Managing a network TeX installation under Windows

Keywords
MikTeX, TeXnicCenter, filename database, Windows, registry, roaming profiles, graphic file formats

This paper is about the MikTeX installation that I maintain for the Economics Department of the Rijksuniversiteit Groningen. We have long been the home of 4TeX. But when development of this project stopped, the time came that this TEX installation had to be replaced by something else. This something else was going to be MikTeX with TeXnicCenter as editor and front end.

There are various Windows editors which support MikTeX, *i.e.* editors which have menu items and buttons for compiling and viewing your TeX documents with MikTeX. Configuring *e.g.* TeXnicCenter or Winedt for MikTeX is almost automatic. TeXnicCenter is free, both in the beer and in the speech sense. The MikTeX site lists a few more free editors. LaTeXEditor (`http://www.ntu.edu.sg/home5/pg03053527/latexeditor/`) and Texmaker (`http://www.xm1math.net/texmaker/`) seem to have a similar focus as TeXnicCenter.

MikTeX itself comes with a configuration program, MikTeX Options or mo.exe[1], which has to be started from outside the editor.

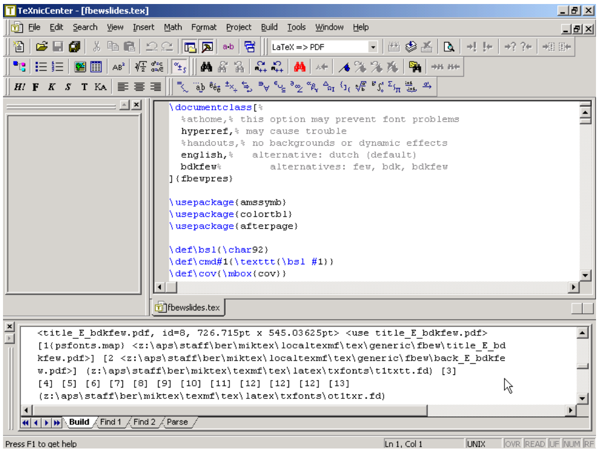
Over time, the MikTeX installation has been accumulating some add-ons, especially for handling graphics; see further on.

Moving from 4TeX to MikTeX

I didn't try to create a unified 4TeX-style interface for everything, and also didn't try to replicate the functionality of 4TeX, but I did collect the local macros, fonts and graphics from 4TeX which were still in use and put them into the MikTeX installation, sometimes with some minor tweaks.

I dropped support for the old LaTeX 2.09 since it would have meant real work for something that might not even get used.

The MikTeX installation was put online early in 2003. For a year and a half, MikTeX and 4TeX were available side by side, but in the end, after six months notice, I removed 4TeX from the network.


TeXnicCenter - [fbewslides.tex]
File Edit Search View Insert Math Format Project Build Tools Window Help
LaTeX => PDF
fbewslides.tex
Build
Find 1
Find 2
Parse
Press F1 to get help
Ln 1, Col 1
UNIX
OVR READ UF NUM RF

```
\documentclass[%
  %athome,% this option may prevent font problems
  hyperref,% may cause trouble
  %handouts,% no backgrounds or dynamic effects
  english,%   alternative: dutch (default)
  bdkfew%         alternatives: few, bdk, bdkfew
]{fbewpres}

\usepackage{amssymb}
\usepackage{colortbl}
\usepackage{afterpage}

\def\bsl{\char92}
\def\cmd#1{\texttt{\bsl #1}}
\def\cov{\mbox{cov}}
```

```
<title_E_bdkfew.pdf, id=8, 726.715pt x 545.03625pt> <use title_E_bdkfew.pdf>
[1{psfonts.map} <z:\aps\staff\ber\miktex\localtexmf\tex\generic\fbew\title_E_bd
kfew.pdf>] [2 <z:\aps\staff\ber\miktex\localtexmf\tex\generic\fbew\back_E_bdkfe
w.pdf>] (z:\aps\staff\ber\miktex\texmf\tex\latex\txfonts\t1txtt.fd) [3]
[4] [5] [6] [7] [8] [9] [10] [11] [12] [12] [12] [13]
(z:\aps\staff\ber\miktex\texmf\tex\latex\txfonts\ot1txr.fd)
```

