

WinEdt

Quick Guide

ALEKSANDER SIMONIČ

© 1993–2023

WinEdt Version 11.1 (2023-02-15)
Typeset with L^AT_EX 2_ε and KOMA-Script class
using Palatino and Bera fonts.
generated: 2023-02-06

Contents

List of Figures	v
Acknowledgements	vii
1 Installing WinEdt and T_EX	1
1.1 Installing WinEdt is easy	1
1.2 What about T _E X?	2
1.3 What Next?	3
1.4 Upgrading	4
2 Getting Started	5
2.1 Setting up a T _E X project in WinEdt	6
2.2 Compile and Preview	7
2.3 PDF Viewer and inverse search	8
2.4 Toolbar Alternatives	9
2.5 T _E X Symbols GUI	10
2.6 T _E X AUX Output Folder	10
2.7 Execution Modes	11
2.8 WinEdt Help System	12
3 WinEdt and Unicode (UTF-8) encoding	13
3.1 Code Page Converter	14
3.2 T _E X and International Characters (UTF-8)	15
3.3 Translation Tables	16
3.4 T _E X and UTF-8	17
3.5 Modes and Submodes	18
3.6 Spell Checking	19
3.7 WinEdt Dictionaries (Word Lists)	20
4 L^AT_EX Demo: Non-WinEdt-Related Bonus	21
4.1 Graphics Inclusion	21
4.2 Color Package Example	23
4.3 Rotated objects examples	24
4.4 Pgfplots: Tikzpicture Example	25
4.5 Landscape mode page example	26

4.6	Another landscape page example	27
4.7	Presentations in L ^A T _E X	28
4.8	L ^A T _E X and paper size	28
4.9	A Simple Revision Control System (RCS)	29
4.10	Useful T _E X-ing Hints	31
4.11	In memoriam Echo (2005-2016)	33

Bibliography		35
---------------------	--	-----------

List of Figures

1	TeX Lion: ready and willing.	viii
1.1	Execution Modes: TeX System	2
1.2	Preferences: Wrapping	3
1.3	About WinEdt Dialog	4
2.1	WinEdt running for the first time	5
2.2	Set and Remove Main File Buttons	6
2.3	WinEdt ready to compile	6
2.4	WinEdt after successful compilation	7
2.5	This document displayed in WinEdtPDF viewer	8
2.6	Toolbar Alternatives	9
2.7	TeX Symbols GUI (Default Light Theme)	10
2.8	TeX Symbols GUI (Dark Solarized Theme)	10
2.9	Execution Modes: TeX Options	11
3.1	Preferences: Unicode	13
3.2	Document Settings: Code Page Converter	14
3.3	Dictionary Manager Dialog	19
3.4	Spell-checking in action	20
4.1	WinEdt classic and new-style logo	21
4.2	SV Adriana docked	22
4.3	About WinEdt Dialog	22
4.4	LaTeX Logo	24
4.5	Rotated circles	24
4.6	Mathematica 3D-objects in landscape mode	27
4.7	Happy TeX-ing from TUG and the WinEdt Team	32
4.8	All Things Must Pass	33

Acknowledgements

Ralf Heckmann supplied the template for L^AT_EX and KOMA-Script package using Palatino and Bera fonts. This template is used to typeset this document.¹

Luca introduced and applied text styling commands, fixed a few inaccuracies, and improved a few T_EX examples.

Adriana proofread and edited the draft version. She fixed numerous language mistakes, cut out some fat, and slightly rearranged the text to improve the *flow* (for those that notice such things).

Here is where you will be acknowledged if you contribute a chapter (or a section) of general interest that will improve this manual and help other users (new and experienced) learn how to better use WinEdt. There is plenty of room for improvements and additional information but it will require the involvement of the WinEdt Community. This is your chance to do something about it!

-
1. L^AT_EX sources for this guide are available for download in a self-contained zipped archive. Unpack its contents in a directory of your choice, open the main document `QuickGuide.tex`, and set it as the main file in WinEdt's tree interface. Now you are ready to compile it with [PDFTeXify](#) button (or the universal compile shortcut [F9](#)). After the compilation is complete the resulting pdf document will be opened in your PDF Viewer (such as Adobe Reader or SumatraPDF). Please think about the environment before you decide to print the whole document...



Figure 1: TeX Lion: ready and willing...

1 Installing WinEdt and T_EX

1.1 Installing WinEdt is easy

Download and execute the setup file `winedt11.exe`.¹ As long as you've downloaded a legitimate copy of WinEdt (e.g. from our website www.winedt.com) you will receive the standard Windows User Account Control warning that displays WinEdt Inc. as the *verified publisher* of this product.

It is now safe to proceed with the installation. The Installer Wizard will guide you. For most users, the default options in the Wizard should be used. If you are creating a portable installation, however, disable User Profile Creation: this way WinEdt will create a `Local` folder inside its install folder and this folder will contain all your custom settings...

During the installation you can decide if you want this version of WinEdt associated with T_EX files (and friends).² Filetype associations can later be made, removed, or repaired (on a selective basis) through WinEdt's Options -> Configuration Wizard. However, on Windows with UAC-enabled, this will require starting WinEdt with elevated privileges (which are already in effect during the installation).

After the installation is completed start WinEdt from the shortcut on your desktop: now you are ready to go. If you are planning to use WinEdt regularly you should consider pinning it to your taskbar. This is done from the Windows popup menu that is displayed when you right-click on the WinEdt icon on your desktop. The same menu can also be used to unpin the program if you find your taskbar overloaded with applications.

WinEdt comes with its own pdf viewer based on the open-source PDFium library. It allows trouble-free previewing of pdf documents with support for Forward and Inverse Search to switch between sources and the compiled pdf document. This is further explained in the next chapter...

-
1. It is recommended that you download and install the latest version from our website. Ordering a CD is not necessary as no extras are included in it. If you need or want a physical copy you can save yourself some money by burning your own CD.
 2. This is the default and recommended for most users.

1.2 What about T_EX?

If you are planning to use WinEdt for writing L^AT_EX documents you must also install a T_EX System such as MiK_TE_X or T_EX Live; both T_EX Systems run well on Windows, the choice is yours.³ If you haven't done it yet this would be a good time! After the T_EX installation is completed restart WinEdt and it will automatically detect your T_EX System and connect with its accessories.

If in doubt, you can always start the Execution Modes dialog in WinEdt's Options menu and check the last page: Diagnosis. It explains everything. Help in this dialog offers a guide on how to manually fix any problems in a rare situation when automatic detection fails.

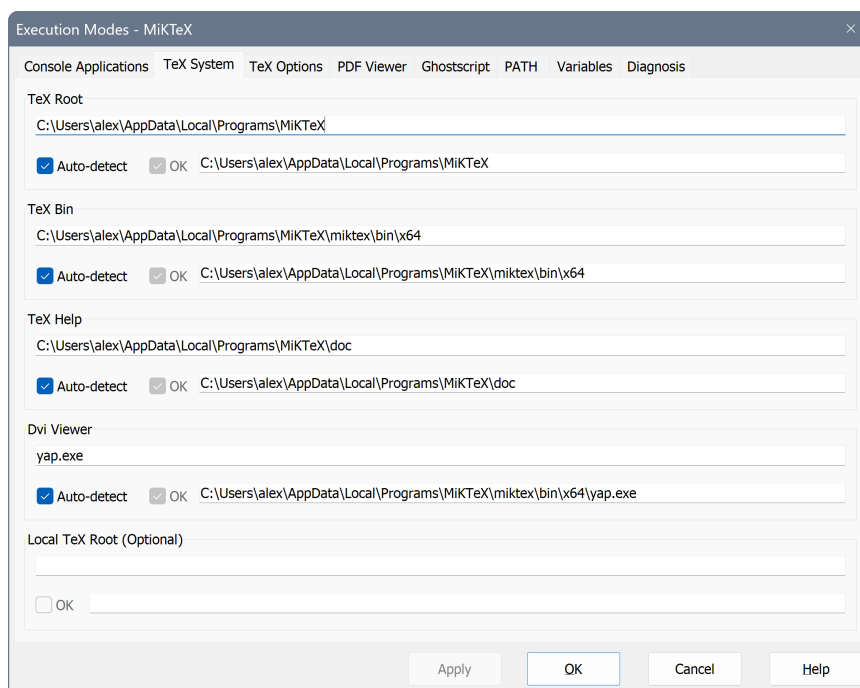


Figure 1.1: Execution Modes: T_EX System

For most users this is all that is required to make WinEdt and T_EX fully functional for creating and previewing the compiled pdf files on Windows!

3. MiK_TE_X download is considerably smaller and the installation is faster. MiK_TE_X Console (available through WinEdt's TeX Menu) will help you keep it current. However, MiK_TE_X may have to use its install-on-demand feature during a compilation to download and install extra packages that you may be using in your documents. This can make compilation slow until all missing packages have been installed. T_EX Live, on the other hand, comes loaded with just about every supported L^AT_EX package. This makes the download huge and the installation slower than MiK_TE_X. It also requires an annual update to keep its large package repository up-to-date. However, once it is installed it runs fast.

1.3 What Next?

If you have a properly installed T_EX system you are ready to proceed to the next chapter and see WinEdt and T_EX in action working on this document.

This is a good time to make some basic choices and configure WinEdt for your needs. The details depend on your intended use of WinEdt and the default settings cannot always please everyone.

