MiKTEX Manual

Revision 1.20d (MiKTEX 1.20d)
August 1999

Christian Schenk <cs@miktex.de>

1 What is MiKTeX?

MiKTeX is a free TeX distribution for Windows.

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 colored text
  - Supports PostScript fonts
  - Supports TrueType fonts
  - 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
  - Setup Wizard can be run unattended.

Components

The MiKTeX distribution consists of the following applications:

TeX 3.14159
: The classic TeX compiler.

e-TeX 2.1
: A feature-extended version of TeX.

Yap 0.96
: 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 5.86 Converts DVI files into PostScript.

MakeIndex 2.12
: Composes indexes.

BibTeX 0.99c
: Composes bibliographies.

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

TeXinfo, TtH, PSutils, . . .
: Lots of utilities.

1.1 How to get MiKTeX

MiKTeX Distribution

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

`systems/win32/miktex/`

Pretest versions can be downloaded from the MiKTeX Project Page (`http://miktex.de`).

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 graphics.

Adobe Acrobat Reader (`http://www.adobe.com/prodindex/acrobat/readstep.html`).
: A PDF viewer.

WinEdt (`http://home.istar.ca/~winedt`)
: WinEdt is a shareware TeX editor/shell. It cooperates with MiKTeX with respect to forward and inverse DVI search (see Section 5.1 [Source Specials], page 20).

[1] CTAN: Comprehensive TeX Archive Network

ActivePerl (`http://www.activestate.com`)
: ActivePerl is an implementation of Perl for the Windows platform. A few MiKTEX utilities (e.g. `psmerge`) are Perl scripts. You should install Perl if you want to use these utilities.

1.2 The MiKTEX Project Page

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

1.3 The MiKTEX Mailing List

MiKTEX Mailing List

Claus Ekstroem from Denmark has created a discussion list for MiKTEX. 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 MiKTEX Manual (which you are reading right now) concentrates on documenting MiKTEX specific features.

Other MiKTEX related documentation includes:

Frequently Asked Questions
: Lists answers to frequently asked questions.

Tips & Tricks
: Lists useful tips.

Shortcuts to these documents can be found in the Start Menu (see Section 3.2 [Start Menu], page 6).

2 What's new in MiKTeX 1.20d?

Updated Packages

- Dvips 5.86 (now includes a Windows help file)
- Texinfo 3.12p (now includes the `texi2dvi` utility)

New: `texify`

`texify` is a new command-line utility that simplifies the production 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.

`texify` imitates the shell script `texi2dvi` (written by Noah Friedman), which is a part of the GNU Texinfo distribution.

New: `hbf2gf`

`hbf2gf` (by Werner Lemberg) is intended to convert Hanzi Bitmap Fonts (HBF) into TeX generic font files (`GF` files). MakePK has been updated to make use of `hbf2gf`.

Resolved Problems

General

- MiKTeX doesn't search the local TEXMF tree, when it is a sub-folder of the installation TEXMF tree.

Setup Wizard

- The installation aborts if you don't have permission to replace system files (e.g. if you aren't Administrator).
- The path to the installation folder may not contain spaces.

3 Installing MiKTEX

3.1 Running the Setup Wizard

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 may have to add something like

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

to your `autoexec.bat`.

3.1.1 Setup Options

Some setup options can be specified on the command-line and/or in a separate text file named `setupwiz.opt`.

You can invoke the MiKTEX Setup Wizard with the following command-line options:

`--additional-directory-trees` *DIRS*
: Specify additional TEXMF directories.

`--allow-unattended-reboot`
: Allow a reboot in unattended mode.

`--dry-run`
: Simulate the installation process. No files will be installed. The log file will be written to the temporary directory.

`--installation-directory` *DIR*
: Specify the installation directory.

`--help`
: Show available options and exit.

`--no-additional-directory-trees`
: Prevent MiKTEX from using additional directory trees.

`--no-local-directory`
: Prevent MiKTEX from using a local directory.

`--program-folder` *FOLDER*
: Specify the MiKTEX program folder.

`--unattended`
: Run Setup Wizard in unattended mode. No user input is required.

