.sty`. The full pathname is printed if the file was found.

6.3 Contents of a MiKTEX Configuration File

A MiKTEX configuration file is divided into several named sections. Each section contains configuration settings for a specific application or feature.

6.3.1 [BibTeX]: BibTEX Configuration Settings

The section `[BibTeX]` contains BibTEX related configuration settings.

`Input Dirs`
> Search path for BibTEX input files (both databases and style files).

`min_crossrefs`
> Minimum number of cross-refs required for automatic `cite_list` inclusion.

6.3.2 [Dvips]: Dvips Configuration Settings

The section `[Dvips]` contains Dvips related configuration settings.

`CONFIGPath`
> Where Dvips searches its configuration files (e.g. `config.ps`).

`ENCPath`   Where Dvips searches for `.enc` files.

`GraphicsPath`
> Where Dvips searches for `.eps` files.

`MAPPath`   Where Dvips searches for `.map` files.

`PSPath`   Where Dvips searches for PS header files.

6.3.3 [Graphics]: Graphics Conversion Rules

The section `[Graphics]` contains graphics conversion rules. Each rule has the syntax

> `.`*fromext*`.`*toext*`=`*commandline*

fromext is the filename extension of the source file. *toext* is the filename extension of the destination file. *commandline* is the command-line which does the conversion. The command-line may include the following placeholders:

`%i`   The name of the input file.

`%o`   The name of the output file.

The standard MiKTEX configuration file contains the following rules:

```
.gif.bmp=giftopnm %i | ppmtobmp -windows > %o
.pcx.bmp=pcxtoppm %i | ppmtobmp -windows > %o
.png.bmp=pngtopnm %i | ppmtobmp -windows > %o
.tga.bmp=tgatoppm %i | ppmtobmp -windows > %o
.tif.bmp=tifftopnm %i | ppmtobmp -windows > %o
.tiff.bmp=tifftopnm %i | ppmtobmp -windows > %o
```

6.3.4 [Magic]: Memory Settings for TeX & Friends

The section `[Magic]` contains memory related configuration settings. The values are used by TeX, pdfTeX and Omega for the dynamic allocation of certain data structures.

Format-Independent Values

The following parameters can be changed at run time to extend or reduce TeX's capacity. They may have different values in `INITEX` and in production versions of TeX.

`mem_min`   Smallest index in TeX's internal `mem` array; must be 0 or more; must be equal to `mem_bot` in `INITEX`, otherwise `<=mem_bot`.

`mem_max`   Greatest index in TeX's internal `mem` array; must be strictly less than 1073741823.

`buf_size`   Maximum number of characters simultaneously present in current lines of open files and in control sequences between `\csname` and `\endcsname`; must not exceed 1073741823.

`error_line`
          Width of context lines on terminal error messages.

`half_error_line`
          Width of first lines of contexts in terminal error messages; should be between 30 and (`error_line` - 15).

`max_print_line`
          Width of longest text lines output; should be at least 60.

`stack_size`
          Maximum number of simultaneous input sources.

`max_in_open`
          Maximum number of input files and error insertions that can be going on simultaneously.

`font_max`   Maximum internal font number; must not exceed 5000.

`font_mem_size`
          Number of words of `font_info` for all fonts.

`param_size`
          Maximum number of simultaneous macro parameters.

`nest_size`
          Maximum number of semantic levels simultaneously active.

`max_strings`
          Maximum number of strings; must not exceed 1073741823.

`string_vacancies`
          The minimum number of characters that should be available for the user's control sequences and font names, after TeX's own error messages are stored.

pool_size

Maximum number of characters in strings, including all error messages and help texts, and the names of all fonts and control sequences; must exceed `string_vacancies` by the total length of TEX's own strings, which is currently about 23000.

save_size

Space for saving values outside of current group; must be at most 1073741823.

trie_size

Space for hyphenation patterns; should be larger for `INITEX` than it is in production versions of TEX.

trie_op_size

Space for "opcodes" in the hyphenation patterns.