The default settings use Notepad-like Soft Wrapping and UTF-8 encoding for T_EX Documents, as expected by most users. If these are not your preferences you should make some changes now before starting to work on your documents (this can save you from a lot of problems and confusion later)! The Options -> Preferences dialog offers some basic choices on your preferred Wrapping, Unicode, and Backup strategy. . .

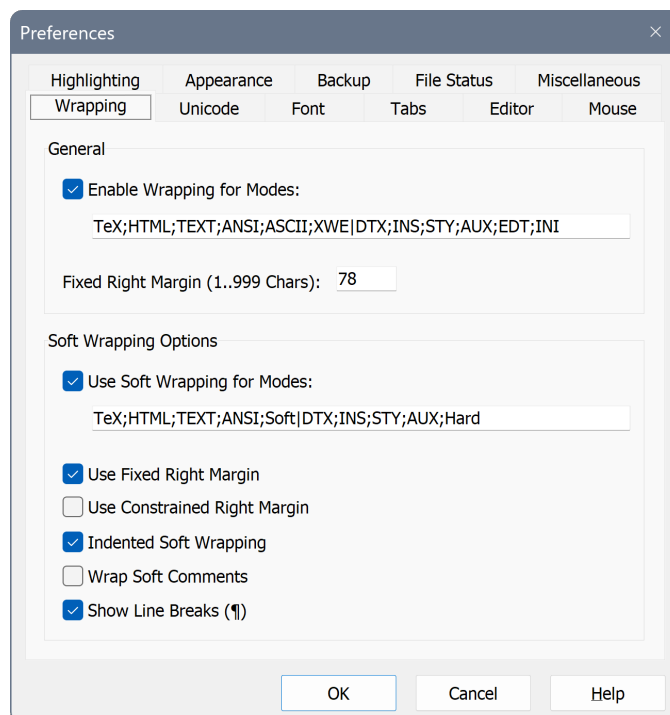


Figure 1.2: Preferences: Wrapping

Help in the dialog explains the meaning of all available options. . . Take your time and explore the possibilities!

1.4 Upgrading

Upgrading WinEdt 11 series is easy!⁴

The Help menu contains an item [Check for Updates](#). If a newer build of WinEdt is available you will be directed to WinEdt's Downloads page to download and install the latest version (using your browser).

Here are the steps that you should perform for smooth upgrading:

- Install the new version of WinEdt 11 (over the old instance of WinEdt 11).
- Execute Options -> Maintenance -> [Upgrade Local...](#) menu command (to merge your settings with the new defaults).
- Restart WinEdt and you are ready to work on your projects...

WinEdt's "About Dialog" displays the version and build of your instance of WinEdt.

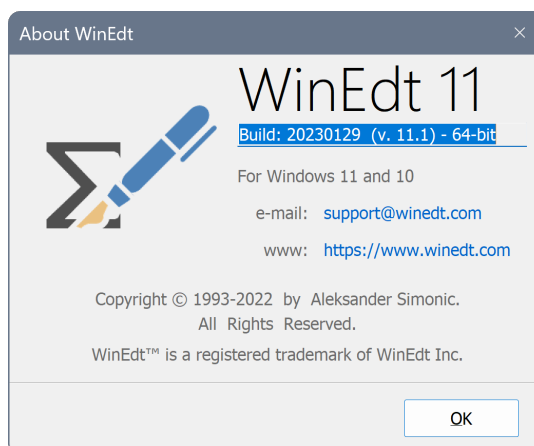


Figure 1.3: About WinEdt Dialog

The "What's New?" section in WinEdt's Manual explains what has been added, changed, or fixed for the latest build. Check it out!

4. You don't have to uninstall the previous build of WinEdt 11 to upgrade to the latest version. You should *not* uninstall it if you want to preserve custom settings in your Application Data or Local folder.

2 Getting Started

So you've downloaded and installed WinEdt and started it for the first time. This is what you see:

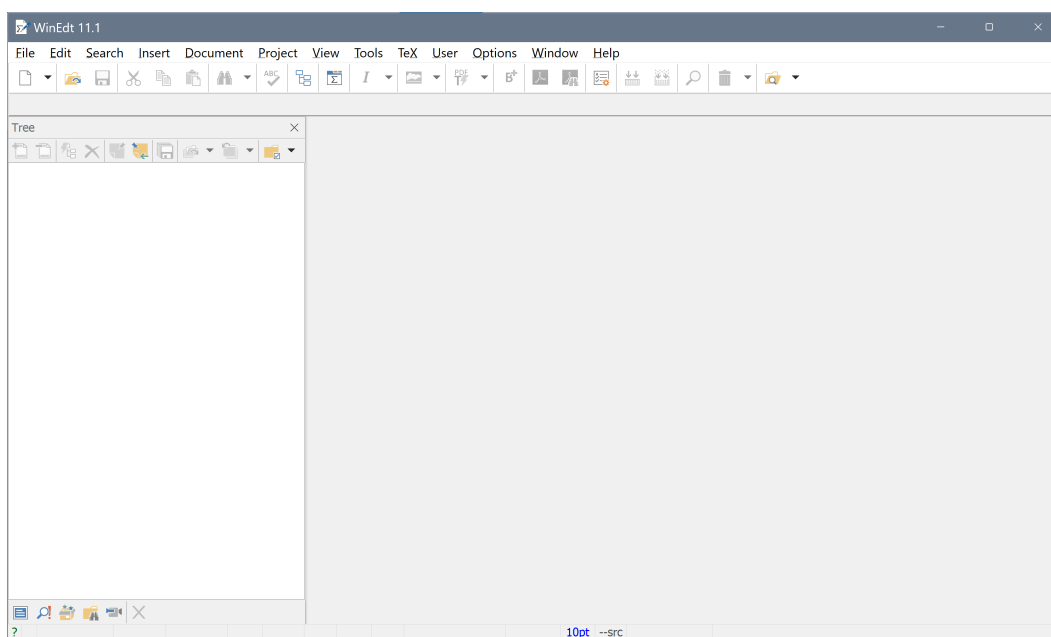


Figure 2.1: WinEdt running for the first time

Not particularly impressive or useful as is. On the left is an empty tree control that will later be used to display the project structure, Table of Contents, and other items collected in the relevant documents. On top is the default 1-row toolbar with many buttons disabled (grayed out). There's more to say about the toolbar interface, but for now, let us point out that while many such buttons are currently disabled that does not mean there's a bug or indicate something's wrong with your WinEdt/TeX installation. That the TeX compiler, converters, and previewer buttons are disabled simply indicates that currently there is nothing to compile, convert, or preview.¹

1. If at this stage you have any doubt about what TeX System and other accessories WinEdt detected on your computer, start the Options -> Execution Modes dialog and check the last, Diagnosis, tab page. The report produced there will clearly indicate whether you forgot to install a TeX system such as MiKTeX or TeX Live (in which case, now is the time to do it!). . .

2.1 Setting up a T_EX project in WinEdt

Now we open the main file used to create this document: `QuickGuide.tex`. Use the button on the left of the toolbar in the tree interface to set this document as the main file for this WinEdt T_EX project.



Figure 2.2: Set and Remove Main File Buttons

With the main file set, WinEdt builds the project tree with all included chapters and displays the TOC branch for this document.² This main file consists of the preamble required for T_EX-ing the entire project and at the end, it includes all chapters. Use the TOC to open any chapter with actual contents: once the main file has been set, WinEdt knows to use that file for compilers and previewers even if you are currently working on a subdocument (or even on an unrelated file).³

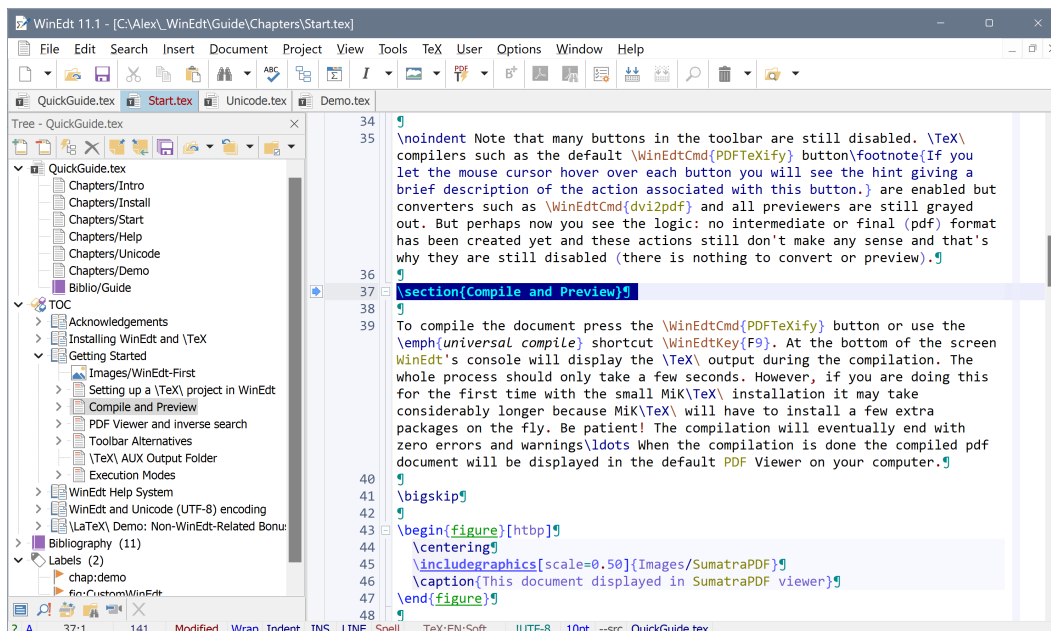


Figure 2.3: WinEdt ready to compile

2. Also, the name of the main document now shows in the last panel in the status line.
3. Note that many buttons in the toolbar are still disabled. T_EX compilers such as the default PDFTeXify button are enabled but BibTeX/ Biber and pdf Viewer are still grayed out. But perhaps now you see the logic: no file has been compiled and these actions still don't make any sense and that's why they are still disabled (there is nothing to preview).