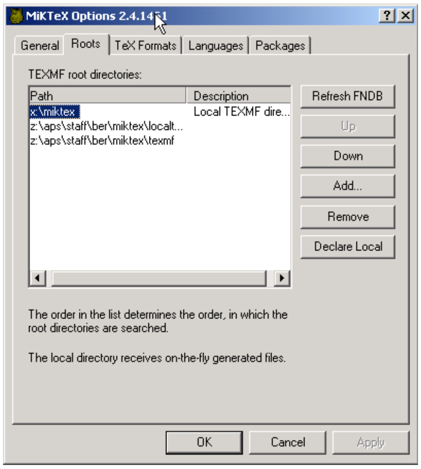
Figure 1. TeXnicCenter, a MikTeX frontend

Layout and contents of the installation

Texmf trees. As to the organization of macros, fonts and other support files, MikTeX is rather similar to other modern free TeX implementations: it organizes its files into *texmf trees*, which have a standardized structure: *e.g.* font-related files are in subdirectories fonts\tfm, fonts\type1 etc., and LaTeX macros are in tex\latex. Each tree follows such a structure and has its own filename database. Users can configure in which order the trees are going to be searched.

I configured the following three trees: a main tree for files coming from the MikTeX distribution; a department tree for local additions, including the files inherited from the 4TeX installation; and a user tree for people's own macros and other files. Users have write access only to their own user tree. With this setup, a MikTeX upgrade won't interfere with the department tree, and anything done to the network installation leaves user files alone.

Package selection. MikTeX is distributed as a setup program and a set of packages. The MikTeX Setup Wizard lets you choose between three initial package sets: small, large or everything, which you can modify later on. I picked the 'large' package set, and added and removed some packages afterwards.



TEXMF root directories:

Path	Description
x:\miktex	Local TEXMF dire...
z:\aps\staff\ber\miktex\localt...	
z:\aps\staff\ber\miktex\texmf	

The order in the list determines the order, in which the root directories are searched.

The local directory receives on-the-fly generated files.

Figure 2. Defining trees and their priorities with MikTeX Options

Add-ons. Non-MikTeX add-ons originally included Ghostscript and GSview, but for the current edition, this was no longer necessary, since Ghostscript and GSview were already installed separately.

For better scripting, I included the Perl .exe and .dll files, but placed them outside the search path. If people have a need for Perl then they can install their own copy, without these two files getting in the way.

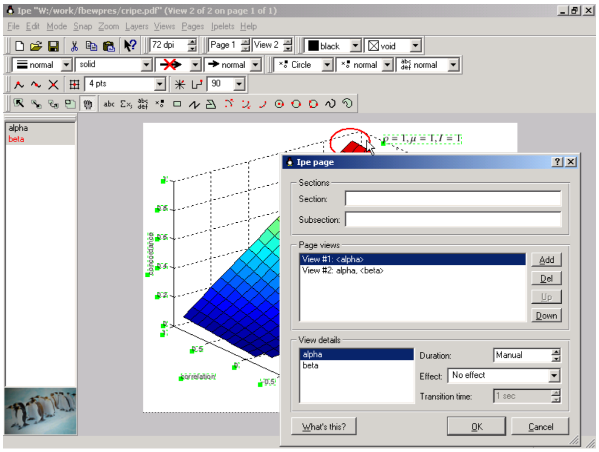
Graphics support

Draw programs. Our MikTeX installation includes a couple of draw programs. One of these is Ipe (`http://ipe.compgeom.org`), which has a few interesting features:

- □ It uses pdflatex in the background for typesetting text elements. You can tune typographic details with LaTeX preamble commands.
- □ It can import arbitrary pdf via a separate conversion utility.
- □ A drawing can be layered in the sense that it can be displayed incrementally in a pdf presentation. In fact, Ipe also advertises itself as a tool for making presentations.

A second draw program is LaTeXCAD, which generates LaTeX picture environments. It is very old and basic and is only included for people who still have old LaTeXCAD drawings in use.

Converters. I also added some PostScript-, .eps and .pdf conversion scripts, with desktop shortcuts which can be used as drag-and-drop targets. For conversion from pdf to EPS or Encapsulated PostScript I added the xpdf utilities.



