

C J Willers \*

# Setting up WinEdt

August 26, 2020



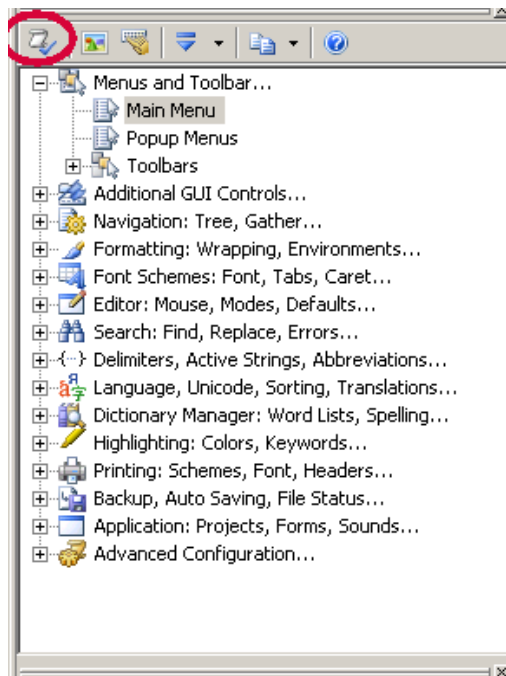
# Chapter 1

## Setting up WinEdt

### 1.1 Install WinEdt

This document assumes that WinEdt v.10.0 is installed.

After you changed an ini file, the changes can be immediately be made available by clicking on the load button:



After you make changes to a particular script you should use the Load Command (the first button in the Option Interface Toolbar) to make the changes effective immediately. It is not necessary to restart WinEdt. In fact, no scripts are loaded at startup: the compiled raw data is stored in WinEdt.dnt (Do Not Touch). This reduces the startup time and reduces the likelihood of error messages during startup.

I am not quite sure how to activate the changes for the next time you start WinEdt. It seems that exporting to WinEdt.dnt have the effect of saving the changes to that these are working next time you load WinEdt.

### 1.2 Viewer Setup

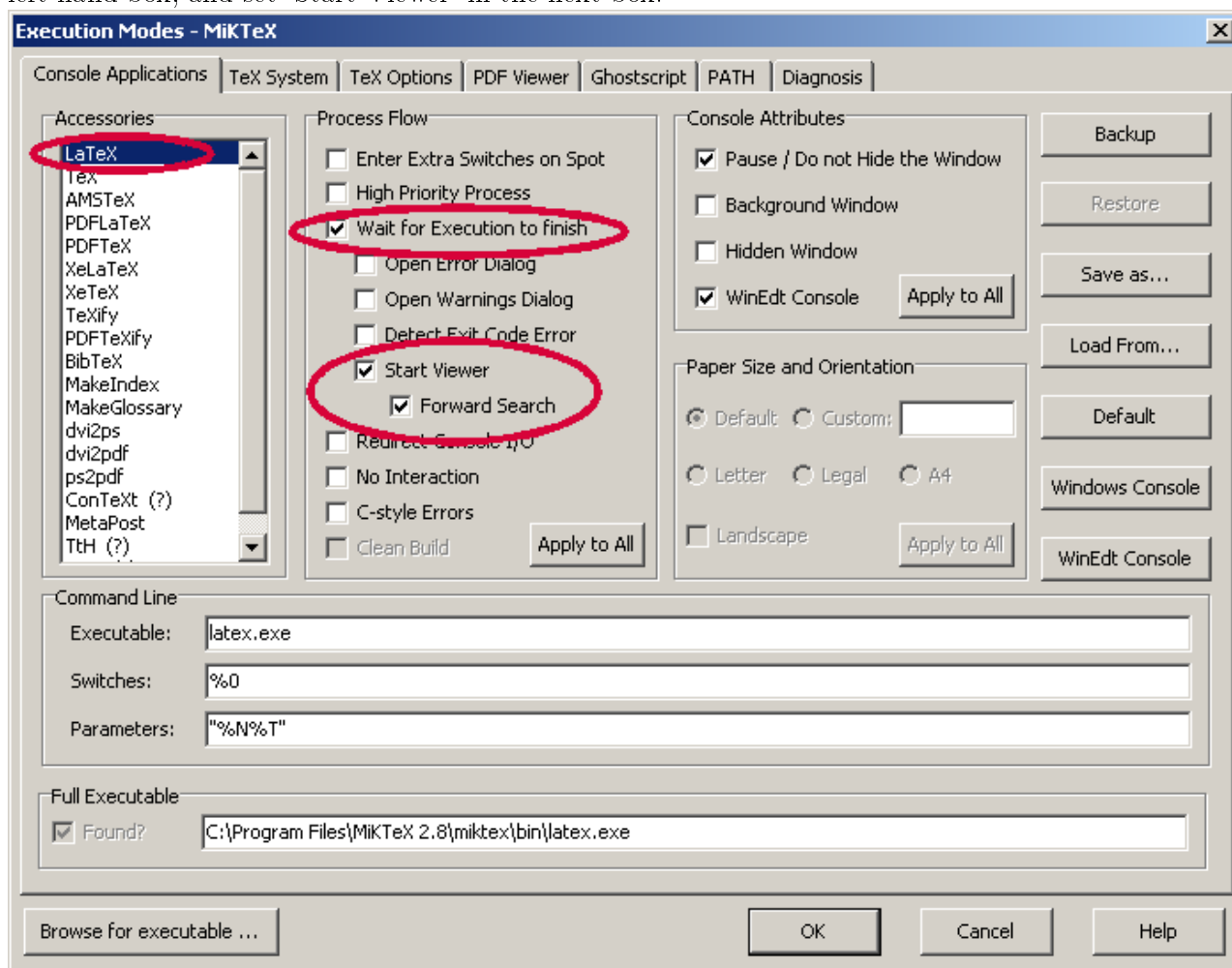
Yap seemed to fail on my WinEdt install.

MikTeX provides a DVI viewer, called YAP that integrates well with WinEdt. If properly set up, WinEdt and YAP are synchronised. When you run the  $\text{\LaTeX}$  compiler, YAP will open at the location of the cursor in the WinEdt file. When you double click in the YAP window, the text cursor in WinEdt will move to the corresponding location.