Format-Dependent Values

Like the preceding parameters, the following quantities can be changed at run time to extend or reduce TEX's capacity. But if they are changed, it is necessary to rerun the initialization program `INITEX` to generate new tables for the production TEX program. One can't simply make helter-skelter changes to the following constants, since certain rather complex initialization numbers are computed from them.

mem_bot   Smallest index in the mem array dumped by `INITEX`; must not be less than `mem_min`.

mem_top   Largest index in the mem array dumped by `INITEX`; must be substantially larger than 0 and not greater than `mem_max`.

6.3.5 [MakeIndex]: MakeIndex Configuration Settings

The section `[MakeIndex]` contains MakeIndex related configuration settings.

INDEXSTYLE

Search path for MakeIndex style files.

6.3.6 [MakePK]: MakePK Configuration Settings

The section `[MakePK]` contains configuration settings that are related to the auto-creation of packed raster fonts.

DestDir   The specification of a directory where newly created PK (Packed Raster Font) files are to be installed.

The specification may include special character sequences which will be replaced at search-time:

%m   The current METAFONT mode.

%d   The horizontal resolution (in dots per inch).

%s	The font supplier (e.g. `public`).
%t	The typeface name (e.g. `cm`). `typeface.map`.

Admin note: All MiKTEX users must have permission to create files in the specified directory.

6.3.7 [MakeTFM]: MakeTFM Configuration Settings

DestDir   Where new `.tfm` files are to be installed.

The specifiation may contain special character sequences which are replaced at search-time:

%s	The font supplier (e.g. `public`).
%t	The typeface name (e.g. `cm`).

Admin note: MiKTEX users must have permission to add files to the specified directory.

6.3.8 [METAFONT]: METAFONT Configuration Settings

The section `[METAFONT]` contains METAFONT related configuration settings.

Input Dirs
        Search path for METAFONT input files.

6.3.9 [MetaPost]: MetaPost Related Configuration Settings

The section `[MetaPost]` contains MetaPost related configuration settings.

Input Dirs
        Where MetaPost searches for input files.

6.3.10 [MiKTeX]: General Configuration Settings

The section `[MiKTeX]` contains general configuration settings and search path specifications.

General Configuration Settings

Trace   This is a comma separated list of trace options:

`notrace`	Inhibits trace output to the console.
`fndb`	Traces the filename database.

`fileseach`
>
> Traces the find-file machinery.

`access`   Traces file accesses.

`process`   Traces secondary processes.

`tcx`   Traces TCX tables.

`error`   Traces error conditions.

`TraceFile`
>
> The name of the trace file.

Search Path Specifications

`AFMPath`   Used to locate Adobe font metric files (`*.afm`).

`BASEPath`   Used to locate METAFONT base files (`*.base`).

`ENCPath`   Used to locate `*.enc` files.

`EXEPath`   Used to locate executables.

`FMTPath`   Used to locate TEX dump files (`.fmt`). Also used to locate e-TEX dump files (`.efmt`).

`GraphicsPath`
>
> Used to locate graphics files (`*.eps;*.bmp;...`).

`MAPPath`   Used to locate font map files (`*.map`).

`MEMPath`   Used to locate MetaPost memory files (`.mem`).

`OFMPath`   Used to locate Omega font metric files (`*.ofm`).

`OVFPath`   Used to locate Omega virtual fonts (`*.ovf`).

`PKPath`   Used to locate packed font raster files (`*.pk`).

`PSPath`   Used to locate PostScript header files (`*.enc;*.map`);

`TCXPath`   Used to locate character translation files (`.tcx`).

`TFMPath`   Used to locate TEX font metric files (`*.tfm`).

`TTFPath`   Used to locate TrueType fonts (`*.ttc;*.ttf`).

`Type1Path`
>
> Used to locate Type1 fonts (`*.pfa;*.pfb`).