Sections
Section:
Subsection:

Page views
View #1: <alpha>
View #2: alpha, <beta>

View details
alpha
beta
Duration: Manual
Effect: No effect
Transition time: 1 sec

Figure 3. The Ipe draw program has views or pages which are really assemblies of 'layers'.

I plan to write a basic GUI tool, custom tailored for our installation, which offers all available conversions from a single interface. Of course, the real work will be done by Ghostscript and other trusty command-line standbys.

There is also an installer for wmf2eps. This program offers a fairly practical way to make graphics from MS Office and other Windows programs available to LaTeX. It seems that not much has happened with it lately, but it still works well enough. It relies on a virtual printer driver, and therefore isn't a good candidate for a network install.

Its main advantage over simply printing to an eps file is that it calculates a tight boundingbox, rather than just turning an entire page into a graphic.

Recent versions of Ghostscript have a 'boundingbox output device'. There are now various scripts which use Ghostscript to fix boundingboxes. The as yet nonexistent GUI tool mentioned above should also offer such an option, as an alternative to using wmf2eps[2].

An installer

Installation of the network version involves:

- □ storing configuration information in the registry
- □ creating the user tree if it doesn't exist
- □ creating desktop and start menu shortcuts
- □ generating the filename database

Because our desktop systems are very standardized, none of this requires user input.

The filename database. MikTeX's texmf trees have an important and annoying difference with the more unixy varieties: the filename database of a tree is not stored with that tree, but in a designated 'local' tree which receives generated files. This local tree can only be the user tree. That means that it is up to the user to update his filename database if items are added to or moved around in the global installation.

The filename database can be generated from the MikTeX Options program but, fortunately for writers of installation scripts, it can also be done from the command line:

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

It happens often enough that the installation fails because the user becomes impatient with the generation of the filename database. Without the help of this database, MikTeX becomes very very slow. To minimize the problem, I saved filename database generation till last in the installation process and preceded it with dire warnings about not interrupting the installer. If these warnings are ignored, then the filename database can still be generated after the fact from the MikTeX Options program. An alternative would be to copy a pregenerated filename database to the right place during installation.

Dealing with the registry

Under Windows, almost any configuration information is stored in the registry.

The registry contains a set of hierarchically organized keys. There are several root keys. The most important ones are HKEY_CURRENT_USER and HKEY_LOCAL_MACHINE, HKCU and HKLM in short. The HKCU part of the registry may be on a network drive. HKLM is normally in a subdirectory of the Windows directory.

The actual information is contained in *values* under those keys. Values are name–data pairs.

For users, there are very few reasons to edit the registry directly. There are almost always specialized menu entries and dialogs available, such as Tools menus and Control Panel entries.

Registry tools. If you do need to view or manipulate the registry directly, Windows contains a number of tools: you can browse the registry with regedit[3]. You can export and import registry keys to and from .reg files with either regedit or the command-line tool reg.exe. These .reg files are editable text files. Reg.exe may not be installed by default, though. Type 'reg /?' for help.

Various programming languages, including Perl, VBScript and installer programs, also have functions for accessing the registry.

Finding out what registry values are needed. There are a number of techniques for identifying the registry settings that you need: one way would be to inspect the source code of the original installer.

A second method is browsing and searching the registry for likely strings, and testing afterwards whether you have captured enough to make the program work as intended. However, such testing can be time-consuming since you sometimes have to re-login or reboot before the changes take effect.

A third method is to export the registry to a text file before installation and after installation or first use, and compare the differences. There exist automated installers which do just this, but the GNU diff program works just fine.

You still have to decide what are the differences that matter. There will probably a lot of spurious differences. For example, most programs record windows positions and their most recently used files in the registry.

Also, some information which occurs only once can appear to occur multiple times. In particular, under Windows 2000 and later, the keys under HKCR are copies from keys under HKLM\software\classes and HKCU\software\classes.

All users or not: HKLM and KHCU. Often, when you install software, there is a choice whether or not to install for all users. If you do, keys are added under HKLM\software; otherwise under HKCU\software.

Where to look. The most important settings are under software\<program name> and under software\classes (either from HKLM or from HKCU). The keys under classes define file types and define what happens when you double-click a file in Windows Explorer. Command-line programs may not need any entries here.