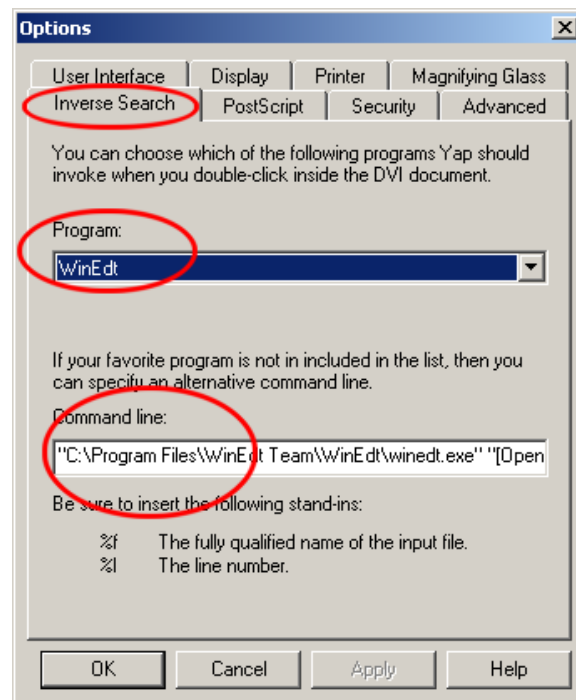
The Yap binary is located in the MikTeX bin file, on my PC at `C:\ProgramFiles\MikTeX2.9\miktex\bin\x64`

<Options><Execution Modes>, select 'LaTeX'. If not set, set 'Wait for execution to finish' and select the 'Start Viewer' and 'Forward Search' boxes. The 'Start Viewer' selection will start the viewer after completion of the compilation process, and the 'Forward Search' selection will move a small, round, cursor to the location in the DVI viewer, where the cursor is in the LaTeX document (when it makes sense).

Other viewers can be started in a similar manner, just select the appropriate compiler in the left-hand box, and set 'Start Viewer' in the next box.



In YAP go to <View><Options><Inverse Search>, select WinEdt, if not yet selected. If option 'WinEdt' does not show in the dropdown box, you have to enter the path to the application in the text box at the bottom of the dialog box. On my installation, the entry is  
`"C:\Program Files\WinEdt Team\WinEdt 10\winedt.exe" "[Open(|%f|);SelPar(%1,8)]"`

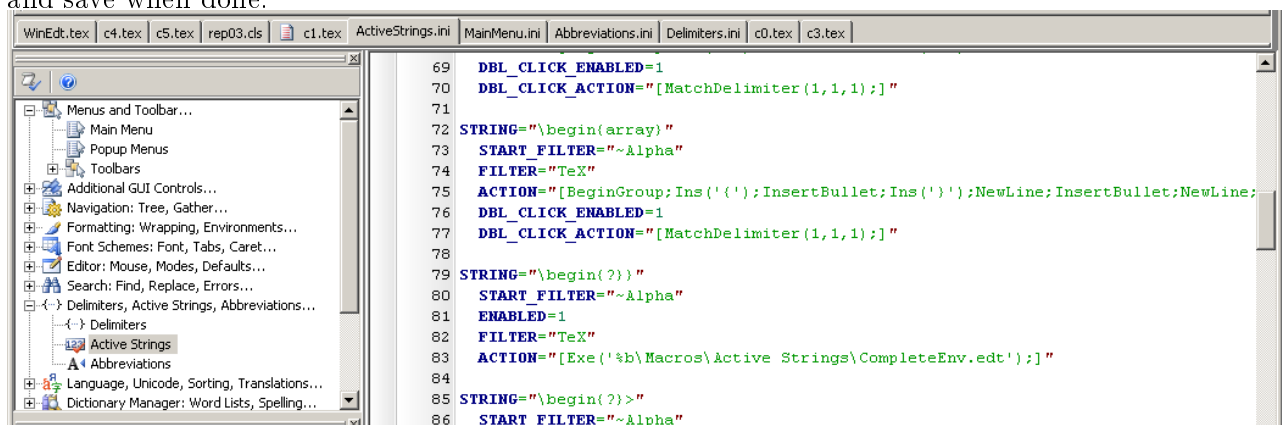


These PDF viewers are currently available on my PC:

```
C:\Program Files\Tracker Software\PDF Viewer\PDFXCview.exe
C:\Program Files\PDF Architect 7\architect.exe
C:\Users\nwillers\AppData\Local\SumatraPDF\SumatraPDF.exe
```

## 1.3 Enabling Active Strings for begin-end Pair

It seems that the default settings are acceptable, no edit required. If you do need to change anything, go to <Options><Options Interface><Delimiters, Active Strings, ...><Active Strings>, edit the macro file just opened to enable "`\begin{?}`". It might already be activated. Edit as required and save when done.



## 1.4 Setting up the Dictionary

<Options><Options Interface><Dictionary Manager:....><Word Lists (.)...>, edit the macro file just opened. You enable or disable sub-dictionaries according to your requirement.

If you want both the UK and US English dictionaries, keep all of the below dictionaries on. You can then select the language on a per-file basis. Use embedded emacs-style mode and submode commands as the first line of each file. For example, use

```
% -*- TeX -*- -- US -- Soft -- for US spelling, or
```

```
% -*- TeX -*- -- UK -- Soft -- for UK spelling in the tex file.
```

For more information search the WinEdt documentation for “modes and submodes”.

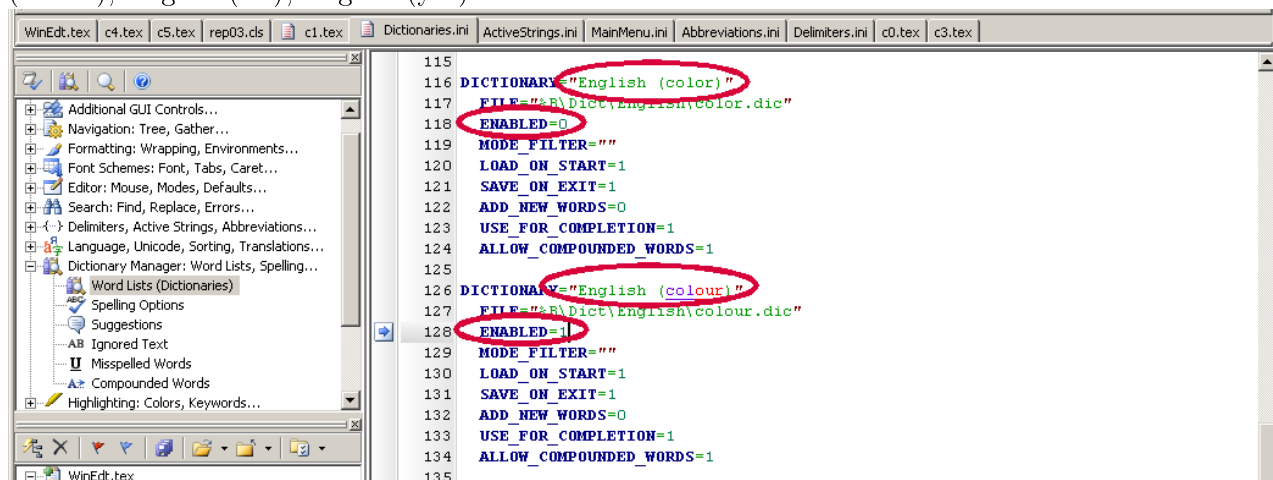
See also

```
\url{https://www.gnu.org/software/emacs/manual/html_node/emacs/Specifying-File-Variables.html}
\url{https://www.gnu.org/software/emacs/manual/html_node/emacs/Choosing-Modes.html}
```