2.2 Compile and Preview

To compile the document press the `PDFTeXify` button⁴ or use the *universal compile* shortcut `F9`. At the bottom of the screen WinEdt's console⁵ will display the \TeX output during the compilation. The whole process should only take about ten seconds. However, if you are doing this for the first time it may take considerably longer because $\text{MiK}\TeX$ will have to install a few extra packages on the fly. Furthermore, \LaTeX and $\text{Bib}\TeX$ have to be run three times to get the bibliography and TOC in order. Be patient! The compilation will eventually end and the pdf file will be produced and displayed in WinEdtPDF viewer...

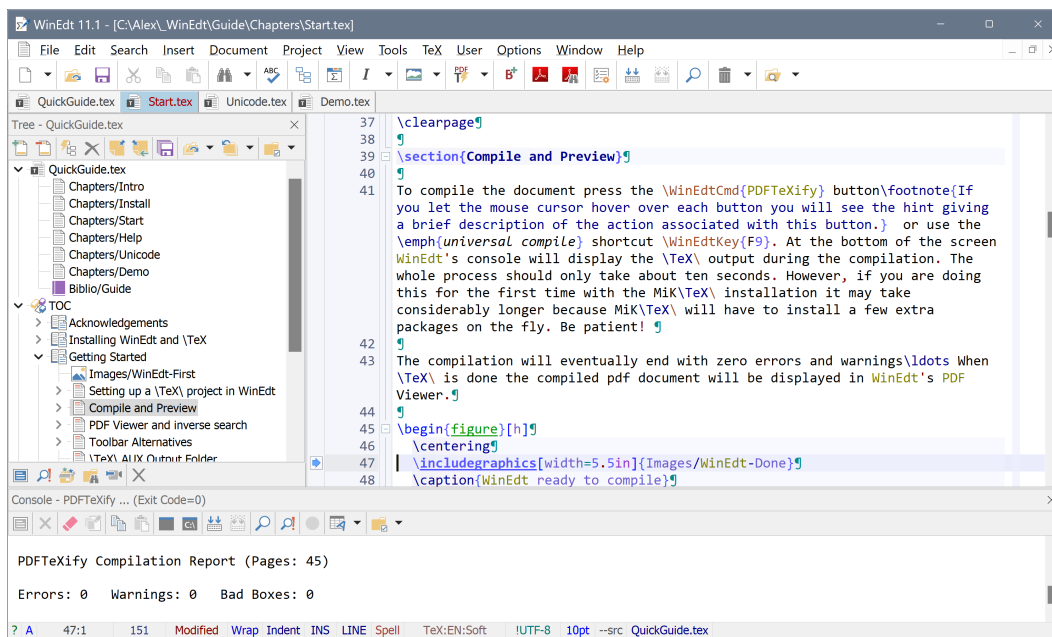


Figure 2.4: WinEdt after successful compilation

WinEdt's Console displays the output produced by the compiler (console application) that is being executed in it. It also allows you to provide input for the compiler when it requires user interaction. In particular, if \TeX encounters an error (like an undefined control sequence) it prompts you for action. You can decide if you want the compilation to continue (e.g by pressing `q` to enter "quiet" mode) or if you want it to stop and place you in the source (by pressing `e`). If you enter a question mark `?` followed by `Enter` \TeX displays the list of all admissible interactive commands with a short description.

4. If you let the mouse cursor hover over each button you will see the hint giving a brief description of the action associated with this button.
5. The console can be hidden or shown by using `Shift+Ctrl+C`. The Execution Modes Dialog can be used to determine if the Console is automatically hidden after the compilation (and more)...

2.3 PDF Viewer and inverse search

When \TeX is finished the compiled pdf document will be displayed in WinEdt's PDF Viewer. WinEdtPDF⁶ is a light-weight PDF Viewer that does not lock the pdf files it is previewing: it automatically refreshes their contents when they are recompiled, and it supports forward and inverse search⁷ as described below.

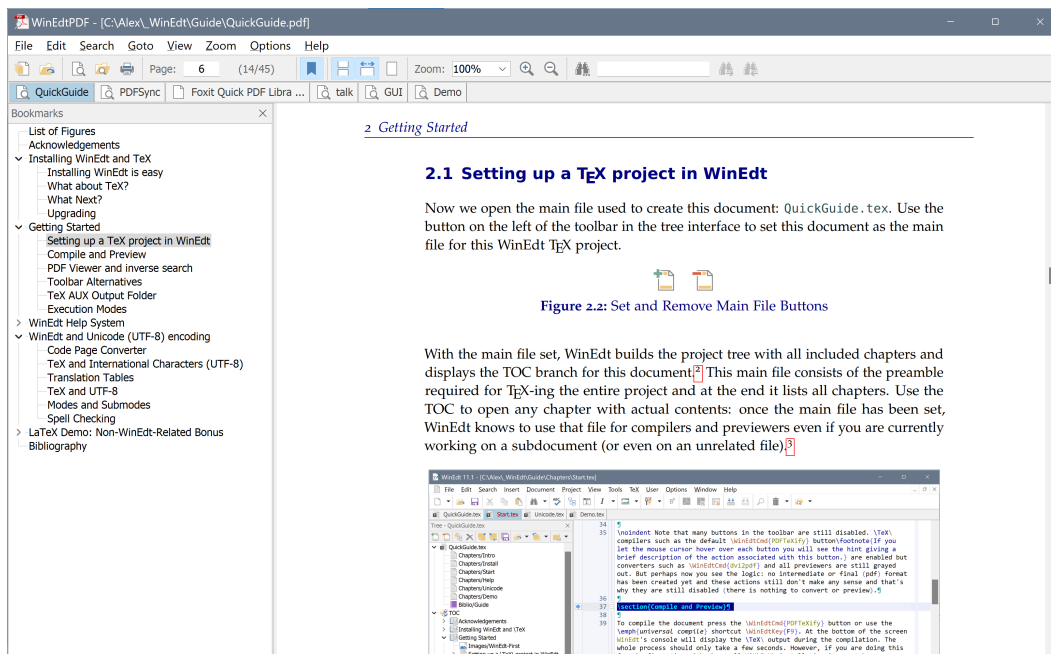


Figure 2.5: This document displayed in WinEdtPDF viewer

In WinEdtPDF you can double-click anywhere in the document and WinEdt will display the corresponding source. This is called *Inverse Search* and is implemented in the PDF Viewer. *Forward Search* is started in the editor and results in the PDF Viewer displaying the compiled text corresponding to the source position. In WinEdt you have a button and shortcut to initiate this action. It can also be done by clicking on the current line bookmark in the left margin (or double-clicking on the left margin in front of the desired line). If you are not already familiar with this concept try it: it is easy and it works very well!

Should you prefer to use some other PDF Viewer you can make your choices in the PDF Viewer Tab Page of the Execution Modes Dialog. Help in the dialog explains the details.

6. WinEdtPDF comes with its manual where you can find more information on how to use this pdf viewer with WinEdt and \TeX .
7. This functionality is based on the syntex technology and it is superior to the older src specials used by DVI Viewers such as YAP.

2.4 Toolbar Alternatives

Notice that many items in the TeX menu (or toolbar buttons) are still disabled: converters like [DVIPS](#), [dvi2pdf](#), [ps2pdf](#) and also [Dvi Preview](#) or [GSView](#) are still greyed out. This makes sense since we did not create any intermediate (legacy) output formats such as dvi or ps! Those who are used to such formats have their reasons⁸ for sticking to them but creating pdf output with one of the PDF engines is all the \TeX -ing most of us will ever need. With vertical space being a precious commodity on most wide screens we don't need or want a two-row toolbar where one-row will do just as well.

Many users will use [F9](#) for compiling. And, as pointed out above, forward PDF search from WinEdt 11 can be initiated by clicking on the current line bookmark in the left margin (or double-clicking on the left margin in front of the desired line).⁹ This practically eliminates the need for the Forward [PDF Search](#) button in the toolbar. It *can* be useful to show whether this command is currently enabled – e.g. the existence of pdf and synctex files, and many of us use such buttons for feedback like this rather than actually clicking on them regularly: another argument for a smaller 1-row toolbar! The Options -> Toolbar menu contains a few predefined (1 or 2-row) toolbar alternatives:

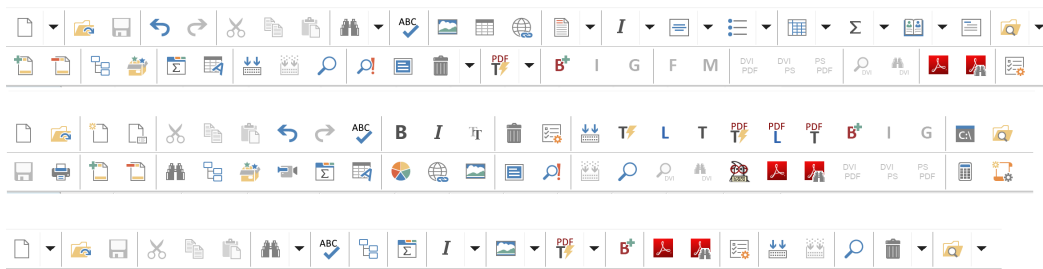


Figure 2.6: Toolbar Alternatives

Try them and see which one best fits your needs.¹⁰ Toolbar can be further customized by adding or removing a particular button. In WinEdt's Help -> Index, type the keyword [Toolbar Alternatives](#) and you will learn how (it is easier than you may think!). Adding or removing a comment in front of predefined buttons or rearranging a few lines in `Toolbar.ini` is usually all it takes.