Command-line options can also be specified in a text file named `setupwiz.opt`. This file must be in the same directory as `setupwiz.exe`.

3.2 Items in the Start menu

The Setup Wizard creates the following Start Menu items:

Shortcuts to documentation files

`MiKTeX | Help | Frequently Asked Questions`
: Opens a Windows Help file which contains answers to frequently asked questions.

`MiKTeX | Help | LaTeX2e Reference`
: Opens a Windows Help file which contains descriptions for many LaTEX commands.

`MiKTeX | Help | Local Guide`
: Opens a Windows Help file which contains the MiKTEX Manual.

`MiKTeX | Help | Release Notes`
: Last-minute notes.

`MiKTeX | Help | Tips and Tricks`
: Opens a Windows Help file which contains a list of useful tips.

Shortcuts to the configuration utility (`initexmf.exe`)

`MiKTeX | Maintenance | Create All Format Files`
: Creates all format files, i.e. runs '`initexmf --dump`'.

`MiKTeX | Maintenance | Create LaTeX Format File`
: Creates the LaTeX format, i.e. runs '`initexmf --dump=latex`'.

`MiKTeX | Maintenance | Reconfigure`
: Creates all format files and refreshes the filename database.

`MiKTeX | Maintenance | Refresh Filename Database`
: Refreshes the filename database.

Shortcuts to GUI-based MiKTeX applications

`MiKTeX | Yap`
: A shortcut to the DVI viewer.

3.3 The TEXMF Directory Hierarchy

The setup program creates a TDS-compliant directory structure. This structure is normally 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.

These directories need not to be on the same drive.

It's possible to change the locations of these directories anytime. See Section 4.1.1 [Defining TEXMF Root Directories], page 9, for more information.

3.3.1 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`.

3.3.2 Local Directory

The Local Directory (usually `c:\localtexmf`) 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.

3.4 Removing MiKTEX

There is no uninstall option (yet), thus you have to do it by hand if you decide to remove MiKTEX:

1. Remove the main TEXMF folder (usually `c:\texmf`) including sub folders.
2. Remove the local TEXMF folder (usually `c:\localtexmf`) including sub folders.
3. Remove 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`.

4 Configuring MiKTEX

4.1 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 and location of a personal configuration file

4.1.1 Defining TEXMF Root Directories

The standard setup procedure creates two TEXMF root directories:

- `c:\texmf`: the installation directory (see Section 3.3.1 [Installation Directory], page 7).
- `c:\localtexmf`: the local directory (see Section 3.3.2 [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 4.1.2 [Maintaining the Filename Database], page 9).

4.1.2 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 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`.

4.1.3 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, i.e. 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
```

4.1.3.1 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
```

4.1.4 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`.

4.1.4.1 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 hyphenation 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 Section 4.1.4 [Standard Dump Files], page 10).

4.1.5 Personal Configuration File

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 *FILENAME* is omitted, 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
```

4.2 The MiKTEX Configuration File

This section discusses the usage of MiKTEX Configuration File.

4.2.1 The Two Kinds of Configuation Files

MiKTEX configuration parameters are stored in two Configuration File:

- 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 optional *Personal Configuration File* contains per-user configuration settings. The location of the personal configuration file can be defined with the help of `initexmf` (see Section 4.1.5 [Personal Configuration File], page 11).

Personal configuration settings override global settings.

4.2.2 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 sequences have a special meaning:

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

`//` A flag, which causes 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.

4.2.3 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.

4.2.3.1 [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.

4.2.3.2 [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.

4.2.3.3 [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
```

4.2.3.4 [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`.

4.2.3.5 [MakeIndex]: MakeIndex Configuration Settings

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

`INDEXSTYLE`
: Search path for MakeIndex style files.

4.2.3.6 [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 specifiation 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.

4.2.3.7 [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.

4.2.3.8 [METAFONT]: METAFONT Configuration Settings

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

`Input Dirs`
: Search path for METAFONT input files.

4.2.3.9 [MetaPost]: MetaPost Related Configuration Settings

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

`Input Dirs`
: Where MetaPost searches for input files.

4.2.3.10 [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.

  `filesearch` Traces the find-file machinery.