If you require only UK English, as in in South Africa, select:

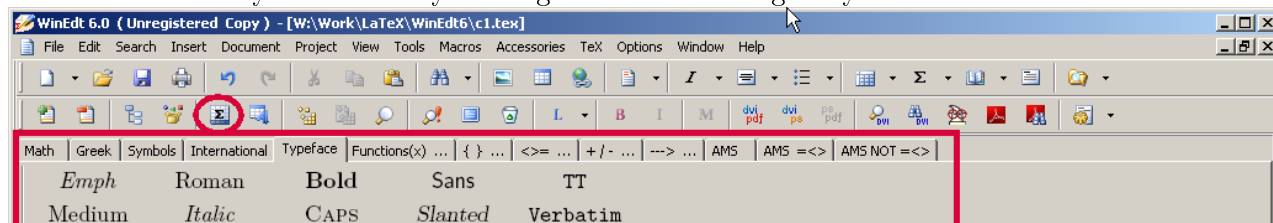
**Enable the following dictionaries:** User (Addon), EDT, LaTeX, WinEdt, English (common), English (colour), English (labelled), English (centre), English (ise), English (yse).

**Disable the following dictionaries:** English (Small), English (color), English (labeled), English (center), English (ize), English (yze). Save the file when done.



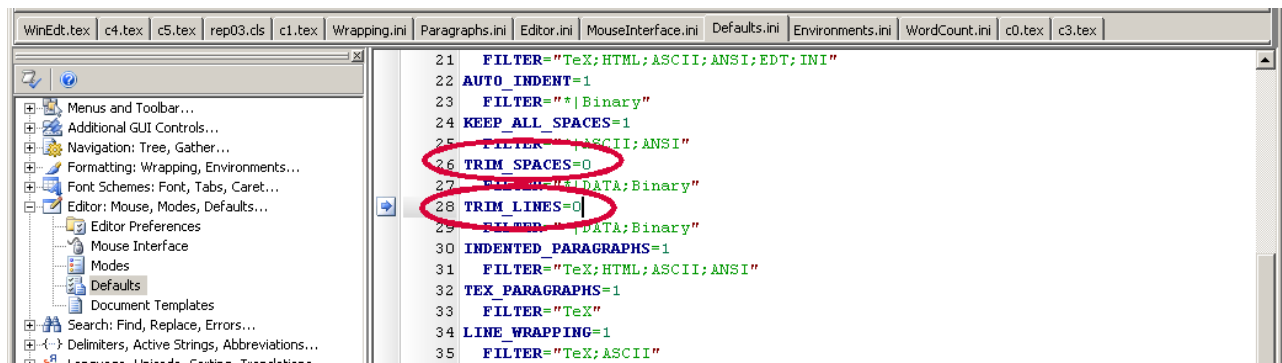
## 1.5 TeX Symbols GUI

Activate the TeX Symbols GUI by clicking on the  $\Sigma$ -on-the grid symbol.



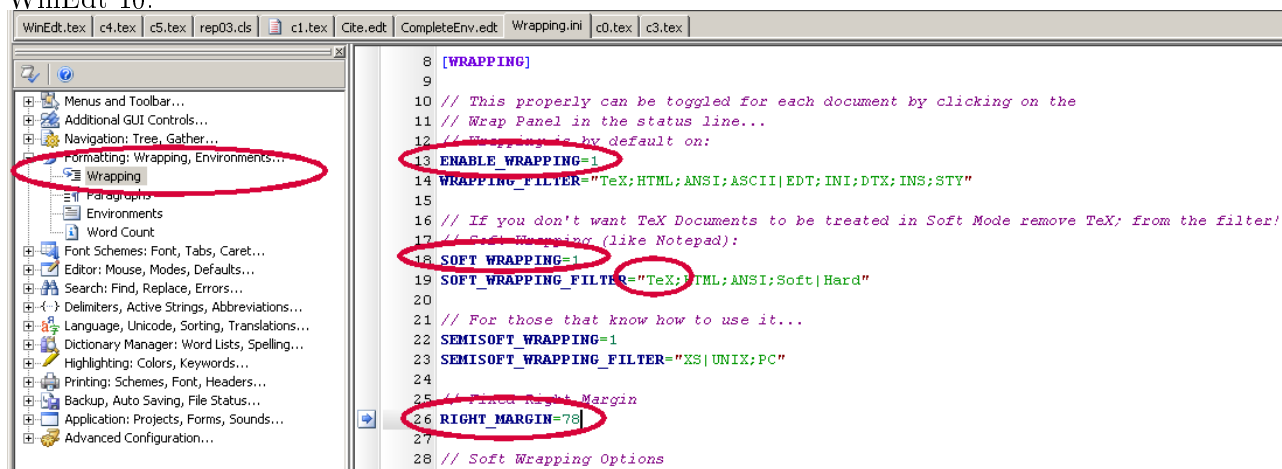
## 1.6 Keep Whitespace at EOF and EOL

<Options><Options Interface><Editor: Mouse, Modes,...><Defaults>, edit the macro file just opened. Scroll down to 'Trim Spaces' and 'Trim lines' and disable these two (make the values 0) — this is stop WinEdt from removing spaces at the end of lines and files.