8. See [Chapter 4](#) to learn how to import images or control paper size in \LaTeX without the need for intermediate formats. For example, most users do not have to install Ghostscript or GSView unless they want to work with the postscript output format. Beware though: this document cannot be compiled into dvi or ps because we did not provide any legacy eps version of graphics. . .

9. This is customizable by "event handlers" but for most users, the default settings will be enough.

10. If your toolbar looks very different and it lacks buttons related to TeX you probably choose one of the alternative (non-TeX) configurations from the Options Menu. Choose the Default one to let WinEdt auto-detect your TeX System.

2.5 T_EX Symbols GUI

In addition to the standard toolbar WinEdt also provides a (dark¹¹ theme-sensitive) graphic interface to almost 500 TeX symbols and commands:

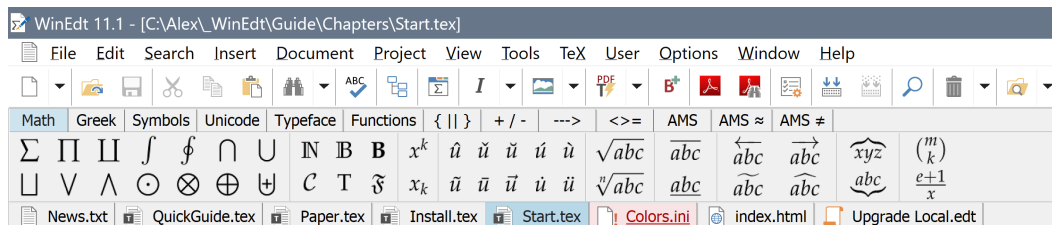


Figure 2.7: TeX Symbols GUI (Default Light Theme)

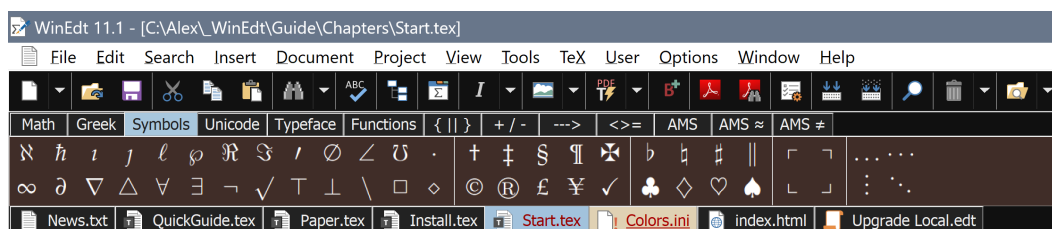


Figure 2.8: TeX Symbols GUI (Dark Solarized Theme)

You can use the View menu item or the associated toolbar button to toggle the visibility of the TeX Symbols GUI Page Control.

2.6 T_EX AUX Output Folder

If you take a look at the folder where you unpacked this project you may be pleasantly surprised to find that *no* T_EX auxiliary and numerous temporary files are polluting the source or the Chapters folders. Your source folders are junk-free!

All T_EX output files are stored in a TeXAux directory that has been created for this purpose and enabled in the Execution Modes Dialog. As a result, only your sources and compiled pdf and synctex¹² files are placed next to the main file: *nice and neat!*¹³

11. Dark mode themes have been significantly enhanced in WinEdt 11.1!

12. This file is there only to allow you to use forward and inverse search and it is not a part of your compiled pdf document. . .

13. For the more discerning, this feature can be further configured in the TeX Options page of Execution Modes. As always, clicking on Help in that dialog will explain the details and limitations of this functionality.

2.7 Execution Modes

The settings and definitions in the Execution Modes dialog govern the behavior of compilers and accessories (T_EX and friends) associated with WinEdt.

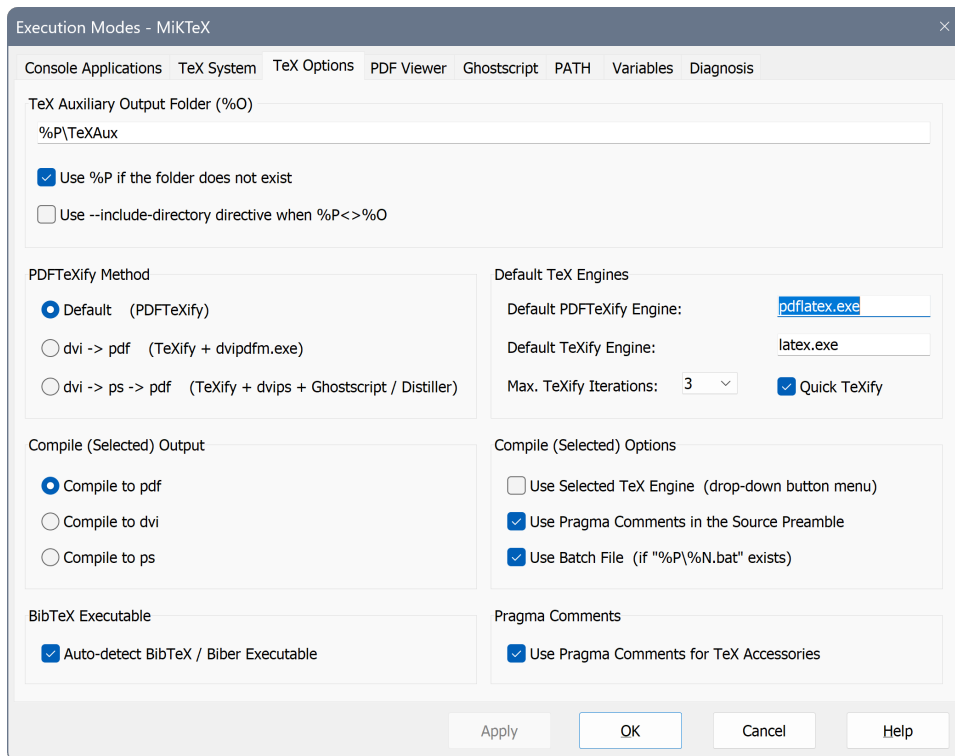


Figure 2.9: Execution Modes: T_EX Options

The Execution Modes dialog offers many possibilities that you may have not been aware of. You are urged to become familiar with this dialog and take advantage of what can be done, including checking that your T_EX system is properly installed, and much more. . .

Help in the Execution Modes dialog is thorough. It explains how WinEdt interacts with external accessories¹⁴ and what available alternatives¹⁵ can be easily incorporated in your custom version. If you are new to T_EX-ing or WinEdt you will greatly benefit from reading the dozen (or so) illustrated pages of Help associated with this dialog. . .

14. Including and especially your T_EX System and friends.

15. Including PDF Viewers, pdf converters, etc. . .

2.8 WinEdt Help System

As already heavily hinted, WinEdt comes with extensive, detailed, and up-to-date documentation in its HTML-Help manual. It is easily opened by pressing **F1**, and here you will find the description of all available WinEdt options and macro functions. Help is also indexed to allow quick access to certain topics that might be of interest to you. You should find it worthwhile to spend a few minutes investigating what's there and how it's organized. Most Dialogs have a Help button that can be used to quickly access the context-sensitive information on the relevant options or additional guidance on how to use the interface.

However, with the possible exception of a few sections in the *WinEdt Manual*, the help system was not written as a gentle introduction for first-time users. You don't have to study WinEdt's documentation to start working on your \TeX documents. The documentation is there to be consulted if and when you need help with some (advanced) editing task.

The *Configuration Manual* explains how to use the Options Interface to perform more advanced customizations. It lists all configuration sections and describes their properties. You should read the preface and possibly the Introduction to the Configuration Manual before you attempt non-trivial customizations through the Options Interface.¹⁶

The most commonly required customizations - such as wrapping, font, etc. - can be done through the Options -> Preferences dialog. However, advanced customizations - such as adding menu items, and making changes to the toolbar - have to be done through the Options Interface, for which the Configuration Manual should be consulted.

Additional information for more demanding and advanced users is available through the *Macro Manual*. The manual explains the syntax of WinEdt's macro language and describes all available functions, together with their parameters. The topics in the Macro Manual also cover some other advanced issues, most notably a chapter on *Regular Expressions* (with plenty of RegEx search and replace examples).

An in-depth understanding of WinEdt's macro language is not required to add extra accessories or slightly modify the existing behavior. Numerous examples in this manual explain how. Comments in the configuration and macro files provide additional guidance.

16. Many users are initially intimidated by the numerous options and possibilities and are afraid of breaking something. However, if you approach it with an open mind you will soon realize that this is a very good way to manage your customizations even if you are not a WinEdt expert. Should something go wrong the Options Menu -> Maintenance menu has a command **Rebuild Default** which will restore the default settings; you can then return and fix any mistake and then use **Rebuild Local** settings.