`VFPath`   Used to locate virtual fonts (`*.vf`).

6.3.11 [Omega]: Omega Configuration Settings

The section `[Omega]` contains Omega related configuration settings:

`Input Dirs`
>
> The search path for Omega input files.

`OCPPath`   Where Omega searches for OCP files.

6.3.12 [otp2ocp]: `otp2ocp` Configuration Settings

`Input Dirs`
> Used by `otp2ocp` to locate OTP files (`.otp`).

6.3.13 [pdfTeX]: pdfTEX Configuration Settings

The section `[pdfTeX]` contains pdfTEX related configuration settings.

`Input Dirs`
> Where pdfTEX searches for input files.

`PSPath`   Where pdfTEX searches for font mapping files.

6.3.14 [ps2pk]: `ps2pk` Configuratio Settings

The section `[ps2pk]` contains configuration settings for the `ps2pk` utility:

`PSResPath`
> Where `ps2pk` searches for PS resource files.

6.3.15 [TeX]: TEX Configuration Settings

The section `[TeX]` contains TEX-related configuration settings.

`Editor`   The command to be started when you press `e` in the error menu.

You can use the following placeholders:

`%f`   Will be replaced by the name of the input file that caused the error.

`%h`   Will be replaced by a help text.

`%l`   Will be replaced by the line number.

`%m`   Will be replaced by the error message.

`%t`   Will be replaced by the name of the transcript file.

For example, a suitable value for WinEdt would be `winedt %f -G(1,%l,0) - S(12,+1,0)`.

For NT Emacs, set `Editor` to `gnulientw -F +%l %f`.

`Input Dirs`
> Used by TEX to locate input files.

6.3.16 [Yap]: Yap Configuratuib Settings

`Input Dirs`
> Used by Yap to locate DVI files (`*.dvi`).

7 The MiKTeX Configuration Utility

`initexmf.exe` is the MiKTeX configuration utility. You can use it to

- redefine the list of TEXMF root directories
- refresh the filename database
- update standard dump files (`plain.fmt` and friends)
- define the name/location of a personal configuration file

7.1 Defining TEXMF Root Directories

The standard setup procedure creates two TEXMF root directories:

- `c:\texmf`: the installation directory (see Section 5.1 [Installation Directory], page 7).
- `c:\localtexmf`: the local directory (see Section 5.2 [Local Directory], page 8).

You can redefine the TEXMF root directories by using the command-line switches `--root-directories` and `--local-root`:

`--root-directories=`*dirlist*

> This switch defines the list of TEXMF root directories. *dirlist* is a semicolon-separated list of directory pathnames.

`--local-root=`*dir*

> This switch defines the local directory.

It is necessary to refresh the filename-database whenever you redefine the TEXMF root directories (see Section 7.2 [Maintaining the Filename Database], page 16).

7.2 Maintaining the filename database

To speed up file search, MiKTeX makes use of a list of known file names. This list is called the filename database (fndb). The fndb is spread over several fndb files, one for each TEXMF root directory.

The fndb file for the first TEXMF tree is called `texmf0.fndb`. For the second tree it is called `texmf1.fndb`. And so on.

It is strongly recommended that you update the fndb whenever files are added to or removed from one of the TEXMF trees.

You update all fndb files by invoking `initexmf.exe` with the command line switch `--update-fndb`:

```
initexmf --update-fndb
```

You can update a certain fndb file by specifying the TEXMF root. For example,

```
initexmf --update-fndb=c:\texmf
```

will update the fndb file for the tree rooted at `c:\texmf`.

7.3 Maintaining the PostScript resource database

The PostScript resource database (PSres) is used by some utilities in order to locate PostScript resources (font outlines/metrics/encodings).

The database is located in the MiKTEX config directory (usually '`c:\texmf\miktex\config`'). The name of the database file is '`dpres.dpr`'. It is a text file, so you can view it with a text editor (e.g. wordpad).