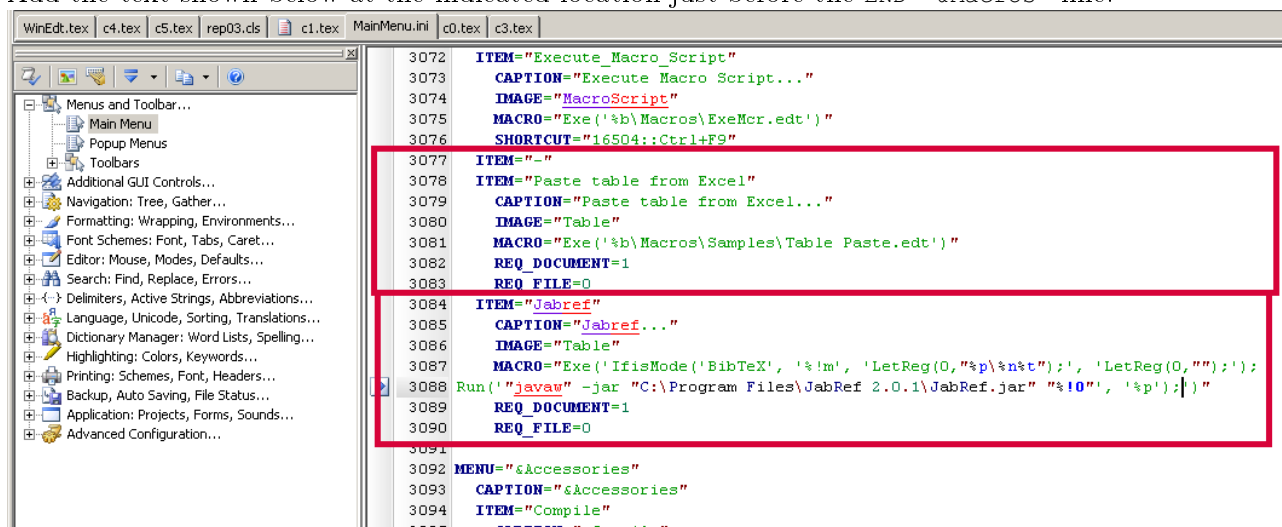
## 1.7 Set Text Wrapping and Text Width

<Options><Options Interface><Formatting, Wrapping, Environments...><Wrapping>, edit the macro file just opened. Switch on or off if required and set the right margin. Defaults work fine in WinEdt 10.



## 1.8 Add a Table-Paste Macro to a Menu

<Options><Options Interface><Menus and Toolbar><Main Menu>, edit the macro file just opened. Add the text shown below at the indicated location just before the END="&Macros" line.



This picture does not shown the END="&Macros" line It is required).

The picture above shows installing JabRef, this is optional if installed.

Listing 1.1: Macro to paste table from Excel

```
ITEM="Paste table from Excel"
CAPTION="Paste table from Excel..."
IMAGE="Table"
MACRO="Exe('%b\Macros\Samples\Table Paste.edt')"
REQ_DOCUMENT=1
REQ_FILE=0
```

Copy and paste [best to do this from the LaTeX source file] the text in Listing 1.1. If it does not work, check the MACRO line: take care to use the upquote (', next to the Enter key) and not the curly quote (').

Then copy the following file (Table Paste.edt) from the repo files folder (copy all three edt files) to the C:/Program Files/WinEdt Team/WinEdt 10/Macros/Samples or C:/Users/nwillers/WinEdt Team/WinEdt 10/Macros/Samples folder.

```
// %*- ASCII -*- -*- EDT -*-
//
// Paste Table and covert it to LaTeX format:
// Tab -> &Tab
// EOLN -> \\EOLN

Requires(20010317); // Requires WinEdt 5 Build 20010317 (or later)

CopyFromClipboard(0); // Get the Clipboard Text Data (eg. Excel)

// Convert Tab to &Tab
LetRegNum(2, -2);
Loop(!|>
    FindInString("%!0", "$[#9]$", 1,2, 1011, %!2+2);>
    IfOK(!'ReplaceInString("%!0", "&\\&9;", %!1, %!2, 1, 0);','!'Stop;')|);

// Convert EOLN to \\EOLN
LetRegNum(2, -3);
Loop(!|>
    FindInString("%!0", ">", 1,2, 1011, %!2+3);>
    IfOK(!'ReplaceInString("%!0", "\\>", %!1, %!2, 1, 0);','!'Stop;')|);

InsText("%!0"); // Insert with no wrapping

End;

// WinEdt also has a Block/ Column mode selection that can come
// handy when manipulating aligned tables...
```

## 1.9 Add a Back Slash to Forward Slash Macro to a Menu

<Options><Options Interface><Menus and Toolbar><Main Menu>

Search down for the entry labelled MENU="&Accessories". In the section just before Accessories, add the following:

```
ITEM="Backslash to Forward"
CAPTION="Backslash to Forward..."
IMAGE="Table"
MACRO="Exe('%b\Macros\Samples\bstofs.edt')"
REQ_DOCUMENT=1
REQ_FILE=0
```



just before the `END="%&Macros"` line.

Add the file (`bstofs.edt`) with the following contents to `C:/Program Files/WinEdt Team/WinEdt 10/Macros/Samples/bstofs.edt`

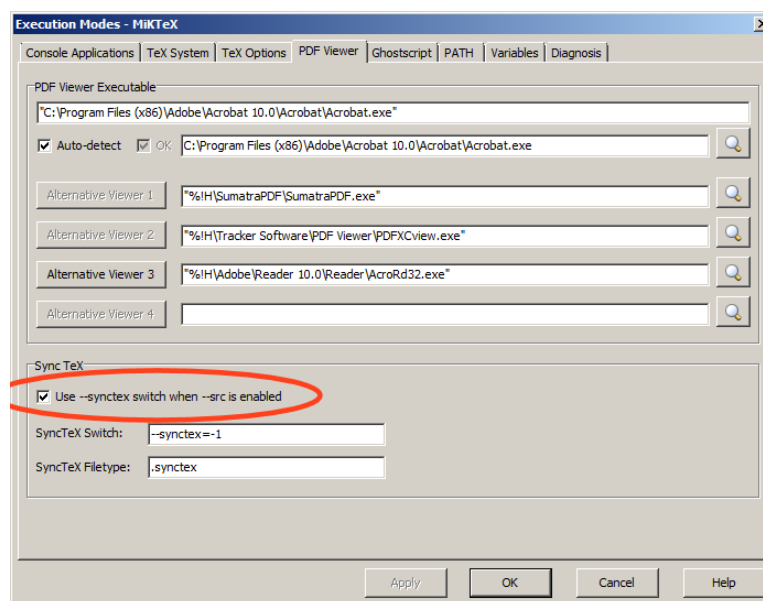
```
SetFindStr("\");
SetReplaceStr("/");
SetSearchCaseSensitive(0);
SetSearchRelaxed(0);
SetSearchWholeWords(0);
SetSearchInline(0);
SetRegEx(0);
SetSearchSelected;
SetSearchCyclic(0);
SetSearchForward(1);
SetSearchEntire(0);
SetReplaceRespectCaps(1);
SetReplacePrompt(0);
SearchReset;
ReplaceAll;
// ReplaceDialog;
```

---

## 1.10 Disable Synctex

In WinEdt10 the default settings for LaTeX is to compile with option `-synctex=-1` and, in WinEdt5.6, the default settings for LaTeX is without the `-synctex` option. However, MiKTeX does not support the `synctex` option for filenames with spaces.