3 WinEdt and Unicode (UTF-8) encoding

WinEdt is a Unicode editor with support for UTF-8 or codes page-specific encoding. UTF-8 is the default format for \TeX documents. This can be configured through the Unicode section of the Options Interface - or through the Unicode page in the Preferences dialog (Help explains the details):

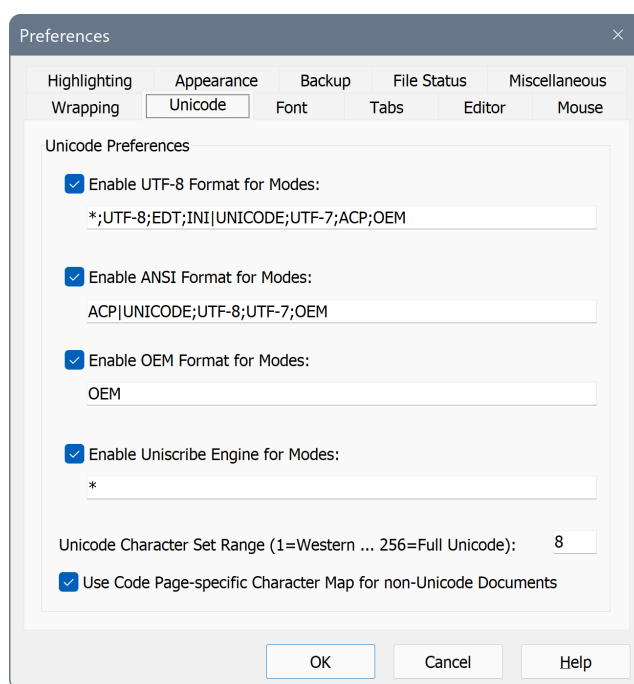


Figure 3.1: Preferences: Unicode

UTF-8 is the best choice for \TeX document encoding. However, if you have to work with legacy documents that were created in your default code page-specific format WinEdt will treat such documents properly and will preserve their encoding. Users that have to deal with code pages that are not native to their version of Windows will have to use the CP converter to tell WinEdt how to treat such documents. WinEdt is capable of handling both Unicode and (legacy) code page-specific documents.

3.1 Code Page Converter

If a document's mode ends with the submode :CPnum then the indicated code page is used to load the file in Unicode format. For example TeX:CP1251 uses a Cyrillic code page.

A comment at the beginning of a TeX document:

```
% !Mode:: "TeX:UTF-8"
```

will ensure that a document is properly loaded and saved. A similar convention is used by emacs:

```
% -*-coding: utf-8 -*-
```

WinEdt understands emacs coding directive for UTF-8.

Some Unicode or UTF-8 documents start with a Byte Order Mark (BOM). Unicode-aware applications can determine the coding of a document from its BOM. Windows Notepad always includes BOM in Unicode or UTF documents. Unfortunately, the BOM signature also causes problems with many applications and compilers (including TeX with UTF-8 encoding) and that makes it rather useless. . .

Without BOM and any convention as described above it is sometimes hard to distinguish between UTF-8 and ANSI (code page-specific) documents.

The Document Settings dialog has a CP Converter tab page. It can be used to change a document's format or reload the document in the proper code page in the rare case of WinEdt not being able to automatically determine the correct encoding.

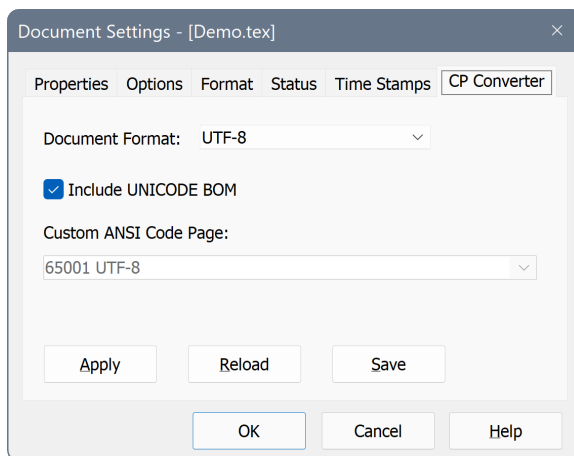


Figure 3.2: Document Settings: Code Page Converter

Help in the dialog explains how to use this functionality.

3.2 T_EX and International Characters (UTF-8)

Putting

```
\usepackage[utf8]{inputenc}
```

enables you to use UTF-8 (unicode) coding in L^AT_EX documents. As long as you open the document in WinEdt in UTF-8 mode you see the same characters in WinEdt as in your compiled document (as is the case with this UTF-8 document):

```
À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï Ñ Ò Ó Ô Õ Ö Ø Ù Ú Û Ü Ý Þ ß à
á â ã ä å æ ç è é ê ë ì í î ï ñ ò ó ô õ ö ø ù ú û ü ý ÿ € ¿
† ‡ § ¶ © ® Ć Š Ž č š ž
```

```
À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï Ñ Ò Ó Ô Õ Ö Ø Ù Ú Û Ü Ý Þ ß ;
à á â ã ä å æ ç è é ê ë ì í î ï ñ ò ó ô õ ö ø ù ú û ü ý ÿ € ¿
† ‡ § ¶ © ® Ć Š Ž č š ž
```

Not all UTF-8 characters are currently supported by L^AT_EX unless you load extra packages. For example the € (€) symbol requires:

```
\usepackage{textcomp} % required for \texteuro
\usepackage{eurosym} % required for \euro
% get a "nicer" looking euro symbol:
%\let\texteuro\euro % if you want \texteuro=\euro
```

Note the difference between the shape of the `\euro` (€) and `\texteuro` (€) symbols. Such issues are non-WinEdt related and you will have to consult T_EX's documentation or, if needed, seek help on the appropriate forum (such as T_EX Newsgroup where L^AT_EX related topics are discussed).

You might have noticed that the preamble of this document also contains:

```
\catcode'\•=13
\def\•{•} % WinEdt bullet (U+007F)-> Unicode Bullet (U+2022)
```

This allows T_EX to process an empty tabular environment as inserted by WinEdt's auto-completion functionality. Bullets are represented by •:

•	•	•	•	•
•	•	•	•	•
•	•	•	•	•
•	•	•	•	•

In WinEdt the shortcut **Ctrl+Space** (Tools menu -> *Next Bullet*) lets you move through placeholders and fill in the actual data.

3.3 Translation Tables

If you prefer your documents to contain plain T_EX notation for international characters (e.g. `\{A}` stands for Å) then you should consider applying WinEdt's read and write translation tables. This will make working with WinEdt more comfortable and it is required if you want to take advantage of WinEdt's spell-checking ability with international dictionaries. UTF-8 encoding is a better solution in most cases!

WinEdt can convert certain strings into their Unicode equivalents when the file is being read and then translate these characters back to the original strings representing international characters in T_EX notation.

Suitable translation tables for TeX mode are already defined (but not enabled) in the default settings: see Options Interface. The help in this interface provides the details.

For example, the default TeX_Read and TeX_Write translation tables contain definitions like:

```
"{\ss}" -> "ß"           "ß" -> "{\ss}"
"{\AA}" -> "Å"           "Å" -> "{\AA}"
"{\AE}" -> "Æ"           "Æ" -> "{\AE}"
"{\aa}" -> "å"           "å" -> "{\aa}"
"{\ae}" -> "æ"           "æ" -> "{\ae}"
"{\OE}" -> "Œ"           "Œ" -> "{\OE}"
"{\oe}" -> "œ"           "œ" -> "{\oe}"
"{\O}" -> "Ø"           "Ø" -> "{\O}"
"{\o}" -> "ø"           "ø" -> "{\o}"
"\c{C}" -> "Ç"           "Ç" -> "\c{C}"
"\c{c}" -> "ç"           "ç" -> "\c{c}"
"\^{A}" -> "Å"           "Å" -> "\^{A}"
"\~{A}" -> "Ã"           "Ã" -> "\~{A}"
"\\"{A}" -> "Ä"           "Ä" -> "\\"{A}"
...

```

Note that the last item is not a typo! To specify double quotes inside a double-quoted string they have to be repeated twice! Failing to observe this convention may completely corrupt WinEdt's translation table.

The read translation table supports two notations (e.g. `\^{A}` and `{\^A}`). The write translation table TeX_Write is the inverse of the read translation table (except that it uses the first, preferable, notation where applicable). You should use translation tables with some care: make a backup copy of your documents until you verify that the tables are set up correctly. Careless application of translation tables may irreversibly corrupt your documents (just like a global replacement)!

3.4 T_EX and UTF-8

While WinEdt supports Unicode pretty well¹ this is not necessarily the case for all T_EX engines. For example, pdfT_EX is not a Unicode application and supports only limited Unicode ranges, which may be enough for most international users that use the Latin alphabet but certainly is not sufficient for CJK or Middle East users. But any T_EX engine's limitation in this respect is not a WinEdt-related issue and you should seek information or help with this on L^AT_EX forums.

T_EX Systems such as MiK_T_EX and T_EX Live include alternative T_EX engines that have a better (or different) approach to handling UTF-8 documents containing characters outside the range supported by pdfT_EX. And WinEdt provides an easy interface to alternative T_EX engines such as XeT_EX and LuaT_EX. All you do is select your alternative from the drop-down button that lists different T_EX compilers. Once an alternative has been selected you do not have to use the drop-down portion of the button to activate the selected compiler until you want to change it again.