It is strongly recommended that you update the database whenever PostScript resources ('`*.pfb;*.afm;*.enc`') are added to or removed from one of the TEXMF trees.

You update the database files by invoking `initexmf.exe` with the command line switch `--mkpsres`:

```
initexmf --mkpsres
```

7.3.1 Incorporating External Font Directories

It is possible to add non-MiKTEX font directories to the resource database. The `--mkpsres` switch accepts as an optional argument the name of an external font directory. You can use several `--mkpsres` switches with on invocation of `initexmf`.

By specifying the command line flag `--search`, you can cause `initexmf` to automatically search your workstation for third party PS resource files (e.g. Acrobat Reader Fonts):

```
initexmf --mkpsres --search
```

7.4 Making Standard Dump Files

Some programs initialize itself by reading parts of the memory from an external file. For the TEX family of programs, such a file is called a *Dump File*.

The MiKTEX configuration file has built-in rules for standard dump files.

Non-standard dump files (i.e. dump files not mentioned in this section) must be created with the ini-version of the program. For example, you would say '`initex texinfo @dump`' to produce a Texinfo dump file (`texinfo.fmt`).

You create standard dump files by invoking `initexmf` with the command line switch `--dump`. This switch takes an optional argument, which is the name of the program for which a new dump file is to be created:

```
initexmf --dump[=program]
```

If *program* is ommited, then all standard dump files will be rebuilt. Otherwise, *program* must be one of the following names:

elatex   This creates the dump file `plain.efmt` which is used by `elatex.exe`.

etex   This creates the dump file `plain.efmt` which is used by `etex.exe`.

lambda   This creates the dump file `lambda.fmt` which is used by `lambda.exe`.

latex   This creates the dump file `latex.fmt` which is used by `latex.exe`.

tex   This creates the dump file `plain.fmt` which is used by `tex.exe`.

metafont This creates the dump file `plain.base` which is used by `mf.exe` (METAFONT).

metapost This creates the dump file `plain.mem` which is used by `mpost.exe` (MetaPost).

omega   This creates the dump file `omega.fmt` which is used by `omega.exe`.

pdflatex This creates the dump file `pdflatex.fmt` which is used by `pdflatex.exe`.

pdftex   This creates the dump file `pdftex.fmt` which is used by `pdftex.exe`.

7.4.1 Controlling which hyphenation patterns are used by LaTeX

You can control the loading of hyphenation patterns by modifying the file `language.dat` (say '`initexmf --find-tex language.dat`' to find out the absolute path).

As distributed with MiKTEX, `language.dat` has the following contents:

```
% File    : language.dat
% Purpose : specify which hypenation patterns to load
%           while running iniTeX
english ushyphen.tex
%ukenglish ukhyphen.tex
german ghyph31.tex
%italian ithyph.tex
%dutch nehyph2.tex
%finnish fihyph.tex
%norwegian nohyph.tex
%french f8hyph.tex
```

Lines starting with `%` are comments. The only uncommented lines in the example are `english ushyphen.tex` and `german ghyph31.tex`. That is, only hyphenation patterns for U.S. English and German will be loaded by TEX. To load other hyphenation patterns, you have to uncomment the corresponding lines.

After modifying `language.dat`, you have to create new LaTEX dump files (see).

7.5 Personal Configuration Files

You can cause MiKTEX to read a personal configuration file (in addition to the global one) by using the command line switch `--personal`:

```
initexmf --personal[=FILENAME]
```

If specified, *FILENAME* must be the name of an existing configuration file. If *FILE-NAME* is ommitted, then MiKTEX will not use a personal configuration file.

Values read from *FILENAME* will override those values that were read from the global configuration file.