  `access` Traces file accesses.

  `process` Traces secondary processes.

  `tcx` Traces TCX tables.

  `error` Traces error conditions.

  `time` Traces execution time.

`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`).

4.2.3.11 [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.

4.2.3.12 [otp2ocp]: `otp2ocp` Configuration Settings

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

4.2.3.13 [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.

4.2.3.14 [ps2pk]: `ps2pk` Configuration Settings

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

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

4.2.3.15 [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.

4.2.3.16 [Yap]: Yap Configuration Settings

`Input Dirs`

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

4.3 The Dvips Configuration File

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.

5 Non-standard TEX Features

This chapter describes the features that were added to the MiKTEX implementation of Donald Knuth's TEX.

5.1 Auto-insertion of Source Specials

What are source specials?

Source specials are pieces of information embedded in a DVI file, which make a connection between the source file location (e.g. line 100 in `foo.tex`) and the DVI location (e.g. page 2 in `foo.dvi`). Source specials can improve the Edit-TEX-View-Edit cycle:

1. You edit your source file.
2. You compile the source file to get a DVI file.
3. You execute a special editor command to open Yap, going directly to the DVI page that corresponds to the cursor location in your source file.
4. You navigate inside the DVI file (e.g. PgUp/PgDn).
5. You double-click somewhere on the DVI view; this causes Yap to bring the editor window to the front, moving the text cursor directly to the line that corresponds to the view location.

How to insert source specials

The TEX compiler option '`--src`' inserts source specials into the DVI file. You would say

```
latex --src foo.tex
```

to create the DVI file `foo.dvi` with embedded source specials.

5.2 TCX files: Character translations

[This section is "borrowed" from the Web2C manual].

TCX (TEX character translation) files help TEX support direct input of 8-bit international characters if fonts containing those characters are being used. Specifically, they map an input (keyboard) character code to the internal TEX character code (a superset of ASCII).

Of the various proposals for handling more than one input encoding, TCX files were chosen because they follow Knuth's original ideas for the use of the '`xchr`' and '`xord`' tables. He ventured that these would be changed in the WEB source in order to adjust the actual version to a given environment. It turned out, however, that recompiling the WEB sources is not as simple task as Knuth predicted; therefore, TCX files, providing the possibility of changing of the conversion tables on on-the-fly, has been implemented instead.

This approach limits the portability of TEX documents, as some implementations do not support it (or use a different method for input-internal reencoding). It may also be problematic to determine the encoding to use for a TEX document of unknown provenance; in the worst case, failure to do so correctly may result in subtle errors in the typeset output.

While TCX files can be used with any format, using them breaks the LaTEX '`inputenc`' package. This is why you should either use *tcxfile* or '`inputenc`' in LaTEX files, but never both.

Specifying TCX files:

- You can specify a TCX file to be used for a particular TEX run by specifying the command-line option '`-translate-file=`*tcxfile*' or (preferably) specifying it explicitly in the first line of the main document '`%& -translate-file=`*tcxfile*'.
- TCX files are searched for along the `TCXPath` path.
- `INITEX` ignores TCX files.

The MiKTEX distribution comes with at least two TCX files, '`il1-t1.tcx`' and '`il2-t1.tcx`'. These support ISO Latin 1 and ISO Latin 2, respectively, with Cork-encoded fonts (a.k.a. the T1 encoding). TCX files for Czech, Polish, and Slovak are also provided.

Syntax of TCX files:

1. Line-oriented. Blank lines are ignored.
2. Whitespace is ignored except as a separator.
3. Comments start with '`%`' and continue to the end of the line.
4. Otherwise, a line consists of one or two character codes:

   src [*dest*]
5. Each character code may be specified in octal with a leading '`0`', hexadecimal with a leading '`0x`', or decimal otherwise. Values must be between 0 and 255, inclusive (decimal).
6. If the *dest* code is not specified, it is taken to be the same as *src*.
7. If the same *src* code is specified more than once, it is the last definition that counts.

Finally, here's what happens: when TEX sees an input character with code *src*, it 1) changes *src* to *dest*; and 2) makes code the *dest* "printable", i.e., printed as-is in diagnostics and the log file instead of in '`^^`' notation.