Furthermore, if you want to make XeLaT_EX or LuaLaT_EX your default PDFT_EXify engine you can easily do so on the TeX Options page of the Execution Modes dialog.

WinEdt also supports powerful *pragma comments* that can be used to define default TeX engines and other accessories (BibT_EX or Biber) on a per-document basis. This makes it possible to use universal compile (F9) and specify the information required to properly process your document as a pragma comment. Advanced users should take advantage of this functionality. For example:

```
% !BIB program = biber
% !IND exe = xindy.exe
% !IND opt = -L english -I xindy
```

If you need to compile your document with restrict-write18 option for all TeX compilers and engines specify:

```
% !TEX+ ext = --restrict-write18
```

You can also change PDFT_EXify behavior to use TeXify -> dvi->ps->pdf conversion with latex as TeX engine:

```
% !PDFTEXify method = ps
% !TeXify engine = latex
% !Max Iterations = 2
```

1. WinEdt uses a powerful MS Uniscribe library with support for bidirectional text and complex text processing.

3.5 Modes and Submodes

A document's mode is initially determined from its file type and is stored as a local attribute of the file in WinEdt's File List (Project File). This works in most cases for the main mode. However, bilingual users might want to tie certain attributes (such as dictionaries) to submodes that may not be apparent from the file type.

Instead of setting such modes through the Document Settings dialog or adopting a practice to name your files with more than one filetype (e.g. `Paper.fr.tex`) it is possible to enter submodes (as comments) in the first (or second) line of a document.

WinEdt modes can be specified as a comment:

```
% !Mode:: "Mode:Submode:Submode"
```

WinEdt also recognizes mode specification as used by emacs:

```
% -*- mode: TeX -*-  
% -*- coding: utf-8 -*-
```

It is also possible to specify mode and submode in a single comment:

```
% -*- TeX:DE:Soft:UTF-8 -*-
```

However, as Emacs might not recognize such specifications, it is better to use WinEdt's convention `!Mode:: "TeX:DE:Soft:UTF-8"` as described above.

Furthermore, for \TeX documents WinEdt also detects the language submode from `babel` and UTF-8 coding from the `inputenc` package:

```
// Determine Language Submodes from babel:  
// \usepackage[french,german,italian,spanish]{babel}  
  
// Determine Coding (UTF-8) from the preamble:  
// \usepackage[utf8]{inputenc}
```

This functionality is implemented through event handler macros that are executed before a document is loaded into WinEdt, and which ensure that WinEdt opens and treats the document properly.² The actual macro that is by default called from this event handler is

```
%b\Macros\Events\GetMode.edt
```

If for some reason mode detection (or some portion of it) from comments is unwanted for your style of work you can edit this macro and comment out unwanted portions or make any other desirable changes. However, note that this macro is executed frequently and it has to be fast or else you'll notice delays when opening documents or even when collecting data in previously unopened documents...

2. Event handlers are defined in the Advanced section of the Options Interface. Modes and submodes are explained further in WinEdt's Help Manual -> User Guide...

3.6 Spell Checking

WinEdt is by default configured to use English dictionaries. Bilingual users may immediately stumble upon the problem of using different dictionaries depending on their preferred or alternative languages. The new Dictionary Manager Dialog (Options Menu) can be used to automatically download and install extra dictionaries from www.winedt.org. With properly used modes and submodes WinEdt can be configured for multi-lingual users.

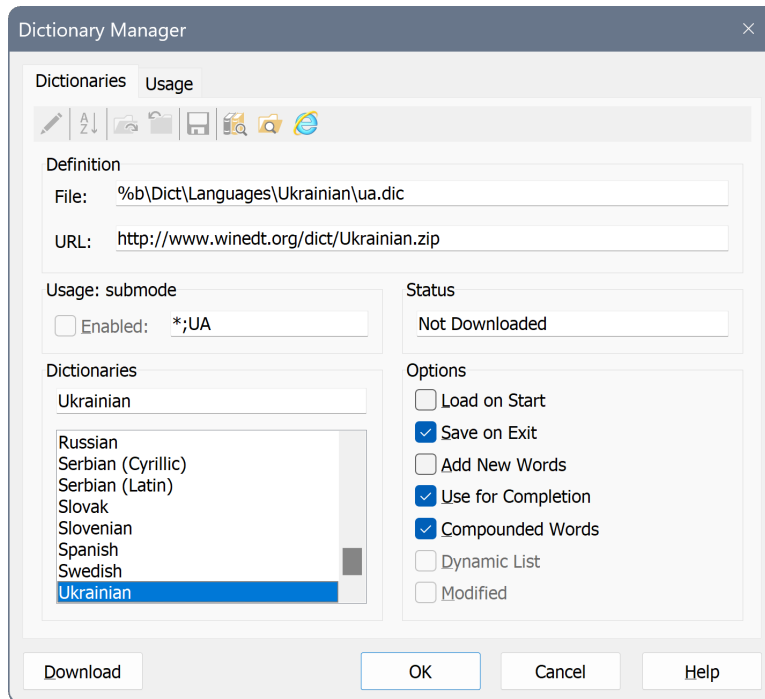


Figure 3.3: Dictionary Manager Dialog

WinEdt underlines misspelled and (another common typo) repeated words. Right-clicking on such a word brings up a spelling dialog that offers suitable suggestions based on the currently enabled dictionaries. Right-clicking on a repeated word prompts you with the option to remove the duplicate. It is also possible to spell-check the whole document (or a selected portion of it) or to manually highlight the next or previous typing mistake by using the items in the Tools Menu and their associated shortcuts.

WinEdt is often used to edit documents containing markup languages such as TeX. Not all environments in such documents should be subject to spell-checking as they may not contain plain text. WinEdt can be configured to turn spell-checking off in such places. By default, it is already loaded with an extensive list of such exceptions for TeX documents. More can be added by users with special needs. . .

3.7 WinEdt Dictionaries (Word Lists)

Any word list (a text file consisting of one word per line) can be converted into a WinEdt dictionary by sorting it as a dictionary (Sort Lines dialog) and saving it as a Unicode (UTF-16) file with BOM and UNIX line terminators (Document Settings dialog). Neglecting to properly sort and save a dictionary will result in increased loading time and possible problems with the spell-checking functionality! Words have to be at most 255 characters long and a dictionary can contain at most 16,000,000 lines. These are reasonable limitations for all situations arising in practice. Lines at the beginning of a dictionary starting with % characters are considered comments and are not subject to sorting. That's all. The hard part is compiling a complete word list for any available language.

WinEdt's dictionaries use Unicode (UTF-16) format with BOM and LF as the line terminator. This means that WinEdt 6 (or earlier) ANSI dictionaries are not compatible with the Unicode versions of WinEdt. However, the CP Converter section of the Documents Settings dialog makes it easy to convert any non-Unicode dictionary into the new format. Help in the Document Settings dialog gives the details, showing, as an example, how the conversion of the Russian dictionary into a Unicode (ready-to-use) format was done.

Of course, all dictionaries included in the default settings are already properly formatted and ready to use!

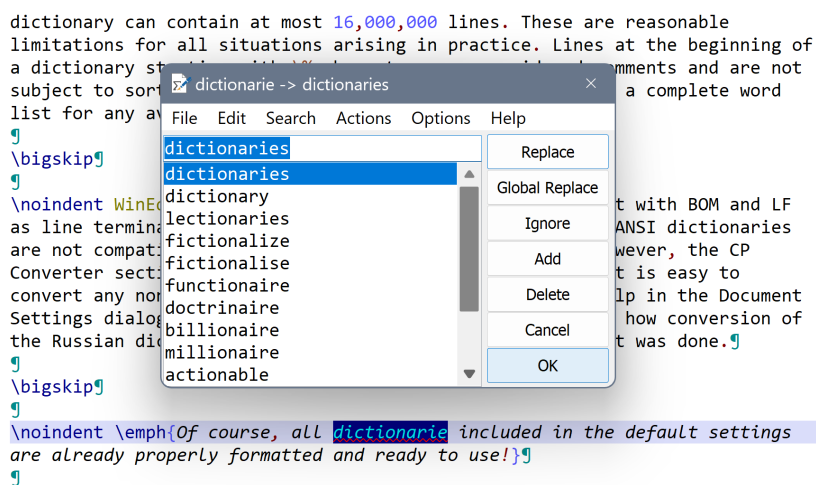


Figure 3.4: Spell-checking in action

If you right-click on an underlined misspelled word WinEdt brings up a dialog with spelling suggestions. The correct suggestions are usually close to the top. Double-clicking on it will substitute a misspelled word with the selected correction.

4 L^AT_EX Demo: Non-WinEdt-Related Bonus

4.1 Graphics Inclusion

Graphics inclusion in T_EX documents is not WinEdt-related and the documentation that comes with your T_EX System (e.g. `graphicx` package) should be consulted. However, below are a few examples that show that it can be done! These work with my version of MiK_TE_X compiled with PDF_LA_TE_X but they come with no guarantee and no support from the WinEdt Team.

The first thing to know is that different graphic formats are supported by T_EX depending on the intended output format (dvi or pdf). Here we will only focus on pdf because dvi format is seldom used these days and using intermediate formats like dvi and ps to create a pdf file via converters is not a good practice.¹ PDF_LA_TE_X supports pdf, png, and jpeg graphics formats.



