

Installing MiKTeX and MusixTeX on Windows 2000

Eva Jaksch (ej@nexta.at)

File Name: musixwininstall.pdf Version: 1.3 Date: 23 November 2003

This document describes the process of installing MusixTeX and MiKTeX 2.1.8. The installation procedure has been tested on Windows 2000. No guarantees are given about its applicability to any other versions of either Windows or MikTeX.

Contents

1. MiKTeX 2.1	1
1.1. Where to get it	1
1.2. Downloading	1
1.3. Installing	1
1.4. Making a local binaries directory	2
2. MusixTeX and friends	3
2.1. Where to get them	3
2.2. MusixTeX	3
2.3. Additional packages	3
2.4. Installing	3
3. Type 1 Fonts for MusixTeX	4
3.1. Where to get them	4
3.2. Installing	4
4. Type M Postscript slurs	5
4.1. Where to get them	5
4.2. Installing	5
5. Upgrading an existing MusixTeX installation	6
6. Writing MusixTeX source files	6

1. MiKTeX 2.1

1.1. Where to get it

You can obtain MiKTeX on the WWW either at a CTAN site such as `ftp.dante.de`, `ftp.tex.ac.uk` or their mirror sites, or directly from `www.miktex.org`.

1.2. Downloading

1. From the download site of your choice, download `setup.exe`.
2. Run `setup.exe` and click "Download".
3. Click "Small", "Large" or "Total" to select the size of your MiKTeX installation. The "Small" installation is sufficient for using MusixTeX. The "Total" installation is **STRONGLY DISCOURAGED** (see below).
4. Follow the prompts to finish downloading the install files.

1.3. Installing

1. Run `setup.exe` again and choose "Install".
2. Select the size of your installation (recommended: "Small") and enter the path to the directory where you saved the install files during download.

Note: In MiKTeX 2.1.8, the "Total" option will automatically install a copy of MusixTeX. Allowing this to happen is strongly discouraged for three reasons:

- This version of MusixTeX won't work as desired, since it does not contain `musixflx.exe`
- It may be an outdated version
- It will be installed in the `texmf` tree where any customisations you choose to undertake may be overwritten if you decide to upgrade to a more recent version of MiKTeX later.

To end up with a working MusixTeX installation, your best bet is to install a "Small" MiKTeX and then add MusixTeX manually according to the procedure described in this document. If you later find that you need some of the things in the "Total" MiKTeX package, it's easy to add them using the "Packages" tab of the MiKTeX Options dialogue. However, if you allow Setup to install its incomplete MusixTeX version in `c:\texmf`, you will have to remove it manually – or manually replace it with a working installation and run the risk of having it overwritten during a subsequent MiKTeX upgrade.

3. Accept the option to install a shared MiKTeX environment. (This is the default setting.)
4. Accept `c:\texmf` as the installation folder unless you have a good reason to put your MiKTeX installation somewhere else. Ditto in the next screen for the shortcut folder name.
5. If you are installing a TeX package for the first time, you must create a `localtexmf` tree. If you like – for instance, if you prefer to keep your personal data files on a separate partition – you can create the `localtexmf` tree on a different partition from the `texmf` tree. Alternatively, just accept the default path and directory name for the `localtexmf` tree.
6. Accept the default option not to incorporate existing `texmf` folder trees. You don't have any if you're installing a TeX package for the first time.
7. Click "Next" to start the installation process.

1.4. Making a local binaries directory

Musix \TeX and related preprocessors come with their own .exe files, which must be placed in a directory where your operating system can find them. It's a good idea to create a separate directory to receive these files (as well as any other \TeX -related executables you may later find you need) rather than dumping them into `c:\texmf\miktex\bin` where they'll be all mixed up with the basic MiK \TeX binaries and difficult to find again if you need to upgrade them.

To make a local binaries directory:

1. Create a new directory on your hard drive, e.g. `c:\texbin`.
2. Now tell Windows where it is. There are three different options for doing so:
 - 2.1. The old-fashioned way:

Open your `autoexec.bat` and add the line: `set path=%path%;C:\texbin` to the end of the file. (This will preserve any pre-existing path statements in your `autoexec.bat`, but will also work if the file is empty to start with.) Save the change and reboot.
 - 2.2. The Windows way: In Control Panel, open the System Properties dialogue box. Select the Advanced tab and click on *Environment Variables*. You will see a dialogue box in which you can edit the User Variables and the System Variables.

