

Installing and Running MusiX_{TEX} in MS Windows

Andre Van Ryckeghem, Don Simons

Contents

1 Introduction	1
2 Getting Started - Naming Directories	1
3 Installing MusiX_{TEX} Files	3
3.1 Option A: Using a batch file	3
3.2 Option B: Manually transferring files	3
4 Refresh the File Name Database	4
5 At Last, Running MusiX_{TEX} and Viewing a Score	4

1 Introduction

This document explains in detail how to install and run MusiX_{TEX} under MS Windows XP or Vista¹. Along the way it includes information about locations of the various file types, which may be useful for any future incremental MusiX_{TEX} upgrades. Sections 2-4 provide this setup information in detail. The MusiX_{TEX} language, and installation and use of the preprocessors **PMX** and **M-Tx**, are all covered elsewhere.

In order to use MusiX_{TEX} you must have installed other supporting software, including a **TEX** system, text editor, and a postscript viewer. This document assumes the **TEX** system MiK_{TEX} 2.8 is already installed, but should apply at least to versions 2.6 and 2.7 as well. The recommended postscript viewer is GSview (the executable is gsview32.exe). It can also serve to produce PDF files from postscript. Some of the batch files in this distribution assume GSview is present.

MiK_{TEX} 2.8 comes with a development environment called **TEXWorks**. In principle it can be made to run all the necessary codes in sequence, but setting it up is very demanding, so it will not be discussed here. Instead, Section 5 details all the individual steps needed to produce a PDF file of a music score, and describes various batch files that can help automate the process.

MiK_{TEX} itself comes with a version of MusiX_{TEX}. This will usually be out of date. It is suggested to leave these files alone. By installing the latest version of MusiX_{TEX} according to the guidelines given here, you will supersede the version supplied with MiK_{TEX}. It is also suggested that if MiK_{TEX} ever offers to update your MusiX_{TEX} installation or provide a missing file, you politely decline, find the missing file yourself on the Icking archive, and position it in your own MusiX_{TEX} installation.

2 Getting Started - Naming Directories

To begin installing MusiX_{TEX}, create a **DOWNLOAD** folder of your choosing and unzip the MusiX_{TEX} distribution into it, being sure to select "use folder names" when unzipping.

Next, you have to decide what directories various categories of files will permanently reside in. The names are entirely at your discretion. It is possible but probably best not to put them in the same folders with the corresponding MiK_{TeX} files. The reason is that if MiK_{TeX} is upgraded to

¹Most of this information can be applied to other Windows versions or operating systems with little or no modification. With more volunteer effort, specifics for other OS's could be added

a newer version, the MusiX_TE_X files will disappear. Some suggested names are given in the table below:

Category	Description	Short Names	Windows Conventional
DOWNLOAD	Unzip musixtex.zip here	c:\musix	<userdocs>\mtunzip
TOOLS	executables	c:\tools	<userprogs>\MusiX _T E _X
ROOT	other MusiX _T E _X files	c:\musixtexmf	<userdata>\MusiX _T E _X
WORK	working files	c:\musix\mymf	<userdocs>\MusiX _T E _X

The suggested “Windows Conventional” names are different in XP or Vista, as given in the table below. Non-English versions of Windows may use different names for the pre-existing folders; you’ll have to make appropriate adjustments.

Windows XP	
<userdocs>	c:\Documents and Settings\<owner>\My Documents
<userprogs>	c:\Program Files
<userdata>	c:\Documents and Settings\<owner>\Application Data
Vista	
<userdocs>	c:\users\<owner>\My Documents
<userprogs>	c:\users\<owner>\My Programs
<userdata>	c:\users\<owner>\Application Data

As mentioned once already, the choice of folder names is entirely at your discretion. One of the authors of this guide advocates the “Short Names” (because they are so much simpler), while the other prefers “Windows Conventional” (because they conform with windows standard locations for executables and data files). Users of certain non-English versions of Windows who choose the “Windows Conventional” route will have to make some adjustments to account for the fact that standard folder names such as “My Documents” are different.

The remainder of this section provides further detail about the directories.