For example, consider the case that you have some private LaTeX style files in you home directory (say `c:\users\me`). You could write a private configuration file (say `miktex.ini`) and place it in your home directory. The configuration file should look like this:

```
[LaTeX]
Input Dirs=.;c:\users\me//;%R\tex\latex//;%R\tex\generic//
```

Then you had to announce the configuration file this way:

```
initexmf --personal=c:\users\me\miktex.ini
```

8 Configuring Dvips

As distributed with MiKTEX, Dvips is configured as follows:

- When generating fonts, Dvips uses METAFONT mode `ljfour` (HP Laserjet 4).
- Horizontal resolution is 600 dpi.
- Paper size is A4.
- Dvips does not make use of the CM & AMS PostScript fonts.

You probably have to change some of theses settings for your site. To do so, open the Dvips configuration file `config.ps` with your favourite text editor.

The line starting with `M` specifies the METAFONT mode which Dvips uses for the generation of new raster fonts. Enter a suitable mode here. If you don't know the mode for your output device, then take a look at `metafont\misc\modes.mf`. This file contains an annotated list of METAFONT modes.

The line starting with `D` specifies the resolution. Enter a value that matches your printer.

See the Dvips manual, for more information about configuring Dvips.

Appendix A Manual Pages

A.1 Common Compiler Options

The following command-line switches are commonly supported by the TeX compilers, i.e. by TeX, Omega, pdfTeX and e-TeX:

--buf-size=n

Set the internal `buf_size` to n. `buf_size` is the maximum number of characters simultaneously present in current lines of open files and in control sequences between `\csname` and `\endcsname`; must not exceed 1073741823.

--c-style-errors

Show C/C++ style error messages. This switch implies `\scrollmode`.

--error-line=n

Set the internal `error_line` to n. `error_line` us the width of context lines on terminal error messages.

--half-error-line=n

Set the internal `half_error_line` to n. `half_error_line` is the width of first lines of contexts in terminal error messages; should be between 30 and (`error_line` - 15).

--halt-on-error

Quit after the first error.

--initialize

Initialize internal tables; these tables can be `\dump`ed to a dump file.

--job-name=$name$

Specify the name of the job. This also sets the name of all output files.

--job-time=$filename$

Set the time of all output files to the time of *filename*.

--help   Show a help screen and exit.

--max-in-open=n

Set the internal `max_in_open` to n. `max_in_open` is the maximum number of input files and error insertions that can be going on simultaneously.

--max-print-line=n

Set the internal `max-print-line` to n. `max-print-line` is the width of longest text lines output; should be at least 60.

--max-strings=n

Set the internal `max_strings` to n. `max_strings` is the maximum number of strings; must not exceed 1073741823.

--mem-bot=*n*

        Set the internal `mem_bot` to *n*. `mem_bot` is the smallest index in the `code` array dumped by `INITEX` (`INIOMEGA`, `INIPDFTEX`); must not be less than `mem_min`.

--mem-max=*n*

        Set the internal `mem_max` to *n*. `mem_max` is the greatest index in the internal `mem` array; must be strictly less than 1073741823.

--mem-min=*n*

        Set the internal `mem_min` to *n*. `mem_min` is the smallest index in the internal `mem` array; must be 0 or more; must be equal to `mem_bot` in `INITEX` (`INIOMEGA`, `INIPDFTEX`), otherwise `<=` `mem_bot`.

--mem-top=*n*

        Set the internal `mem_top` to *n*. `mem_top` is the largest index in the `mem` array dumped by `INITEX` (`INIOMEGA`, `INIPDFTEX`); must be substantially larger than 0 and not greater than `mem_max`.

--nest-size=*n*

        Set the internal `nest_size` to *n*. `nest_size` is the maximum number of semantic levels simultaneously active.

--param-size=*n*

        Set the internal `param_size` to *n*. `param_size` is the maximum number of simultaneous macro parameters.

--pool-size=*n*

        Set the internal `pool-size` to *n*. `pool_size` is the maximum number of characters in strings, including all error messages and help texts, and the names of all fonts and control sequences; must exceed `string_vacancies` by the total length of the program's own strings, which is currently about 30000.