If you want your local binaries directory to be accessible to all users on your computer, you need to edit the *Path* statement in the System Variables. To do so, select the *Path* variable in the System Properties list and click *Edit*. (You will notice that MiK \TeX has already added `c:\texmf\miktex\bin` to the System Path.) In the *Variable Value* text box, place the cursor at the end of the Path entry and add the string `;c:\texbin`. Click OK as often as necessary to exit System Properties. You do not need to reboot for the change to become effective.

Note: *If tampering with the System Path makes you nervous, you might want to select the entire System Path string and copy it into a text file before you make your change, so that you can restore the original entry if anything goes wrong. But don't be overly scared – even if you make a complete mess of the Path statement, your computer won't come to a grinding halt. Windows won't be able to find command-line programs any more, but the operating system as such will still function.*

- 2.3. If you are the only person using your computer and/or if you have a strong aversion to altering the System Path, you can create a User Path instead. To do so, click *New* below the list of User Variables. In the *Variable* text box, enter the word `Path`. In the *Variable Value* text box, enter the string `c:\texbin`. Click OK as often as necessary to exit System Properties. You do not need to reboot for the change to become effective.

2. Musix \TeX and friends

2.1. Where to get them

2.1.1. Musix \TeX

You can obtain Musix \TeX from the software pages of the Icking Music Archive (<http://icking-music-archive.org/software>). Look for a file called `musixtex.zip`.

2.1.2. Additional packages

In addition to basic Musix \TeX , a package of extensions and preprocessors is available which you may find useful or even indispensable for your music typesetting. All these add-ons are now bundled in the archive `muswin.zip`, which is also available from the software pages of the Icking Music Archive and which contains the following packages:

- `musixlyr`, for typesetting vocal music with lyrics
- `musixser`, which offers the option of entering music one voice at a time
- `PMX`, a preprocessor for Musix \TeX which simplifies the process of music coding
- `M-Tx`, a preprocessor for `PMX` which includes the option of entering lyrics
- the `psslur-K` archive containing utilities for Postscript slurs.

Please note that `muswin.zip` may not always contain the most recent versions of any of these packages. Updated ("beta") versions of individual packages may be found separately on the software pages of the Icking Music Archive: <http://icking-music-archive.org/software/indexmt6.html>. As the updates may contain essential improvements, fixes, etc., it is recommended to check whether more recent versions are available before installing the contents of `muswin.zip`.

2.2. Installing

1. Use your favourite unzipping tool to unpack **both** `musixtex.zip` and `muswin.zip` into the same temporary directory, for example `c:\musixtemp`.
2. Unzipping `muswin.zip` yields several subsidiary zip archives: `pmx*.zip`, `mtx*.zip`, `musixser*.zip`, `musixlyr*.zip`, and `psslur-k.zip`. Unzip each of these archives into `c:\musixtemp`.

You should now find that several new subdirectories have been created in your temporary directory: `bin`, `doc`, `dvips`, `metafont`, `pmxab`, `prepmx`, `scor2prt`, `tex`, and `tfm`.

3. In your `localtexmf` tree, create a series of subdirectories to receive Musix \TeX files.
 - 3.1. In `localtexmf`, create a subdirectory called `tex`. In `tex`, create a subdirectory called `generic`. In `generic`, create a subdirectory called `musixtex`. You should now have a directory path that looks like this:
`c:\localtexmf\tex\generic\musixtex`.
*(Note: Strictly speaking, you can skip the directory `generic` if you like. I put it in for the sake of tradition and symmetry – it reflects both the structure of the local `texmf` tree under previous versions of `MiK \TeX` and the structure of the `texmf` tree under `MiK \TeX 2.1`. What is crucial is that the `musixtex` directory **MUST** be located below the `localtexmf\tex` folder for `TEX` to be able to find the Musix \TeX macro files. In other words, a path such as `c:\localtexmf\tex\musixtex` or `c:\localtexmf\tex\music\musixtex` would be fine, but `c:\localtexmf\anyotherdir\musixtex` will not work.)*
 - 3.2. In `localtexmf\fonts\source`, create a subdirectory called `public`. In `public`, create a subdirectory called `musixtex`. You should now have a directory path that looks like this:
`c:\localtexmf\fonts\source\public\musixtex`.
 - 3.3. In `localtexmf\fonts\tfm`, create a subdirectory called `public`. In `public`, create a subdirectory called `musixtex`. You should now have a directory path that looks like this:
`c:\localtexmf\fonts\tfm\public\musixtex`.
 - 3.4. In `localtexmf`, create a subdirectory called `doc`.