The TOOLS directory will hold executables and must be placed in the Windows path. One way to do this that always works in XP or Vista is to right click on the “My Computer” desktop icon, left click on Properties|Advanced|Environment Variables, in the “System Variables” section scroll down to “path”, select it, click edit, and append the full path name you have selected for the TOOLS directory. Note that if there are any command windows open at the time you do this, the path change will not affect them; it will only affect newly opened command windows. It may be tempting to avoid this path-setting exercise by using the directory that MiK_TE_X already established for its executables and placed in the path, viz., c:\Program Files\MiK_TE_X 2.8\miktex\bin; however as noted above, this is not recommended because this directory will be overwritten every time MiK_TE_X is updated.

The ROOT directory will house all of the MusiX_TE_X files except the executables and your personal working files. MiK_TE_X needs to know where these files are. To accomplish that, you must run the MiK_TE_X “Settings” program, which is found under Start|Programs|MiK_TE_X 2.8|Maintenance. After starting the program, select the “Roots” tab, click the “Add” button, and enter the full path name for your MusiX_TE_X ROOT directory.

A final directory, WORK, houses user-created MusiX_TE_X source files. It need not ever be formally specified, and can change at will. It is simply the place you keep your current working files.

3 Installing MusiX_TE_X Files

3.1 Option A: Using a batch file

For adventurous and trusting souls, there is a batch file `INSTMUS.bat` in `DOWNLOAD\InstallMusiXTeX\installWindows` that will copy all files to the proper locations and do the necessary editing, but before running it you must edit it with your selected directory path names. The first `SET` command in the file defines the `DOWNLOAD` directory; the second, `TOOLS`; the third, `ROOT`, and the fourth, the `MiKTeX` root directory. Caution: any path name that contains spaces must be surrounded by double quotes when entered in the the batch file. If you are uncertain what to enter for the `MiKTeX` root, you should run `MiKTeX Settings` as explained above, select the “Roots” tab, and check the “Show MiKTeX-maintained root directories” box if it is not checked. The path you want is labeled either “CommonInstall” or “Install”. If you find both then check to see which one houses `MiKTeX`.

The reason you need the `MiKTeX` root directory is that some files from there need to be copied, edited, and the edited version placed in your `ROOT` directory. The batch file does all of this.

Once you have carefully edited `INSTMUS.bat`, run it either from the command line or by double-clicking it in Windows Explorer.

3.2 Option B: Manually transferring files

Instead of using the batch file, you could perform all the steps manually. All the necessary steps are enumerated below.

1. Copy all the files `DOWNLOAD\fonts\mf*.mf` into the folder `ROOT\fonts\source\public\musixtex`
2. Copy all the files `DOWNLOAD\fonts\tfm*.tfm` into the folder `ROOT\fonts\tfm\public\musixtex`
3. Copy all the files `DOWNLOAD\tex*.*` into the folder `ROOT\tex\generic\musixtex`
4. Copy all the files `DOWNLOAD\musixtexadd*.tex` into the folder `ROOT\tex\generic\musixtex`
5. Copy all the files `DOWNLOAD\bin*.*` into the `TOOLS` directory.
6. Copy all the files `DOWNLOAD\fonts\type1*.pfb` into the folder `ROOT\fonts\type1\public\musixps\type1`
7. Copy the pdf configuration file `c:\Program Files\MiKTeX 2.8\dvipdfm\config\config` to `ROOT\dvipdfm\config\config`. Edit the copied file by adding the line
`f musix.map`
8. Copy the file `DOWNLOAD\MUSIXTEXdistribution\dvipdfm\musix.map` to `ROOT\dvipdfm\config\musix.map`. Edit the copied file by adding the line
`f musix.map`
9. Copy the postscript configuration file `c:\Program Files\MiKTeX 2.8\dvips\config\config.ps` to `ROOT\dvips\config\config.ps`. Edit the copied file by adding the line
`p +musix.map`

10. Copy the file `DOWNLOAD\MUSIXTEXdistribution\dvips\musix.map` to `ROOT\dvips\config\musix.map`
11. Copy the file `DOWNLOAD\MUSIXTEXdistribution\dvips\psslurs.pro` to `ROOT\dvips\base\psslurs.pro`
12. Copy the file `c:\Program Files\MiKTeX 2.8\dvips\tetex\config.pdf` to `ROOT\dvips\tetex\config.pdf`. Edit the copied file by adding the line `p +musix.map`

4 Refresh the File Name Database

Whether you have transferred and edited files manually or using the batch file, one final chore is to inform MiKTeX where the MusiXTeX files are located. Do this by running the MiKTeX “settings” program from the “start” menu and clicking the button labelled “Refresh FNDB” (File Name DataBase).