Uninstall information can be found under HKLM\software\Microsoft\Windows\CurrentVersion\Uninstall. Our installer doesn't yet include an uninstaller.

Registry entries for MikTeX itself. MikTeX makes very modest use of the registry. It just records the locations of the texmf trees, stores uninstall information, and defines the .dvi filetype, associating it with the yap previewer.

I also added the MikTeX binaries directory to the search path, for those people who prefer to run MikTeX from the command line. On Windows 2000 and Windows XP the search path and other environment variables are stored in the registry; for the current user under HKCU\software\environment, for the local system under HKLM\system\currentcontrolset\control\session manager\environment.

Registry entries for Ghostscript and GSview. Ghostscript needs to record the location of the main dll and of its own fonts and support files. GSview defines the .eps and .ps file types and associates them with itself.

Registry entries for TeXnicCenter. TeXnicCenter stores a lot of information in the registry, but it can configure itself when it is started for the first time if it can find the MikTeX, Acrobat, Ghostscript and GSview registry settings. All the user has to do is to answer 'yes' a few times. I decided to leave configuration of TeXnicCenter to itself.

It is possible to rerun the TeXnicCenter configuration wizard at a later date. This may come in handy whenever MikTeX or Ghostscript or GSview has moved, or the Adobe Reader has been upgraded.

It would have been nice if TeXnicCenter checked at startup whether these programs are still at their previous location.

More installers

I started out with one installer, but now there are several.

A network installation for a LaTeX course. A second network installation was needed for a computer course for econometrics students. This installation is a slightly stripped-down version of the first one: no department tree, and without some of the add-ons.

A cd installation. Earlier, I had already made a cd containing the standard installers for MikTeX, Ghostscript, GSview and TeXnicCenter, and a copy of the department tree, plus a file with instructions how to put everything together.

There were two problems with this: it was complicated enough that some people preferred to let me install MikTeX for them, and other people figured that they might as well download and install MikTeX directly from the internet. Which was not exactly wrong, but differences in their setup sometimes made it difficult to debug their problems.

So I hope that the new installer cd has persuaded some people not to do a do-it-yourself install.

The differences. I already listed some of the differences between staff and student installations.

Some differences between network and cd installers are:

□ With the cd installer, users can choose locations for the main installation and for their own data. These locations are fixed in the network version.

□ The cd installer has to copy everything to the harddisk, whereas in the network version everything is already in place. In fact, re-running the network installer is no big deal. Which is just as well, since time and again configurations get messed up by a malfunctioning network or other mishaps.

□ The cd installer tests for Ghostscript and GSview. If they are missing, the user first has to install these programs, *e.g.* with the installers provided on the cd. The network version simply knows that Ghostscript and GSview are present and where they are.

□ The cd does an 'All users' install; the network version doesn't. Since on our network the `HKCU` part of the registry and the start menu are on the user's network drive, you can run MikTeX from any workstation on the network.

□ The cd only contains MikTeX fonts; not the additional department fonts. Adding fonts in MikTeX 2.4 is tricky at best. Adding them to systems that I didn't control caused too much grief.

It was not difficult to create the installer script as one main script with four different wrapper scripts.

I kept the installation and the installer on a Linux Samba server. I managed to put all 'real' files in a single directory tree, and to access these files through four different sets of symlinks. This prevented worries about keeping the versions in sync.

Installer programs. The standard way to distribute applications at the university is to create entries in NAL, or Novell Application Launcher, using Novell ZENworks. As I understand, ZEN identifies file system and registry differences before and after installation. With ZEN, an installer can make system changes which the user wouldn't have permissions for without ZEN. However, a first attempt to create such a NAL entry for MikTeX, done together with our NAL specialist, wasn't exactly plain sailing.

I needed something that I could develop and test on my own system. This was even more important for the student install for the LaTeX course, where I had to do everything through intermediaries who weren't even in the same building.

For the first edition, which didn't include a cd counterpart, I could make do with a batchfile with some embedded Perl code[4] for manipulating the search path.

The cd version of the second edition required user interaction, first for telling users to install Ghostscript if it wasn't found, second for asking users where MikTeX should be installed. So it was really time to switch to a GUI installer.