4. Move the Musix \TeX files from your temporary directory to their destined locations.
 - 4.1. Move all *.exe files from `c:\musixtemp\bin` to your local binaries directory if you created one while installing MiK \TeX . If you prefer not to use a local binaries directory, move the *.exe files to `c:\texmf\miktex\bin`.
 - 4.2. Move all *.tex and *.sty files from `c:\musixtemp\tex` to `c:\localtexmf\tex\generic\musixtex`.
 - 4.3. Move all *.mf files from `c:\musixtemp\metafont` to `c:\localtexmf\fonts\source\public\musixtex`.
 - 4.4. Move all *.tfm files from `c:\musixtemp\tfm` to `c:\localtexmf\fonts\tfm\public\musixtex`.
 - 4.5. Move the file `psslurs.pro` from `c:\musixtemp\dvips` to `c:\localtexmf\dvips\base` (if this directory doesn't exist in your localtexmf tree, create it).
 - 4.6. Move everything in the `c:\musixtemp\doc` directory to `c:\localtexmf\doc`.
5. Go to the MiK \TeX Options dialogue and refresh the File Name Database to tell MiK \TeX about all its beautiful new files.
6. Read the documentation.

To refresh the MiK \TeX File Name Database: In your Start menu, navigate to Programs / MiK \TeX / MiK \TeX Options, select the "General" tab, and click "Refresh Now".

Musix \TeX , PMX, M-Tx, Musixlyr, Musixser and the K type PostScript slurs* are now installed and ready to use. There will be some files left over in your temporary directory, but you don't need any of them for running Musix \TeX on a Windows platform.

3. Type 1 Fonts for Musix \TeX

For creating PDF and Postscript versions of Musix \TeX output, Type 1 fonts frequently provide better results (and smaller output files) than the bitmapped fonts which are part of the regular Musix \TeX package.

3.1. Where to get them

Musix \TeX Type1 fonts are available from the software page of the Icking Music Archive in the file `musixps.zip`.

3.2. Installing

1. Unzip `musixps.zip` into a temporary directory. Make sure the directory structure remains intact when unzipping (check "Use folder names" in WinZip, type "`unzip -a musixps.zip`" if running `pkzip`).

When correctly unpacked, the Zip archive will yield three subdirectories: `dvipdfm`, `dvips`, and `pfb`. The first two directories will each contain a file called `musix.map`. The `pfb` directory will contain the Type1 font files. Of the files in the top-level directory, you should read the files `readme.txt` and `readme_mik.txt`. The batch files can be used to automate the installation process if you edit the settings for `texmf`, `localtexmf` and your local binaries directory to reflect the location of these directories on your system. You can also install the fonts by hand as follows.

2. Make sure that your `texmf` and `localtexmf` trees contain the following directories (if any of

* As of the writing of this document, the version of `psslurs.tex` included in `muswin.zip` contains a bug which is fixed in a newer version available separately on the Icking Archive software page. To replace the faulty file, download `musixps92a.zip`, unzip, and, in `c:\localtexmf\tex\generic\musixtex`, overwrite the existing `psslurs.tex` with the bugfix version. (The bugfix version is dated 23.06.2002.)

them don't exist, create them now):

- 1.1. `c:\localtexmf\fonts\type1\musixtex`
- 1.2. `c:\texmf\dvips\config`
- 1.3. `c:\texmf\dvipdfm\config`
2. Move the files from your temporary directory to their destined locations.
 - 2.1. Move all *.pfb files to `c:\localtexmf\fonts\type1\musixtex`.
 - 2.2. Move the file `musix.map` which is in the `dvips` subdirectory of your temporary directory to `c:\texmf\dvips\config`.
 - 2.3. Move the file `musix.map` which is in the `dvipdfm` subdirectory of your temporary directory to `c:\texmf\dvipdfm\config`.
3. Edit the config files.
 - 3.1. In `c:\texmf\dvips\config`, open `config.ps` in a text editor and add the line:
`p +musix.map`
 - 3.2. In `c:\texmf\dvips\config`, open `config.pdf` in a text editor and add the line:
`p +musix.map`
 - 3.3. In `c:\texmf\dvipdfm\config`, open the file `config` in a text editor and add the line:
`f musix.map`
4. Refresh the MiKTeX File Name Database.