Figure 4.1: WinEdt classic and new-style logo in a suitable png format

-
1. This document cannot be compiled to dvi format because we did not provide alternative (legacy) graphic formats like eps for the included images (on purpose!). Should you create an eps image for each graphic file in the Images folder you will be able to compile it to dvi format. But YAP (and other DVI viewers) will not properly display landscape pages or rotated tables. Although some users may disagree, using `dvipdf` or `dvips+pspdf` to create pdf files with eps images is not very efficient: eps images have to be converted to pdf every time the source is compiled. Converters usually degrade the graphics quality and can result in other problems such as misaligned or improperly placed images. It is much better to create images that can be included in pdf documents and convert non-supported formats once and forever! They are not supported because they are deemed obsolete and this will not change. If intermediate formats work for you that's fine. If not there is no point complaining to the WinEdt Team since now you know how we feel about it. Besides WinEdt is completely irrelevant to such issues. . .



Figure 4.2: SV Adriana docked

JPEG is a good format for pictures (non-vector graphics) like the one above. PNG format is suitable for vector graphics or computer screenshots like the one below:



Figure 4.3: About WinEdt Dialog

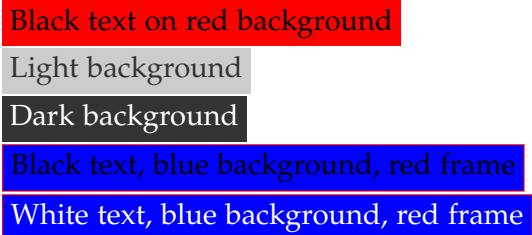
Now you know it can be done! Learn how to use `graphicx` package if you need to do something more sophisticated. . .

4.2 Color Package Example

This section is borrowed from MiKTeX's Samples folder. You should consult more documentation and examples that come with your TeX System...

Text starts off in green a little red nested blue text returning to green

1. magenta cmyk black
2. predefined blue gray text



This is how it is done:

```
\usepackage{color}
...
\begin{enumerate}
  \item \textcolor{cmyk}{0,1,0,0}{magenta cmyk} black
  \item \color[gray]{0.5} \textcolor{blue}{predefined blue}
    gray text
\end{enumerate}

\definecolor{Light}{gray}{.80}
\definecolor{Dark}{gray}{.20}
\colorbox{red}{Black text on red background}
\par\colorbox{Light}{%
  \textcolor{Dark}{Light background}}
\par\colorbox{Dark}{%
  \textcolor{white}{Dark background}}

\fcolorbox{red}{blue}{Black text,
  blue background, red frame}

\fcolorbox{red}{blue}{\color{white}%
  White text, blue background, red frame}
```

4.3 Rotated objects examples

rotation = 30°				
1	2	3	4	5
A	B	C	D	E
a	b	c	d	e

rotation = 330°				
1	2	3	4	5
A	B	C	D	E
a	b	c	d	e

Table 4.1: Rotated tables



Figure 4.4: L^AT_EX Logo rotated by $\pm 15^\circ$

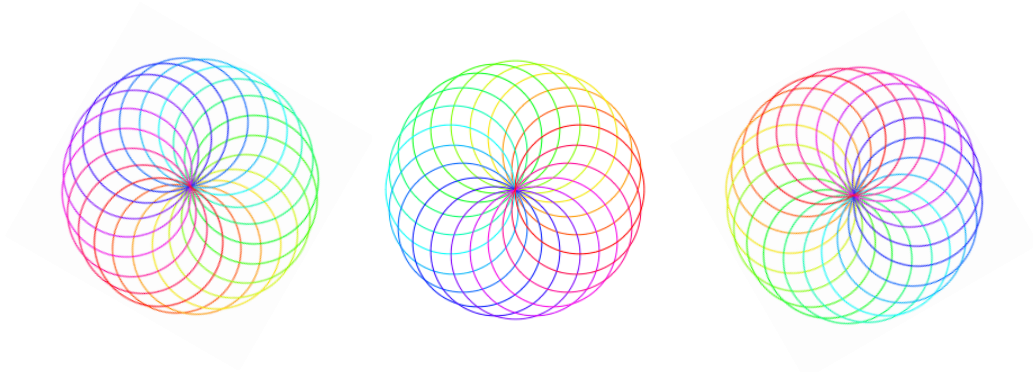
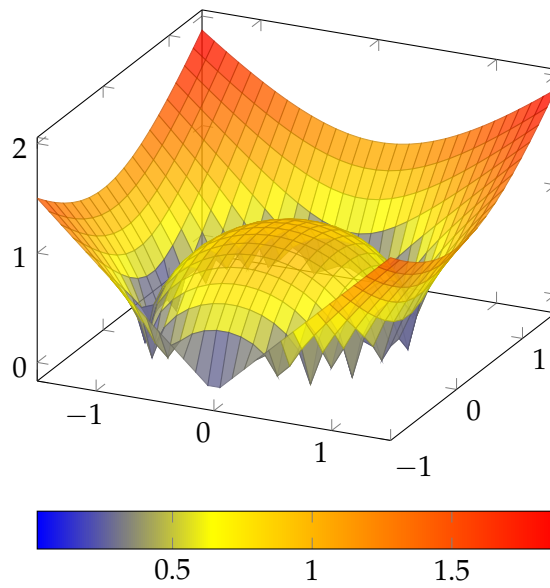
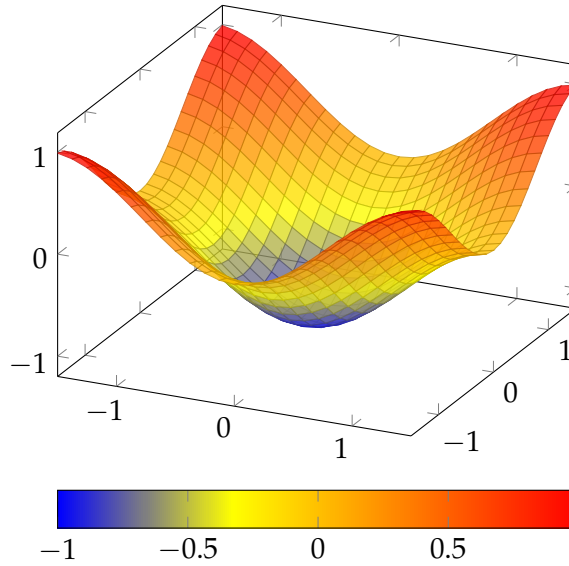


Figure 4.5: Rotated circles

4.4 Pgfplots: Tikzpicture Example

Below are some examples of 3-dimensional vector graphics generated with pgfplots package:



4.5 Landscape mode page example

The preamble must load the following package for this to work:

```
\usepackage{landscape} %landscape pages support
```

Name	Base colors/notions	Parameter range	Default
rgb	<i>red, green, blue</i>	$[0, 1]^3$	
cmy	<i>cyan, magenta, yellow</i>	$[0, 1]^3$	
cmk	<i>cyan, magenta, yellow, black</i>	$[0, 1]^4$	
hsb	<i>hue, saturation, brightness</i>	$[0, 1]^3$	
Hsb	<i>hue^o, saturation, brightness</i>	$[0, H] \times [0, 1]^2$	$H = 360$
tHsb	<i>hue^o, saturation, brightness</i>	$[0, H] \times [0, 1]^2$	$H = 360$
gray	<i>gray</i>	$[0, 1]$	
RGB	<i>Red, Green, Blue</i>	$\{0, 1, \dots, L\}^3$	$L = 255$
HTML	<i>RRGGBB</i>	$\{000000, \dots, FFFFFFF\}$	
HSB	<i>Hue, Saturation, Brightness</i>	$\{0, 1, \dots, M\}^3$	$M = 240$
Gray	<i>Gray</i>	$\{0, 1, \dots, N\}$	$N = 15$
wave	<i>lambda (nm)</i>	$[363, 814]$	

L, M, N are positive integers; H is a positive real number

Table 4.2: Table in landscape mode example (from xcolor's documentation)

4.6 Another landscape page example

The preamble must load the following package for this to work:

```
\usepackage{landscape} %landscape pages support
```

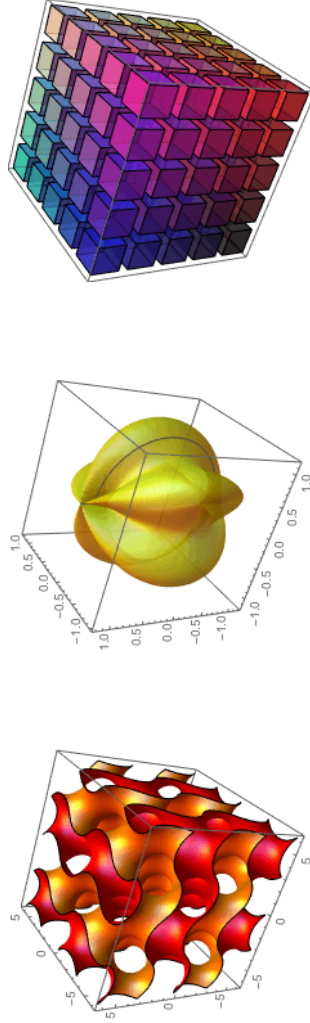


Figure 4.6: Mathematica 3D-objects in landscape mode

4.7 Presentations in L^AT_EX

Presentation packages and software that can be used with L^AT_EX are not WinEdt-related topics. However, since we frequently get asked about such things we posted a question to WinEdt's Mailing List and the response was overwhelming.