I picked NSIS (`http://nsis.sourceforge.net/`). It is completely scriptable and can be used from the command line[5]. It has functions for reading and writing the registry and for creating shortcuts. It offers string handling and conditionals. You can choose to what extent you want to package files into the installer itself, *i.e.* you can also tell the installer to copy files straight from the installation media to the target system.

The principal drawback of NSIS is very low-level string handling, which is quite painful if you are used to Perl string handling and regular expressions.

I have also heard good things about InnoSetup (`http://www.jrsoftware.org/isinfo.php`), but by then I was almost finished with my NSIS installer.

Development and testing

Virtual machines. Nowadays, you don't need physical test machines anymore; with software such as VMware you can create virtual guest machines for testing inside a host, *i.e.* inside your everyday computer. The hard disk of this guest computer is a very large file on the host's disk. Its screen can take up the entire physical screen, but it may also run inside a window, whatever is convenient. If host and guest have similar processors, performance can be quite decent.

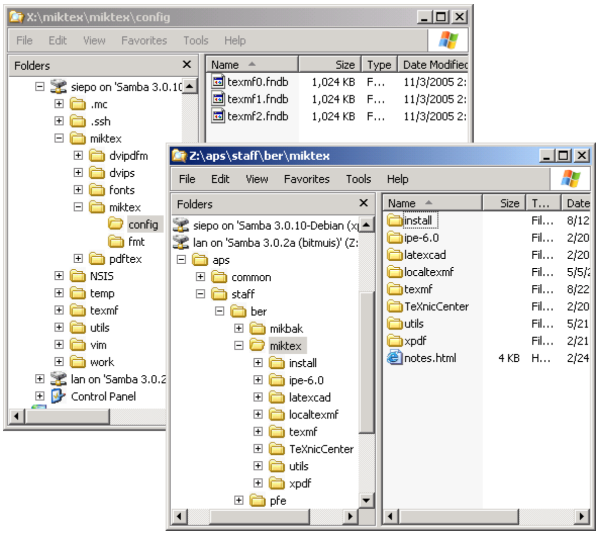
VMware has versions for Windows and for Linux hosts. There are other options for virtual machines, both commercial and free: the Xen project (`http://www.xensource.com`) is getting a lot of attention, and may in time become a very interesting alternative. See also QEMU (`www.qemu.org`), Win4Lin (`http://win4lin.com`) and Virtual PC (`http://www.microsoft.com`[6]. Some of these emulators are focused more on running Windows applications than on creating a realistic test environment.

For testing installers, you can create a 'clean' virtual machine with just Windows and some indispensable programs installed. Then you can run simulations on copies of this virtual machine. Getting another clean test machine is just a matter of making a fresh copy of the original machine, which takes only minutes.

Virtual networking. For networking, I let VMware create what it calls a host-only network, with no direct access to an external network. This saved me the hassle of protecting virtual Windows machines against malware from the internet. I configured the Linux host as a Samba server, with the MikTeX installation in a Samba share, and the user's home directory in another share.

Roaming profiles. The university started using *roaming profiles*. The idea is to place user configuration data as much as possible on their own network home drive. These user configuration data include *e.g.* their start menu and the HKCU part of the registry.

With Samba, roaming profiles means configuring the (or a) Samba server as a PDC or Primary Domain Controller. This is no fun. It means creating things called machine accounts for the client machines, and painstakingly reading Samba documentation. A very helpful and funny guide was 'The Unoffical Samba HOWTO'. You can find this document via the Samba site. Its current location is `http://hr.uoregon.edu/davidrl/docs/samba.html`.



X:\miktex\miktex\config

- siepo on 'Samba 3.0.1(
  - .mc
  - .ssh
  - miktex
    - dvipdfm
    - dvips
    - fonts
    - miktex
      - config
      - fmt
    - pdftex
  - NSIS
  - temp
  - texmf
  - utils
  - vim
  - work
- lan on 'Samba 3.0.2
- Control Panel