By default, no characters are translated, and character codes between 32 and 126 inclusive (decimal) are printable. It is not possible to make these (or any) characters unprintable.

Specifying translations for the printable ASCII characters (codes 32–127) will yield unpredictable results. Additionally you shouldn't make the following characters printable: `^^I` (TAB), `^^J` (line feed), `^^M` (carriage return), and `^^?` (delete), since TEX uses them in various ways.

Thus, the idea is to specify the input (keyboard) character code for *src*, and the output (font) character code for *dest*.

6 TEXify: The MiKTEX Compiler Driver

`texify` is a command-line utility that simplifies the production 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 (where '`c:>` ' is your shell prompt):

```
c:> texify foo.tex
```

As shown in this example, the input filenames to `texify` must include any extension ('`.tex`', '`.ltx`', etc.).

Appendix A Manual Pages

A.1 Common Compiler Options

The following command-line switches are commonly supported by all TEX compilers.

`--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*
`--translate-file=`*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 hyphenation 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 bibtex

BibTEX is a preprocessor for the LaTEX document-preparation system. It handles most of the formatting decisions required to produce a reference list, outputting a `.bbl` file; with this file LaTEX actually produces the reference list.

Synopsis

`bibtex [`*option*`...] `*name*

Reads the file '*name*`.aux`' and outputs the file '*name*`.bbl`'.

Options

`--help`
: Shows a help screen and exits successfully.

`--min-crossrefs=`*N*
: Sets the internal `min_crossrefs` parameter to *N*.

`--version`
: Shows version information and exits successfully.

Documentation

See *BibTEXing*, available as file '`btxdoc.dvi`'.

A.3 bibtex8

BibTEX8 is an enhanced version of BibTEX. Enhanced by conversion to "big" (32-bit) capacity, addition of run-time selectable capacity and 8-bit support extensions. National character set and sorting order are controlled by an external configuration file.

Synopsis

`bibtex8 [`*option*`...] `*name*

where *name* is the name of the TEX auxilliary output file to be processed by `bibtex8`. The trailing `.aux` may be omitted.

Options

`-?`

`--help` Display some brief help text and then exit.

`-7`
`--traditional`
Operate in the original 7-bit mode. A CS file is not read: only 7-bit ASCII characters are supported and sorting is strictly by ASCII code value.

`bibtex8` will not allow you to specify `--traditional` with either the `--8bit` or `--csfile` option.

`-8`

`--8bit` Force 8-bit mode. A CS file is not read. All 8-bit characters (code > 127) are treated as letters and sorting is strictly by code page value.

BibTeX will not allow you to specify `--8bit` with either the `--csfile` or `--traditional` option.

`-c` *file*
`--csfile` *file*
Read *file* as the BibTEX8 codepage and sort definition (CS) file. The CS file is used to define the 8-bit character set used by BibTEX8 and the order in which those characters should be sorted.

BibTEX8 will not allow you to specify `--csfile` with either the `--8bit` or `--traditional` option.

`-d` *type*
`--debug` *type*
Report debugging information to the BibTEX8 log file and the standard error device. The value *type* controls the type of debugging information reported. *type* can be one or more of:

`all`	all debugging categories
`csf`	CS file processing
`io`	file I/O
`mem`	memory allocation and capacity
`misc`	other debugging information
`search`	path searching and file location

`-s`
`--statistics`
Report internal statistics to the BibTEX8 log file.

`-t`
`--trace` Report execution tracing to the BibTEX8 log file.

`-v`
`--version`
Report BibTEX8's version and then exit.