4. Type M PostScript slurs

4.1. Where to get them

If you wish to use Hiroaki Morimoto's PostScript slurs, you may download the most recent version from <http://homepage1.nifty.com/kuuku/notation/index-e.html> in a gzipped TAR archive called `musixpss-0.50.tar.gz`. (WinZip can unpack this archive format.)

4.2. Installing

In order to use these slurs, you must first install MetaPost in your TeX system. Here's how to do this in MikTeX 2.1:

1. Go to the MikTeX Options dialogue box and select the "Packages" tab.
2. Select your Download Site.
 - 2.1. Click on the "Change" button next to the "Download Site" text box.
 - 2.2. In the "Change Download Site" dialogue box, choose "Internet" if you don't have the installation files for the full MikTeX installation anywhere on your computer, or select the local folder that contains the installation files.
 - 2.3. Click "Finish" to set the download site.
3. Now select the packages you want to install.
 - 3.1. Go to the directory tree in the "MikTeX Packages" pane. Expand "Formats" by clicking on the plus sign next to the word, then do the same for "ConTeXt" and "MetaPost".
 - 3.2. Click the check box next to "MetaPost" to select the package. Make sure that both subfolders, "MetaPost base" and "MetaPost executables" are selected.
 - 3.3. Click "Apply" to install the MetaPost package.
 - 3.4. Refresh the MikTeX File Name Database.

Now you're ready to install the actual slurs.

1. Use WinZip (or another unpacker than can handle tarballs) to unpack `musixpss-0.50.tar.gz` into a temporary directory, e.g. `c:\musixtemp`.

2. Move `musixpss.tex` to `c:\localtexmf\tex\generic\musixtex`.
3. Move `musixpss.mp` to `c:\localtexmf\metapost\generic\musixtex` (you will probably need to create these subdirectories first if they don't exist in your localtexmf tree).
4. Move `musixpss.exe` to your local binaries directory if you have one. If you prefer not to use a local binaries directory, move the file to `c:\texmf\miktex\bin`.
5. Move the documentation file `musixpss_e.html` from `c:\musixtemp\doc_en` to `c:\localtexmf\doc\musixtex`.
6. Refresh the MikTeX File Name Database.

5. Upgrading an existing MusixTeX installation

To upgrade an existing MusixTeX installation, download the latest version of MusixTeX from the same place you got your previous versions and unzip to a temporary directory. Then simply overwrite the old MusixTeX files with the new ones, following the same procedure (described in 2.2 above) as when you installed the package for the first time.

6. Writing/compiling MusixTeX source files

To create input files, all you really need is a text editor. However, there are many TeX editors (a.k.a. TeX shells) available that make the job a bit easier by providing syntax highlighting, toolbar buttons for easy access to TeX executables, etc. Some of these editors are freeware, and many are preconfigured for MiKTeX by default. (None, however, will come preconfigured for a MusixTeX installation. You will need to **carefully read the documentation** of the editor you choose in order to set it up to work with the MusixTeX executables.)

When compiling MusixTeX files, you will be invoking `tex.exe`, `musixflx.exe`, the preprocessor executables `pmxab.exe` and `mtx.exe`, an output viewer (which, in MiKTeX, is `yap.exe`), and other utilities such as `dvips`, `dvipdfm`, etc. "Configuring your editor" to work with these programs means setting up a way to start these programs and run them on your source file without having to exit the editor. This will typically involve a toolbar button, dropdown list entry, or keyboard shortcut for each program you wish to invoke. Your editor's online help will tell you how to set these things up. It's up to you to find this information in the online help; it will certainly be there to be found, but may go by different names ("TeX Programs", "User Tools", "Compiler Type", etc.) in different editors.

Browse the links at <http://www.miktex.org/links.html> as a starting point to find a TeX editor you like. Some of the more popular ones include WinEdt (many bells and whistles; shareware; turns into increasingly annoying nagware after 30 days), TeXnicCenter (many bells and whistles; freeware; **WARNING:** the last beta this writer tested had a tendency to overwrite a standalone MusixTeX source file with a default LaTeX master file and did this without warning, without saving a backup, and without any visible means of disabling this behaviour), and WinTeX2000 (not quite as many bells and whistles, but more than enough for MusixTeX; easier to customise than WinEdt; not free, and refuses to start after 30 days' free trial). There is also the powerful freeware Crimson Editor (<http://www.crimsoneditor.com>), which supports a number of different programming languages and may support MusixTeX syntax one day if someone creates the necessary custom syntax files; meanwhile the setting for LaTeX, while not ideal, will serve for MusixTeX purposes.