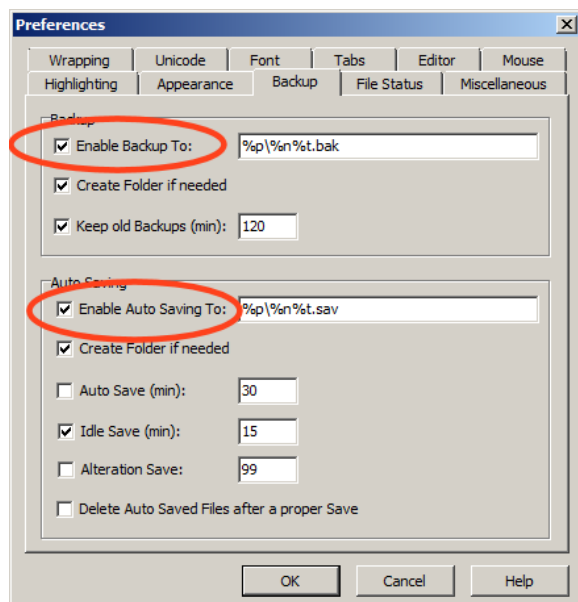
The solution is to open WinEdt and uncheck 'Use `-synctex` switch' in <Options> <Execution Modes><PDF Viewer>.



In the latest installs on Windows 7 it seems that this is no longer required; there can be spaces now. Experiment and find the best solution.

## 1.11 Disable Autosaving and Backup

Disable auto-saving and keeping backup files, by <Options> <Preferences><Backup> and unchecking the relevant boxes.



## 1.12 Adding to TeX Symbols

You may want to add new buttons to the TeX symbols Typeface GUI. Rename the existing Typeface.bmp to Typeface-old.bmp in the following folders, and then add the new (bigger) Typeface.bmp from the repo files folder to the same GUI folders. Copy the appropriately sized bitmap according to the folder structure.

<i>Emph</i>	Roman	<b>Bold</b>	Sans	TT	url	inline	index	textsuper
Medium	<i>Italic</i>	CAPS	<i>Slanted</i>	<b>Verbatim</b>	TBC	scmnd	nomen{}	textsub

```
C:\ProgramFiles\WinEdtTeam\WinEdt10\Bitmaps\Gui
C:\ProgramFiles\WinEdtTeam\WinEdt10\Bitmaps\Gui125
C:\ProgramFiles\WinEdtTeam\WinEdt10\Bitmaps\Gui200
or (replace with your user name)
C:\Users\nwillers\WinEdtTeam\WinEdt10\Bitmaps\Gui
C:\Users\nwillers\WinEdtTeam\WinEdt10\Bitmaps\Gui125
C:\Users\nwillers\WinEdtTeam\WinEdt10\Bitmaps\Gui200
```

<Options><Options Interface><Additional GUI Controls...><TeX Symbols>, edit the macro file just opened. Locate this section corresponding to the Typeface part of the GUI and replace the entire block with the following block:

```
PAGE="Typeface"
CAPTION="Typeface"
CONFIG_FILTER=" "
```

```

MODE_FILTER=""
GROUP="Typeface.bmp"
TOP=0
SPACE=0
ROWS=2
COLUMNS=9
WIDTH=111
HEIGHT=32
ITEM="\emph{...}"
    MACRO="Exe('%b\Macros\Fonts\Emphasize.edt');"
ITEM="\textrm{...}"
    MACRO="Exe('%b\Macros\Fonts\Roman.edt');"
ITEM="\textbf{...}"
    MACRO="Exe('%b\Macros\Fonts\Bold.edt');"
ITEM="\textsf{...}"
    MACRO="Exe('%b\Macros\Fonts\SansSerif.edt');"
ITEM="\texttt{...}"
    MACRO="Exe('%b\Macros\Fonts\TypeWriter.edt');"
ITEM="\url{...}"
    MACRO="Exe('%b\Macros\Fonts\FormatURL.edt');"
ITEM="\inline{...}"
    MACRO="Exe('%b\Macros\Fonts\FormatInline.edt');"
ITEM="\index{...}"
    MACRO="Exe('%b\Macros\Fonts\FormatIndex.edt');"
ITEM="\textsup{...}"
    MACRO="Exe('%b\Macros\Fonts\Formattextsuperscript');"
ITEM="\textmd{...}"
    MACRO="Exe('%b\Macros\Fonts\Medium.edt');"
ITEM="\textit{...}"
    MACRO="Exe('%b\Macros\Fonts\Italic.edt');"
ITEM="\textsc{...}"
    MACRO="Exe('%b\Macros\Fonts\SmallCaps.edt');"
ITEM="\textsl{...}"
    MACRO="Exe('%b\Macros\Fonts\Slanted.edt');"
ITEM="\verb"..."
    MACRO="Exe('%b\Macros\Fonts\Verbatim.edt');"
ITEM="\TBC{...}"
    MACRO="Exe('%b\Macros\Fonts\FormatTBC.edt');"
ITEM="\scmd{...}"
    MACRO="Exe('%b\Macros\Fonts\FormatScmd.edt');"
ITEM="\nomen{...}"
    MACRO="Exe('%b\Macros\Fonts\FormatNomenclature.edt');"
ITEM="\textsub{...}"
    MACRO="Exe('%b\Macros\Fonts\Formattextsubscript.edt');"

```

---

If required, you can add even more new buttons to the current image, but it is somewhat tricky. Take care that you keep the new buttons the same size. Also add the code for the new buttons as shown above.

It seems that you specify the number of rows and columns in the GUI and also the width and height of each button (which do not quite match up to the image size). Change the number of columns and insert the new button's code in the appropriate location in the list (e.g. url and TBC above).

After you have edited the above ini file, save the file and load the changes as shown in Section 1.1.

Either copy the `edt` files from the repo folder `files/Macros/Fonts` to the installation folder `Macros/Fonts`. Alternatively reconstruct the files from the following text. Create the appropriate scripts, with the contents below, and save these in `Macros/Fonts/`:

```

--- old 555x88 96 ppi  new 999x88 72 ppi

125 old  520x56 168ppi new

200 old  800x88 168ppi new  1798x88 72 ppi

```

---

File FormatIndex.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
    'SelWord(1);>
    IfSel(''0'', '=',>
        ''Ins("\index{}");>
        CharLeft;'',>
        ''InsLabel("\index","{","}")'';','>
    'InsLabel("\index","{","}")'';');
EndGroup;
End;
```

---

## Filename FormatInline.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
    'SelWord(1);>
    IfSel(''0'', '=',>
        ''Ins("\lstinline{}");>
        CharLeft;'',>
        ''InsLabel("\lstinline","{","}")'';','>
    'InsLabel("\lstinline","{","}")'';');
EndGroup;
End;
```

---

## Filename FormatNomenclature.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
    'SelWord(1);>
    IfSel(''0'', '=',>
        ''Ins("\nomenclature{{{}}");>
        CharLeft;'',>
        ''InsLabel("\nomenclature","{",""}{{{}}")'';','>
    'InsLabel("\nomenclature","{",""}{{{}}")'';');
EndGroup;
End;
```

---

## Filename FormatScmnd.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
    'SelWord(1);>
    IfSel(''0'', '=',>
        ''Ins("\scmnd{}");>
        CharLeft;'',>
        ''InsLabel("\scmnd","{",""}")'';','>
    'InsLabel("\scmnd","{",""}")'';');
EndGroup;
End;
```