`-B`

`--big` Set BibTEX8's capacity to "big". The size of particular parameters will be set as follows (the default sizes are shown in brackets):

Hash_Prime	8,501	(4,253)
Hash_Size	10,000	(5,000)
Max_Cites	2,000	(750)
Max_Ent_Ints	4,000	(3,000)
Max_Ent_Strs	6,000	(3,000)
Max_Fields	30,000	(17,250)
Max_Strings	10,000	(4,000)
Pool_Size	130,000	(65,530)
Wiz_Fn_Space	6,000	(3,000)

`-H`
`--huge` Set BibTEX8's capacity to "huge". The size of particular parameters will be set as follows (the default sizes are shown in brackets):

Hash_Prime	16,319	(4,253)
Hash_Size	19,000	(5,000)
Max_Cites	5,000	(750)
Max_Ent_Ints	5,000	(3,000)
Max_Ent_Strs	10,000	(3,000)
Max_Fields	85,000	(17,250)
Max_Strings	19,000	(4,000)
Pool_Size	500,000	(65,530)
Wiz_Fn_Space	10,000	(3,000)

`-W`
`--wolfgang`
Set BibTEX8's capacity to "really huge" - required for Wolfgang's PhD thesis. The size of particular parameters will be set as follows (the default sizes are shown in brackets):

Hash_Prime	30,011	(4,253)
Hash_Size	35,000	(5,000)
Max_Cites	7,500	(750)
Max_Ent_Ints	7,500	(3,000)
Max_Ent_Strs	10,000	(3,000)
Max_Fields	125,000	(17,250)
Max_Strings	30,000	(4,000)
Pool_Size	750,000	(65,530)
Wiz_Fn_Space	10,000	(3,000)

`-M` *n*
`--min_crossrefs` *n*
: Set `min_crossrefs` to *n*. If an item is cross-referenced at least *n* times, it will be placed in the list of citations, even if it is not explicitly `\cited` as a reference. The default value is 2.

`--mcites` *n*
: Allow a maximum of *n* distinct `\cite`s in the `.aux` files. This number must be less than the maximum number of strings (settable with `--mstrings`).

`--mentints` *n*
: Allow a maximum of *n* integer entries in the `.bib` databases.

`--mentstrs` *n*
: Allow a maximum of *n* string entries in the `.bib` databases.

`--mfields` *n*
: Allow a maximum of *n* fields in the `.bib` databases.

`--mpool` *n* Set the string pool to *n* bytes.

`--mstrings` *n*
: Allow a maximum of *n* unique strings. This number must be less than the hash size and greater than the maximum number of `\cite`s (settable with `--mcites`).

`--mwizfuns` *n*
: Allow a maximum of *n* wizard functions.

Documentation

For a general description of `bibtex8`, see the text file '`00readme.txt`'.

For a description of the CS file syntax see the text file '`csfile.txt`'.

A.4 `dvicopy`

`dvicopy` is a utility program that allows one to take a DVI file that references composite fonts (VF) and convert it into a DVI file that does not contain such references.

Synopsis

`dvicopy [`*option*`...] `*old* *new*

Converts DVI file *old* into *new*.

Options

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

`--mag=`*MAG*
: Sets magnification to *MAG*.

`--select=`*range*
: Selects a range of pages to be copied.

`--version`
: Prints version information and exits successfully.

A.5 `dvipdfm`

`dvipdfm` is a DVI to PDF conversion utility.

Synopsis

`dvipdfm [`*option*`...]` *dvifile*

Options

`-c` Ignore color specials (for printing on B&W printers).

`-f` *filename*
: Set font map file name [*t1fonts.map*].

`-o` *filename*
: Set output file name [*dvifile*.pdf].

`-l` Landscape mode.

`-m` *number*
: Set additional magnification.

`-p` *papersize*
: Set papersize (`letter`, `legal`, `ledger`, `tabloid`, `a4`, or `a3`) [`letter`].

`-r` *resolution*
: Set resolution (in DPI) for raster fonts [600].

`-s` *pages* Select page ranges (-).

`-x` *dimension*
: Set horizontal offset [1.0in].

`-y` *dimension*
: Set vertical offset [1.0in].

`-e` Disable partial font embedding [default is enabled].

`-z` *number*
: Set compression level (0-9) [default is 9].

`-v` Be verbose.

`-vv` Be more verbose.

All dimensions entered on the command line are "true" TEX dimensions. Argument of `-s` lists physical page ranges separated by commas, e.g., '`-s 1-3,5-6`'.

Documentation

See *Dvipdfm User's Manual* (available as file '`dvipdfm.dvi`', for a complete description.