ALWAYS REMEMBER TO REFRESH THE FILE NAME DATABASE whenever you add a file to MusiXTeX.

5 At Last, Running MusiXTeX and Viewing a Score

To test your setup, copy the sample file `DOWNLOAD\TestTeXInputs\test1.tex` to your working directory. MusiXTeX operates from the command line. The most straightforward way to run it in Windows XP or Vista is from a command window, which is opened by clicking from `start|run`, then entering “cmd”. In the command window, navigate to your working directory. MusiXTeX is a 3-pass system. For the first pass, enter `tex test1`. This should produce `test1.mx1`. Next, enter `musixflx test1`. This should produce `test1.mx2`. Finally, enter `tex test1`. This should produce `test1.dvi`. The basic processing is now complete.

The files `test1.mx1` and `test1.mx2` are intermediate working files produced by T_EX (on its first pass) and `musixflx` respectively. Once the 3-step process is complete, they will never be needed again and should be deleted; in fact, if you have made any changes to your MusiXTeX file, you *must* delete `test1.mx1` before repeating the 3-step process. A great deal more information about the process is included in the MusiXTeX manual in

`DOWNLOAD\MusiXTeXDistribution\doc\musixtex>manual\musixdoc.pdf`.

To view the result, you should create a postscript file by typing `dvips test1`. This should produce `test1.ps`, which can be viewed by running GSview from the Windows GUI. Note that all T_EX systems include a “DVI viewer” such as YAP. These will not display Type K postscript slurs, and so are generally not recommended.

As with file copying and editing, the above steps can be accomplished with batch files. The initial 3-step process `tex ⇒ musixflx ⇒ tex` is accomplished with the batch file `musixtex.bat`, which should have been copied into your TOOLS directory, and should work on any Windows system without modification, provided the TOOLS directory is in the path. The batch file needs a single argument, the base name of your typesetting project. This batch file checks for the `.mx1` and `.mx2` files and deletes them if they are present.

The file `mups.bat` does everything `musixtex.bat` does, and then produces a postscript file from the `.dvi` by running `dvips`. You may wish to edit the file to (a) produce letter size output rather than A4, which is accomplished by changing the `dvips` option “-t a4” to “-t letter”, and (b) adjust the page image offsets. For example, PMX-generated MusiXTeX files with default sizing will be properly centered on letter paper if you add the `dvips` option “-O0.25in,-0.22in”.

We recommend using GSview to view the resulting postscript. It runs from the Windows GUI. When iteratively editing a MusiXTeX file, it is very convenient to simply leave GSview

running, because once a particular postscript file is loaded, whenever the file changes, GSview will *automagically* renew it. And when you finish your masterpiece, if you wish to distribute it, you can use GSview to produce a PDF file. This is done from File|Convert, then select “pdfwrite”, click “OK”, and provide a file name.

Finally, once MusiX_{TEX} is set up, if you would like to avoid the command window, there is a way that you can operate entirely from the Windows GUI. But to do so, you’ll probably want to keep every individual typesetting project in a separate working folder, and that folder will have to contain a copy of `mupsall.bat`. You may then run the batch file by double-clicking on it from Windows Explorer. The reason for having a separate folder for each project is that to avoid having a command line argument, `mupsall.bat` simply runs `mups.bat` on every `.tex` file in the folder.

Users of preprocessors M-Tx and/or PMX may wish to create their own batch files, starting with `mudvi.bat`. Drawing on examples already in the file, simply add commands to run the desired preprocessor(s) and perform the necessary error checks. This is discussed a little more in the MusiX_{TEX} manual.

For advanced users, the two other batch files in the TOOLS directory, `lapsview.bat` and `lapdfview.bat`, may be used to process LaTeX files that contain MusiX_{TEX}. They will produce postscript and PDF files respectively. Combining MusiX_{TEX} and LaTeX is not for the beginner. These files will not be discussed further here.