--save-size=*n*

        Set the internal `save_size` to *n*. `save_size` is the amount of space for saving values outside of current group; must be at most 1073741823.

--src-specials

        Insert source file information into the DVI file.

--stack-size=*n*

        Set the internal `stack_size` to *n*. `stack_size` is the maximum number of simultaneous input sources.

--string-vacancies=*n*

        Set the internal `string_vacancies` to *n*. `string_vacancies` is the minimum number of characters that should be available for the user's control sequences and font names, after the program's own error messages are stored.

--tcx=*name*

        Causes TeX to process the TCX table *name*.

--terminal=oem

        Causes TeX to uses the current DOS codepage (e.g. cp850) for console output.

`--trace=`*traceflags*
>   Set trace flags.

`--trie-size=`*n*
>   Set the internal `trie_size` to *n*. `trie_size` is the amount of space for hyphen-
>   ation patterns; should be larger for `INITEX` (`INIOMEGA`, `INIPDFTEX`) than it is
>   in production versions of the program.

`--trie-op-size=`*n*
>   Set the internal `trie_op_size` to *n*. `trie_op_size` is the amount of space for
>   "opcodes" in the hyphenation patterns.

`--try-gz`   Try *file*`.tex.gz` if *file*`.tex` cannot be found.

`--undump=`*name*
>   Causes TeX to read the dump file *name*.

`--version`
>   Print version information and exit.

A.2 `initexmf`

`initexmf` is the MiKTeX Configuration Utility.

Command-Line Switches

`--dump`   Refresh all dump files (`*.base;*.efmt;*.fmt;*.mem`).

`--dump=`*program*
>   Refresh the dump files related to a specific program. *program* must be one of:
>   `elatex`, `etex`, `lambda`, `latex`, `metafont`, `metapost`, `omega`, `pdflatex`, `pdftex`,
>   `tex`.

`--find-elatex-input` *FILE*
>   Find e-TeX input file.

`--find-etex-input` *FILE*
>   Find e-LaTeX input file.

`--find-lambda-input` *FILE*
>   Find Lambda input file.

`--find-latex-input` *FILE*
>   Find LaTeX input file.

`--find-metafont-input` *FILE*
>   Find METAFONT input file.

`--find-metapost-input` *FILE*
>   Find MetaPost input file.

`--find-omega-input` *FILE*
>   Find Omega input file.

`--find-pdflatex-input` *FILE*
          Find pdfLaTeX input file.

`--find-pdftex-input` *FILE*
          Find pdfTeX input file.

`--find-tex-input` *FILE*
          Find TeX input file.

`--list-modes`
          List all known METAFONT modes.

`--local-root` *root*
          Specify the local TEXMF root.

`--mkpsres`
          Update the PostScript resource database 'psres.dpr'. You can use this option
          in conjunction with `--search` (see below).

`--mkpsres='dir'`
          Add a new font directory to the PostScript resource database 'psres.dpr'.

`--personal`
`-p`   Do not use a personal configuration file.

`--personal=`*FILENAME*
`-p`*FILENAME*
          Define the location of the personal configuration file.

`--print-only`
`-n`   Print what would be done. Nothing is changed.

`--reconfigure`
          Reconfigure MiKTeX.

`--report`   Create a configuration report.

`--root-directories` *dirlist*
`-r` *dirlist*   Specify the list of TEXMF root directories.

`--search`   Search for PS resource files (requires `--mkpsres`).

`--update-fndb`
`-u`   Refresh the whole filename database.

`--update-fndb=`*root*
`-u`*root*   Refresh the filename database for a specific TEXMF root.

`--verbose`
`-v`   Print information on what is being done.

`--version`
`-V`   Print the version number and exit.