---

## File FormatTBC.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
  'SelWord(1);>
  IfSel(''0'', ''='',>
    ''Ins("\TBC{}");>
    CharLeft;'',>
    ''InsLabel("\TBC","{","}")'';','>
  'InsLabel("\TBC","{","}")'';');
EndGroup;
End;
```

---

#### Filename Formattextsubscript.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
  'SelWord(1);>
  IfSel(''0'', ''='',>
    ''Ins("\textsubscript{}");>
    CharLeft;'',>
    ''InsLabel("\textsubscript","{","}")'';','>
  'InsLabel("\textsubscript","{","}")'';');
EndGroup;
End;
```

---

#### Filename Formattextsuperscript.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
  'SelWord(1);>
  IfSel(''0'', ''='',>
    ''Ins("\textsuperscript{}");>
    CharLeft;'',>
    ''InsLabel("\textsuperscript","{","}")'';','>
  'InsLabel("\textsuperscript","{","}")'';');
EndGroup;
End;
```

---

#### Filename FormatURL.edt

```
// -*- ASCII:EDT -*-

BeginGroup;
IfSel('0','=',>
  'SelWord(1);>
  IfSel(''0'', ''='',>
    ''Ins("\url{}");>
    CharLeft;'',>
    ''InsLabel("\url","{","}")'';','>
  'InsLabel("\url","{","}")'';');
EndGroup;
End;
```

---

If you are using nomenclature download and install the following macro:  
<http://www.winedt.org/config/menus/Nomenclature.html>

## 1.13 Adding New Environments to Insert Menu

You may want to add new menu items to the existing menu system.

<Options><Options Interface><Menu and Toolbar...><Main Menu>, edit the macro file just opened. Locate this section corresponding to the submenu for environment insertions:

```
SUBMENU="Environments>"
  CAPTION="&Environments"
  CONFIG_FILTER="Default;MiKTeX;TeX Live"
  IMAGE="Env"
  REQ_DOCUMENT=1
```

---

and insert the code to create new listings at the end, just before `END="Environments>"`

```
ITEM="-"
ITEM="lstlisting"
  CAPTION="&lstlisting"
  MACRO="LetReg(9, 'lstlisting');LetReg(8);Exe('%b\Menus\Insert\Env.edt');"
  REQ_DOCUMENT=1
ITEM="-"
```

---

After you have edited the above ini file, save the file and load the changes as shown in Section 1.1. Then copy the file `env.tab` from the `repo files` folder to `Menus/Insert`

## 1.14 Change the Default Insert Templates

WinEdt keeps its templates for inserted text in the following directory:

`C:\Program Files\WinEdt Team\WinEdt 10\Templates\LaTeX.`

Modify any of the templates in this directory to suit your needs. The text insertion is available under the 'Insert' menu entry.

## 1.15 Add a Macro to Activate Jabref

If you are using Jabref:

Using the same procedure as described in Section 1.8, create a new menu entry and paste the following code into the 'Macro' box (all on one line):

```
[IfisMode('BibTeX', '%!m', 'LetReg(0,"%p%n%t");', 'LetReg(0,"");');
Run('"javaw" -jar "C:\Program Files\JabRef 2.0.1\JabRef.jar" "%!0"', '%p');]
```

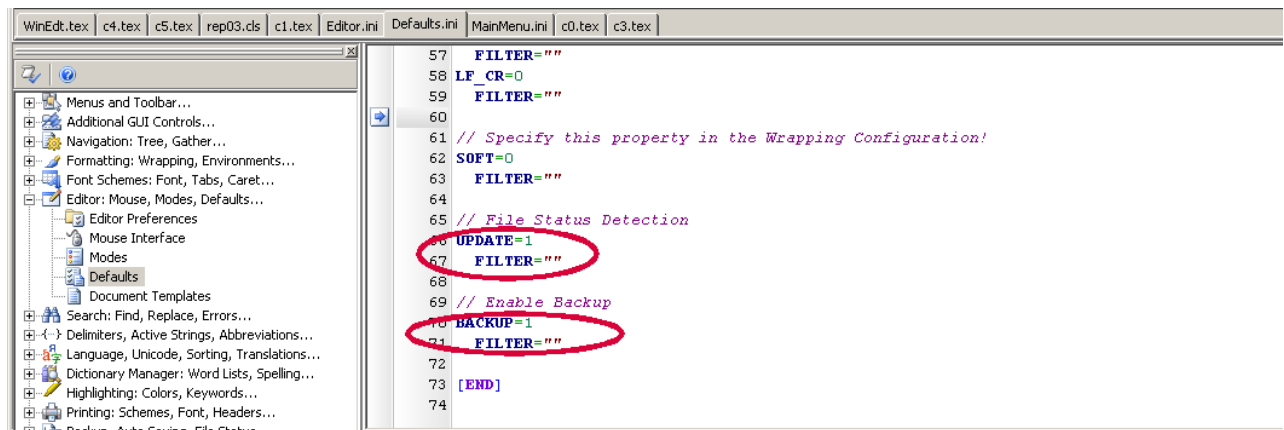
where the reference to Jabref must be to the jar file, as installed on your PC.

```
ITEM="Jabref"
  CAPTION="Jabref..."
  IMAGE="Table"
  MACRO="Exe('IfisMode('BibTeX', '%!m', 'LetReg(0,"%p%n%t");', 'LetReg(0,"");');
Run('"javaw" -jar "C:\Program Files (x86)\JabRef\JabRef-2.10.jar" "%!0"', '%p');')"
  REQ_DOCUMENT=1
  REQ_FILE=0
```

## 1.16 Enable File Update Warnings

Sometimes you update a file externally (e.g. subversion updates the file version) and you want to be warned of this event.

<Options><Options Interface><Editor: Mouse, Modes.....><Defaults>, edit the macro file just opened. Confirm that the values are set to 1 as shown below at the indicated location.



## 1.17 Enable Multiple Instances of WinEdt

<Options><Options Interface><Application: Projects, Forms, ....><Additional Preferences>, near line 62 set the value to 0:

```
// Rather than disabling this option use -C command switch!
RUN_ONE_INSTANCE_ONLY=1
```

Or as suggested, leave this at 1 and just open an instance of WinEdt on the commandline with the following:

```
WinEdt -C="window title"
```

## 1.18 Colour Theme

To set a colour theme select from the <Options><Theme>< menu. If this does not work, experiment with

<Options><Options Interface><Highlighting, Colors ...><Colors>. Scroll to Solarized Light and uncomment the theme you require, and comment out all those not required. Now it is a bit of a hit-and-miss using the menu options and selecting Default from an empty list, to get the scheme to work.

## 1.19 See also

<http://nakhmani.wordpress.com/2010/09/28/winedt-6-configuration/#more-44>





## Chapter 2

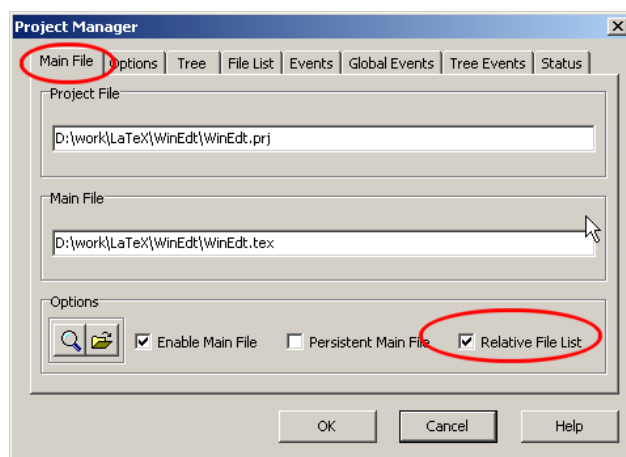
# Setting up WinEdt Projects

### 2.1 Setting up the Main File

This section assumes that your project consists of more than one file (e.g. each chapter in a new file). This technique requires one file as a root file, calling up all the other files.

Load the root file into the editor, and set it as the main file by using the menu path <Project><Set main file>. This would set the current file as the root or main file for the project. This is the file that will be compiled when the TeX or LaTeX buttons are pressed.

To ensure that the included file locations are saved as relative to the main file, open the 'Project Manager' using the menu path <Project><Project Manager>. Ensure that the relative file list check box is checked. On previous version of WinEdt this relative check box appeared to be checked, but the relative search did not work. To fix this, first deselect the checkbox and 'OK' the dialog box, then reopen it again and select the relative file list check box. This seemed to have fixed the problem in the past.





## Chapter 3

# Setting DVIPS

### 3.1 DVIPS settings

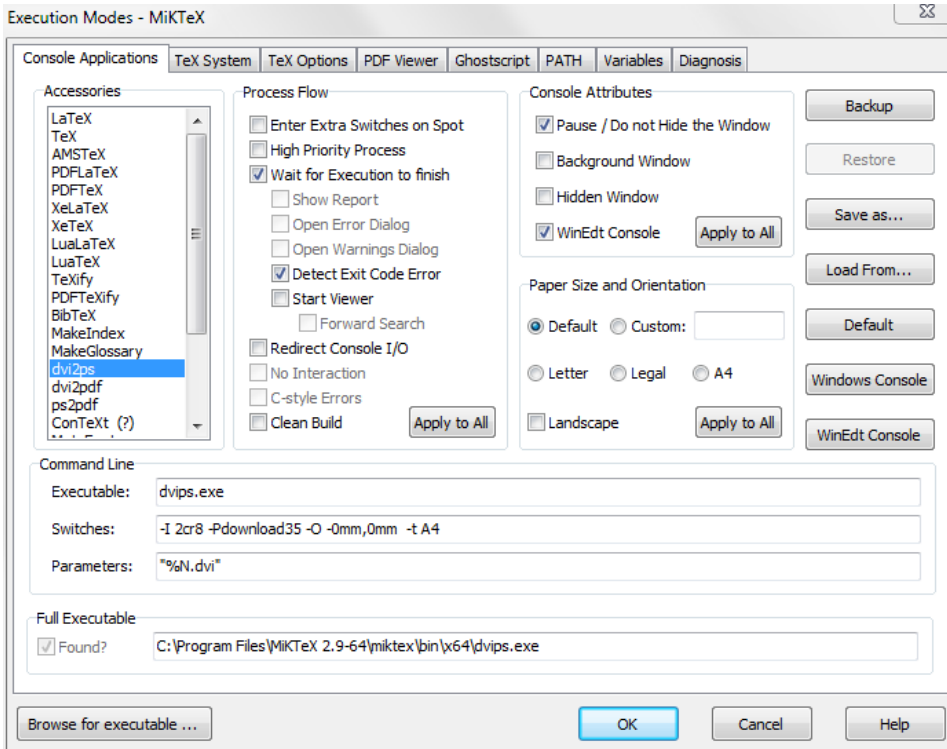
If you are using DVIPS: do the following.

#### Enable Colour PNG in DVIPS

DVIPS converts  $\text{\TeX}$  DVI files into PostScript files.

DVIPS only supports EPS files as input graphics. If you use DVIPS and you want to use PNG files, you need to use bmeps to convert the PNG to an EPS first. Fortunately bmeps is included in the MikTeX DVIPS (a so-called 'bmeps enabled DVIPS' version). Unfortunately the default WinEdt mode only converts the files to a monochrome EPS file, destroying all colour information. The switch to convert PNG files to EPS format in colour must be set manually.

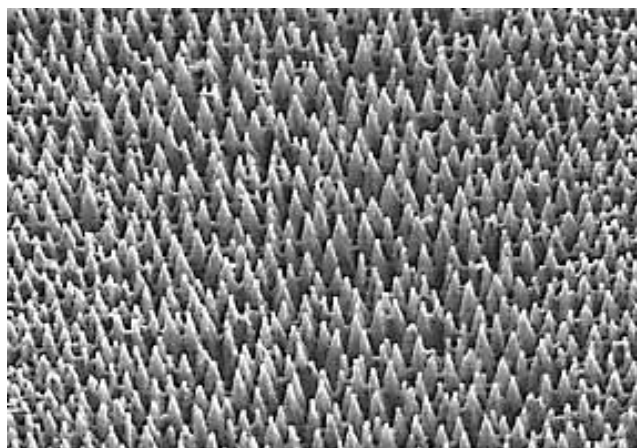
Activate <Options><Execution Modes> and select the `dvi2ps` entry in the 'Accessories' list box. Then enter the switch `-I 2cr8` in the 'Switches' textbox, as shown in the following figure. The bmeps FAQ is located at <http://bmeps.sourceforge.net/faq.html>



## Enable JPG Images in DVIPS

bm2eps also converts jpeg files, just use it as follows:

```
\centerline{\includegraphics[bb= 0 0 288 200,width=0.5\textwidth ]{pic/light_trap2.jpg}}
```



## Control Font Download to the PS File

In the previous graph the dvi2ps application is also instructed to download the ‘standard Adobe’ fonts to the PS file, by entering the option

-Pdownload35

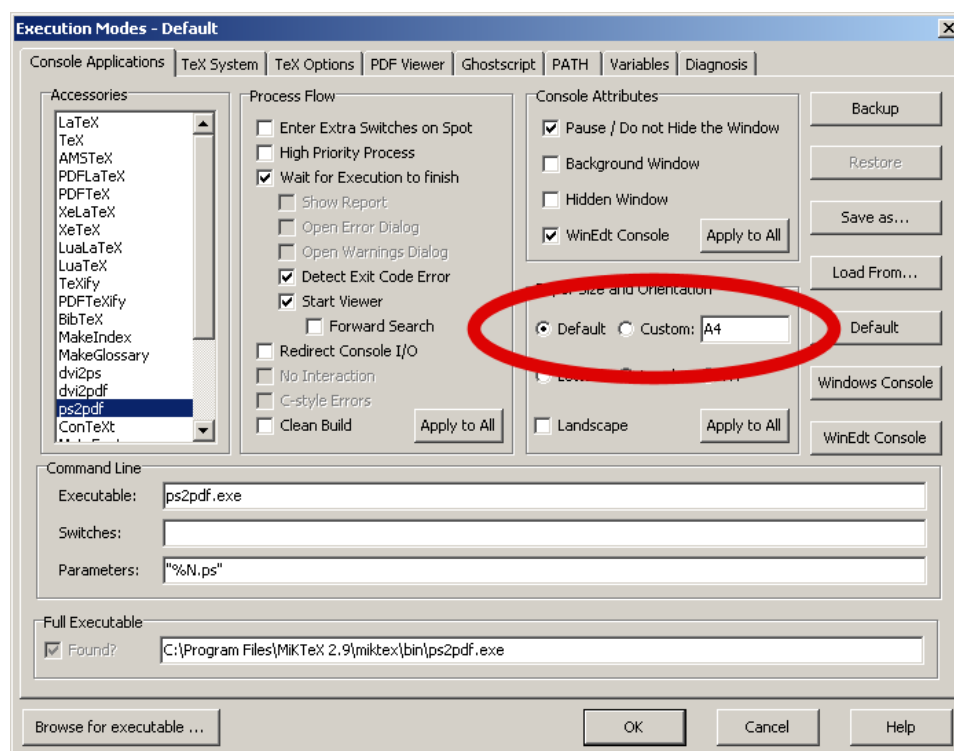
### Setting up the Page Size for dvi2ps

To set the page size in dvi2ps, add the following additional text to the dvi2ps switches:

```
-I 2cr8 -Pdownload35 -O -0mm,0mm -t A4
```

### Setting up the Page Size for ps2pdf

Activate <Options><Execution Modes> and select the ps2pdf entry in the 'Accessories' list box. Then change the page size as shown in the next figure. It does not make sense that the A4 should be in effect when the page size is set to default, but it does seem to work.



## 3.2 PostScript Specials and Security

The issue discussed here became evident in MikTeX 2.5, but could also apply to any other application that uses DVIPS. The problem presented itself as an inability of YAP to preview some DVI pages.

As from Version 5.95b DVIPS does not support 'unsecure' path names, such as

- absolute path names (like C:/temp/eurotour.eps)
- parent-relative path names (like ../../temp/eurotour.eps)

which means that the following will not work:

```
\includegraphics[bb= 0 0 512 387,scale=0.7]{eps/projectmanager.png}
```

YAP complains with a message similar to:

MiKTeX Problem Report

Message: The page could not be rendered.

Data: This is dvips(k) 5.95b Copyright 2005 Radical Eye Software (www.radicaleye.com)

' TeX output 2006.05.28:2135' ->

<tex.pro><texps.pro><special.pro>. <cmbx12.pfb><cmr10.pfb>[2<eurotour.eps>

C:\Program Files\MiKTeX 2.5\miktex\bin\dvips.exe:

Could not find figure file c:/temp/eurotour.eps; continuing

On Cristian Schenk's page he states:

([http://dojo.miktex.org/blogs/christian\\_schenk/archive/2006/03/06/328.aspx](http://dojo.miktex.org/blogs/christian_schenk/archive/2006/03/06/328.aspx))

It would be possible to break these security rules by

- using the Dvips option -R0
- by specifying z0 in the Dvips configuration file

Schenk offered to implement the first option with a future release of YAP, but as of version 2.7 is still is not implemented. Our recourse is then to implement the second option ourselves.

On my PC, the DVIPS config file, config.ps, is located at the following location:

C:\Program Files\MiKTeX 2.7\dvips\config

In this file, find these lines:

```
% z1 is "secure", i.e., inhibits execution of `shell commands` in
% \specials. Dvips allows this by default.
z1
```

and change it to this:

```
% z1 is "secure", i.e., inhibits execution of `shell commands` in
% \specials. Dvips allows this by default.
z0
```

At the top of the config file it instructs us to use initexmf to change this file, but I could not find an easy way to do this, so I just manually edited config.ps. It seems to work.

## Chapter 4

# Installing LZMA-Archive Packages on MikTeX

### 4.1 Problem

MikTeX uses the LZMA compression tool to compress tar package files, such as `fancyhdr`. For some reason my implementation of MikTeX 2.5 and 2.7 are unable to read its own compressed files (e.g. `fancyhdr.tar.lzma`). This means that the package manager is unable to install such files.

### 4.2 Solution

On <http://comments.gmane.org/gmane.comp.tex.miktex/6578> Christian Schenk writes:

Re: Format of new LZMA archives

```
> just a technical question: Is there something special about the format
> of the new .lzma archives? When I try to unpack those files with 7z
> (v4.47 beta) it gives me an error (Can not open file as archive).
```

The `.tar.lzma` files in the MikTeX package repository were created with `lzma.exe` from the LZMA SDK (<http://www.7-zip.org/sdk.html>). You can run

```
lzma d PACKAGE.tar.lzma -so | tar -xvf
```

to extract files from a package.

In order to install packages archived with LZMA I had to resort to the following procedure:

1. Download the LZMA SDK as Schenk advises from <http://www.7-zip.org/sdk.html>
2. Rename `PACKAGE.tar.lzma` to `PACKAGE.lzma`, where `PACKAGE` is the package name.
3. Unzip the archive to a tar file with the following command

```
lzma d PACKAGE.lzma PACKAGE.tar
```
4. Untar the tar-ball with Winzip or whatever tool you are using, retaining the directory structure.
5. Copy the directory structure to the MikTeX directory (e.g. `C:\Program Files\MikTeX 2.7`). Take care to copy the untarred files to the appropriate directories in the MikTeX directory structure.





## Chapter 5

# Setting ps2pdf

Open the file `C:\Program Files\WinEdt Team\WinEdt\Exec\TeX\ps2pdf.edt` and do the following change

```
//LetReg(4, '');  
LetReg(4, "%!4 -sPAPERSIZE=a4");
```