A.6 `dvips`

`dvips` is a program to translate a DVI file into PostScript.

Synopsis

`dvips [`*options*`...] `*filename*`[.dvi]`

Options

See the Dvips documentation, for a list of available options.

Documentation

For a complete description of Dvips, see *Dvips: A DVI driver*. This document is available

- as a Windows help file (`dvips.hlp`
- as a DVI file (`dvips.dvi`)

A.7 `initexmf` (MiKTEX Configuration Utility)

`initexmf` is the MiKTEX Configuration Utility.

Synopsis

`initexmf [`*option*`...]`

Options

`--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-LaTEX input file.

`--find-etex-input` *FILE*
: Find e-TEX input file.

`--find-executable` *FILE*
: Find a MiKTEX executable.

`--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-other-executable` *FILE*
: Find an executable.

`--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.

`--quiet`
: Suppress screen output.

`--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.8 `mp` (MetaPost)

MetaPost (installed as `mp`) reads a series of pictures specified in the MetaPost programming language, and outputs corresponding PostScript code.

Synopsis

```
mp [option...] [name[.mp]] [command...]
mp [option...] "&format" [command...]
```

Options

`--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 successfully.

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

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

Aliases

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

`mpost` Equivalent to '`mp`'.

`virmp` Equivalent to '`mp`'.

Environment Variables

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

Documentation

For a complete description of the MetaPost language, see AT&T technical report CSTR-162, available as the file '`mpman.ps`'.

A.9 `omega`

Omega is a 16-bit enhanced version of TEX.

Synposis

`omega [`*option*`...] [`*name*`[.tex]] [`*command*`...]`

Options

Omega supports the common compiler options (see Section A.1 [Common Compiler Options], page 23).

Aliases

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

`viromega` Equivalent to `omega`.

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

Documentation

For a complete description of Omega, see the Omega manual, available as the file '`omega-manual.dvi`'.

A.10 `pdftex`

pdfTEX is a special version of TEX that outputs PDF.

Synopsis

`pdftex [`*option*`...] [`*name*`[.tex]] [`*command*`...]`
`pdftex [`*option*`...] "&`*format*`" [`*command*`...]`

Options

Besides the common command-line switches (see Section A.1 [Common Compiler Options], page 23), pdfTEX supports these options:

`--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"`.

Documentation

For a complete description of pdfTEX, see the the pdfTEX User Manual, available as file '`pdftexman.pdf`'.

A.11 `tex`

`tex` is Donald Knuth's TEX compiler.

Synopsis

```
tex [option...] [name[.tex]] [command...]
tex [option...] "&format" [command...]
```

Options

Besides the common command-line switches (see Section A.1 [Common Compiler Options], page 23), `tex` supports the following options:

`--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`.

`virtex`
: Equivalent to `tex`.

See Also

See Section A.12 [texify], page 35, for an alternative way to invoke TEX.

Documentation

For a complete description of TEX, see *The TEXbook* by Donald E. Knuth.

A.12 `texify`

`texify` runs Texinfo or LaTEX input files through TEX (pdfTEX) in turn until all cross-references are resolved, building all indices.

Synopsis

`texify [`*option*`]...` *file*`...`

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 TEX when needed.

Options

`-@` Use `@input` instead of `\input`; for preloaded Texinfo.

`-b`
`--batch` No interaction.

`-c`
`--clean` Remove all auxiliary files.

`-e`
`--expand` Force macro expansion using makeinfo.

`-I` *DIR* Search *DIR* for input files.

`-h`
`--help` Display this help and exit successfully.

`-l` *LANG*
`--language=`*LANG*
Specify the *LANG* of *FILE*: `latex` or `texinfo`.

`-p`
`--pdf` Use pdfTEX or pdfLaTEX for processing.

`-q`
`--quiet` No output unless errors (implies `--batch`).

`-s`
`--silent` Same as `--quiet`.

`--src-specials`
: Insert source specials into the DVI file.

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

`-v`
`--version`
: Display version information and exit successfully.

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.

Aliases

`texi2dvi` Equivalent to `texify`.

Appendix B Index

%

-

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

Table of Contents