A.3 `mp`

`mp` is the MetaPost compiler. The general command-line syntax is

> `mp` [*switches...*] [*filename*]

Command-Line Switches

`--c-style-errors`
> Show C/C++ style error messages. This switch implies `\scrollmode`.

`--initialize`
> Initializes MetaPost's internal tables so that they can be dumped.

`--help`   Shows a short help screen and exits.

`--tex=`*texprogram*
> Uses *texprogram* instead of `tex` when compiling text labels. This flag overrides the environment variable `TEX`.

`--version`
> Prints version information and exits.

Aliases

`inimp`   Equivalent to `mp --ini`.

Environment Variables

`TEX`   Specifies the TeX compiler which should be used when compiling text labels.

A.4 `omega`

Omega is an 16-bit enhanced version of TeX.

The general command-line syntax is

> `omega` [*switches*] [*firstinputline*]

firstinputline, if supplied, specifies the first input line. This is usually the name of an input file.

For example, the command

> `omega hello.tex`

causes Omega to produce the DVI file `hello.dvi` from the input file `hello.tex`. You can specify the input file without the `.tex` extension:

> `omega hello`

You must specify the `.tex` extension if the filename contains more than one dot (`.`). For example, it does not work to say

```
    omega foo.bar
```

You have to say

```
    omega foo.bar.tex.
```

instead.

Please note: you cannot specify file names that contain space characters, even if the file system allows such names.

Command-Line Switches

Omega supports the common compiler options (see Section A.1 [Common Compiler Options], page 21).

Aliases

`iniomega`   Equivalent to `omega --ini`.

`viromega`   Equivalent to `omega`.

`lambda`   Equivalent to `omega "&lambda"`.

A.5 `pdftex`

pdfTeX is a special version of TeX that outputs PDF.

The usual way to start pdfTeX is as follows:

```
    pdftex [options] [firstinputline]
```

firstinputline, if supplied, specifies the first input line. This is usually the name of an input file.

For example, the command

```
    pdftex hello.tex
```

causes pdfTeX to produce the PDF file `hello.pdf` from the input file `hello.tex`. You can specify the input file without the `.tex` extension:

```
    pdftex hello
```

You must specify the `.tex` extension if the filename contains more than one dot (`.`). For example, it does not work to say

```
    pdftex foo.bar
```

You have to say

```
    pdftex foo.bar.tex.
```

instead.

Please note: you cannot specify file names that contain space characters, even if the file system allows such names.

Command-Line Switches

Besides the common switches (see Section A.1 [Common Compiler Options], page 21), pdfTEX supports these command-line switches:

`--font-max=`*n*

> Sets the internal `font_max` to *n*. `font_max` is the maximum internal font number; must not exceed 5000.

Aliases

`inipdftex`

> Equivalent to `pdftex --ini`.

`virpdftex`

> Equivalent to `pdftex`.

`pdflatex`   Equivalent to `pdftex "&pdflatex"`.

A.6 `tex`

The usual way to start TEX is as follows:

> `tex` *options firstinputline*

firstinputline, if supplied, specifies the first input line. This is usually the name of an input file.

For example, the command

> `tex hello.tex`

causes TEX to produce the DVI file `hello.dvi` from the input file `hello.tex`. You can specify the input file without the `.tex` extension:

> `tex hello`

You must specify the `.tex` extension if the filename contains more than one dot (`.`). For example, it does not work to say

> `tex foo.bar`

You have to say

> `tex foo.bar.tex.`

instead.

Please note: you cannot specify file names that contain space characters, even if the file system allows such names.

Command-Line Switches

Besides the common switches (see Section A.1 [Common Compiler Options], page 21), TEX supports the following command-line switches:

`--font-max=`*n*

> Sets the internal `font_max` to *n*. `font_max` is the maximum internal font number; must not exceed 5000.

Aliases

`latex`   Equivalent to `tex "&latex"`.

`initex`   Equivalent to `tex --ini`.

Appendix B Index

T

U

V

W

Table of Contents