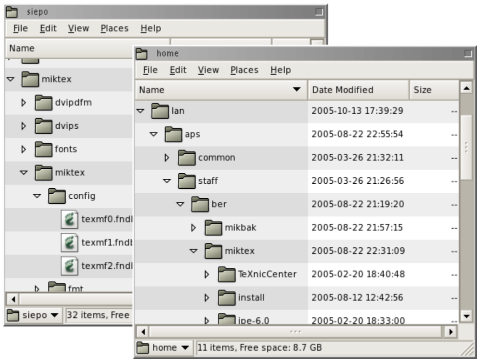
Name	Size	Type	Date Modifiec
texmf0.fndb	1,024 KB	F...	11/3/2005 2:
texmf1.fndb	1,024 KB	F...	11/3/2005 2:
texmf2.fndb	1,024 KB	F...	11/3/2005 2:

Z:\aps\staff\ber\miktex

- siepo on 'Samba 3.0.10-Debian (x
- lan on 'Samba 3.0.2a (bitmuis)' (Z:
  - aps
    - common
    - staff
      - ber
        - mikbak
        - miktex
          - install
          - ipe-6.0
          - latexcad
          - localtexmf
          - texmf
          - TeXnicCenter
          - utils
          - xpdf
        - pfe

Name	Size	T...	Date
install		Fil...	8/12
ipe-6.0		Fil...	2/20
latexcad		Fil...	2/20
localtexmf		Fil...	5/5/2
texmf		Fil...	8/22
TeXnicCenter		Fil...	2/20
utils		Fil...	5/21
xpdf		Fil...	2/21
notes.html	4 KB	H...	2/24

Figure 4. The Samba shares seen from the Windows client machine. It doesn't make a difference whether the Windows machine is physical or virtual.



siepo

Name
miktex
dvipdfm
dvips
fonts
miktex
config
texmf0.fndl
texmf1.fndl
texmf2.fndl
fmt

siepo 32 items, Free

home

Name	Date Modified	Size
lan	2005-10-13 17:39:29	--
aps	2005-08-22 22:55:54	--
common	2005-03-26 21:32:11	--
staff	2005-03-26 21:26:56	--
ber	2005-08-22 21:19:20	--
mikbak	2005-08-22 21:57:15	--
miktex	2005-08-22 22:31:09	--
TeXnicCenter	2005-02-20 18:40:48	--
install	2005-08-12 12:42:56	--
ipe-6.0	2005-02-20 18:33:00	--

home 11 items, Free space: 8.7 GB

Figure 5. The Samba shares seen from the Linux server. It makes no difference whether the server is a separate machine, a VMware host or a second VMware guest machine.

For testing, I now make clean MikTeX-free profiles, with just the worst default Windows settings fixed, and work with copies of those clean profiles, just as I already did with guest machines.

Disappearing file types. In theory, with MikTeX and TeXnicCenter, roaming profiles should work perfectly: there should be no need to install or configure anything on the machine itself. In practice, definitions of filetypes under HKCU, *i.e.* all keys and values under `HKCU\software\classes`, got lost in between logins – in real life, not in my test setup. A good workaround is to recreate those filetype definitions automatically at every login, *e.g.* via the Start, Programs, startup menu.

I made some mistakes here: I could have caught the problem beforehand if I had been more rigorous with using clean test machines, and my original workaround was a good deal clumsier than the fix with the startup menu.

Final thoughts

I would have liked to add some nice generalities, but all I can think of is that Windows is still evil and has been giving me a hard time.

Notes

1. MikTeX also has a package manager. But of course that is not useful to users who don't have write access to the installation.
2. There is also a Linux program called wmf2eps, but I have had mixed results with it. It seems better to convert wmf and emf graphics to .eps or .pdf on the original system *before* trying to use them elsewhere.
3. Under earlier versions of Windows, regedit and regedt32 each could do things that the other couldn't. Under Windows XP, regedit combines the functionality of the earlier regedit and regedt32. Its version of regedt32 simply starts up regedit.
4. That is, the batchfile calls Perl with the `-x` switch and itself as parameter; see the perlrun manpage.
5. For GUI addicts, there is an interface with some buttons to push. There are also third-party editors with a GUI for building dialog boxes.
6. Virtual PC was bought from Connectix by Microsoft in the second half of 2003. The Macintosh version of Virtual PC was at the time the only real option for running Windows on the Mac. I, along with many other Mac owners, was duly shocked by this sell-out. But in the meantime, other emulators appeared, and now that the Mac is moving to Intel, we can hope for VMware-quality virtual machines on Mac OS X from other companies than Microsoft.

Siep Kroonenberg
`siepo@cybercomm.nl`