Most users thought that beamer is currently the best when it comes to ease of use and the quality of the output. Alternatives texpower and seminar have also been mentioned. . .

You can use MiK_TE_X's Package Manager to install beamer. MiK_TE_X's doc folder has all the documentation and examples that will help you start working on your presentations. Some users mentioned that they had to upgrade their MiK_TE_X to be able to compile the examples, which rely on up-to-date packages. If you encounter any such problems you may have to do the same. . .

4.8 L^AT_EX and paper size

The best way to set the paper size in L^AT_EX documents is to use the `geometry` package. T_EX itself does not have a notion of output paper size and this package is essential if you need to change the paper size.

The following will properly handle paper size in the printed document:

```
\usepackage[letterpaper]{geometry} % or a4paper
```

Some users use `dvi2pdf` or even `dvips` and then `ps2pdf` because these conversions seem to handle their choice of paper size correctly while PDF_LA_TE_X does not. However, this does not make much sense as default paper size settings in these converters may change in future versions. Furthermore, and as already mentioned, using intermediate formats to produce a pdf document is not very efficient and tends to result in problems.

Learning how to use the `geometry` package is a much better way to handle paper size issues. And this package can do much more. You should read its documentation to learn about it. Type *geometry* in the interface that can be started from WinEdt's Help Menu -> [LaTeX Doc](#) (or `Shift+Ctrl+F1` shortcut) and you can open the pdf manual describing this package² in detail.

2. Both the package and its documentation are, of course, a part of your T_EX system and not WinEdt! WinEdt is completely irrelevant when it comes to such issues. If you need assistance you should seek help on T_EX forums.

4.9 A Simple Revision Control System (RCS)

On www.winedt.org you'll find a link to the page that describes how to use RCS or CS-RCS with WinEdt. RCS (Revision Control System) is the proper way to deal with revisions...

However, simple revisions or corrections done by the copy editor and intended for the authors can be handled in a much simpler manner. WinEdt provides a sample \LaTeX package `xrcs.sty` that can be used for such editing. The package defines two macros `\RCSAdd{...}` and `\RCSDel{...}`. These two macros can be used to mark simple additions and deletions, respectively. In WinEdt the environments are colored in blue and red (as defined in the Switches section of the Options Interface). Depending on the options the compiled document can contain additions and/or deletions (in color or plain text).

Furthermore, the package also provides a tag `\RCSMark{...}` which can be used to mark the argument with a yellow marker, and `\RCSRem{...}` which can be used to include remarks. All four RCS tags are defined as switches in WinEdt's default highlighting scheme for TeX mode.

The `xrcs.sty` package provides the following options (with the default values displayed in red):

active	inactive
marker	nomarker
remarks	noremarks
new	nonew
old	noold

Examples of usage:

```
\usepackage[active,new,old,remarks,marker]{xrcs}
\usepackage[active]{xrcs} % Only Additions- in blue colors
\usepackagep[active,old,nonew]{xrcs} % Only old text - in red
\usepackage[nomarker]{xrcs} % Only Additions: final version
```

In your preamble you have to also include the color package:

```
\usepackage{color}
```

Text example:

```
\RCSMark{IMPORTANT;} WinEdt's \RCSDel{menu}\RCSRem{use capitals!}
\RCSAdd{Menu} should be thought of as an Action List...
```

with `\usepackage[active,new,noold,marker]{xrcs}` is processed as:

IMPORTANT: WinEdt's **Menu** should be thought of as an Action List...

Besides the highlighting definitions for switches `\RCS*{...}` WinEdt also has a popup menu `Edt RCS` containing some commands that can make the revisions easier. This popup menu is displayed in response to the `Alt+R` keystroke. The properties of the popup can be adjusted through the Options Interface (Popup Menus)...

The `xrcs.sty` file is included with this document's sources. Feel free to make changes and improvements... Note that the WinEdt Team does not provide support for this package: take it or leave it...

```
% -----
% File: xrcs.sty
%
% A (very) simple Revision Control System for LaTeX2e/WinEdt
% *****
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{xrcs}[2005/01/30 v0.002 RCS]
\RequirePackage{color}
\newif\ifMarker \Markertrue
\newif\ifRemarks \Remarkstrue
\newif\ifAddDel \AddDeltrue
\newif\ifAddNew \AddNewtrue
\newif\ifAddOld \AddOldfalse
\DeclareOption{active}{\AddDeltrue}
\DeclareOption{inactive}{\AddDelfalse}
\DeclareOption{marker}{\Markertrue}
\DeclareOption{nomarker}{\Markerfalse}
\DeclareOption{remarks}{\Remarkstrue}
\DeclareOption{noremarks}{\Remarksfalse}
\DeclareOption{new}{\AddNewtrue}
\DeclareOption{nonew}{\AddNewfalse}
\DeclareOption{old}{\AddOldtrue}
\DeclareOption{noold}{\AddOldfalse}
\ExecuteOptions{inactive,noold,noremarks,new,marker}
\ProcessOptions
% -----
\def\RCSMark#1{\ifMarker{\colorbox{yellow}{#1}}\else#1\fi}
\def\RCSRem#1{\ifRemarks{\textsf{#1}}\fi}
\def\RCSDel#1{\ifAddOld\ifAddDel{\color{red}#1}\else#1\fi\fi}
\def\RCSAdd#1{\ifAddNew\ifAddDel{\color{blue}#1}\else#1\fi\fi}
%-----
```

Once again, this is a very simplified revision system; it is somewhat primitive and it is lacking all the features available in proper RCS... However, it may be of some interest since it is very simple to use: in any text editor it is easy to search for `\RCS...`

4.10 Useful T_EX-ing Hints

Check the source code of this document in WinEdt. Pay attention to comments included in the preamble...

For author-year references use:

```
\usepackage{natbib}
```

Specify your bibliography database in a different folder:

```
\bibliography{Biblio/articles.bib}
```

Note that you have to specify the path UNIX-style (using forward instead of backward slash as folder separator). Avoid spaces in filenames (some T_EX accessories may not work properly with spaces in the filename specification).

Most T_EX Systems allow you to place your bib files in a separate folder on your localtexmf tree. For details consult the documentation that comes with your T_EX System. MiK_TE_X users can create a bibtex folder in their localtexmf tree, place their bib files there, and refresh the FNDB in MiK_TE_X's Options interface.

For fancy pdf files use:

```
\usepackage{hyperref}
\hypersetup{
  pdftitle={Shown in AR File Information},
  pdfstartview=FitH,    % Fit the page horizontally
  bookmarks=true,      % Open Bookmarks in AR
}
% more options can be found in
% TEXMF/doc/latex/hyperref/manual.pdf
```

To manually correct the hyphenation of a word that was not properly handled by T_EX (e.g. Weltauffassung) put the following in the preamble:

```
\hyphenation{Welt-auf-fas-sung}
```

To prevent long titles in your table of contents (generated by L^AT_EX) use the alternative short title:

```
\section[Short Title for TOC]{Long long long title}
```

You can find everything about T_EX and L^AT_EX on: TUG...

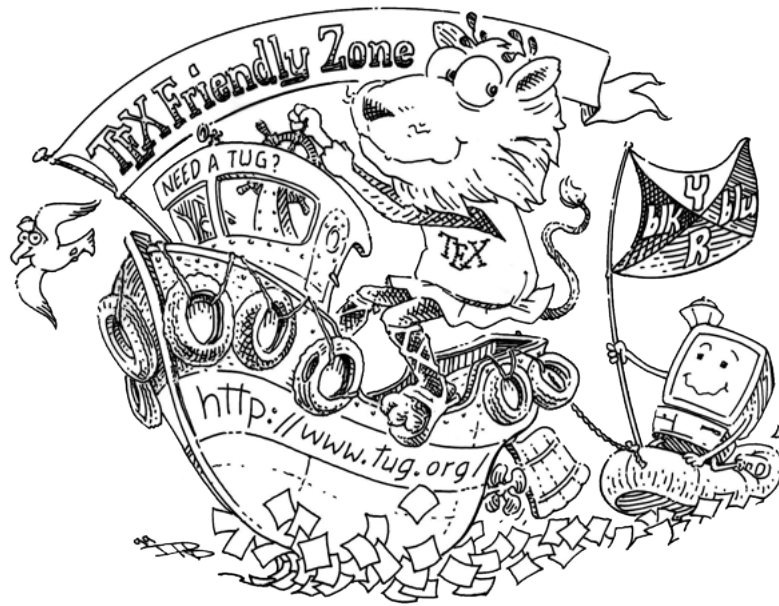


Figure 4.7: Happy \TeX -ing from TUG and the WinEdt Team

4.11 In memoriam Echo (2005-2016)

Echo, our faithful companion and a long-time CSO (Chief Security Officer) at WinEdt HQ, passed away on January 21, 2016 (just over a month short of her eleventh birthday).



Figure 4.8: All Things Must Pass

She is being missed, no kidding...

Bibliography

TUG Online Book Catalogue. *Books about T_EX and Friends*. TUG Web Site, 2017. URL <https://www.tug.org/books>.

[Everybody loves books, and everybody loves T_EX. On this page you'll find it all...](#)

Donald E. Knuth. *The T_EXbook*. Addison-Wesley, 1984. ISBN 0-201-13448-9.
The basis of everything, covering plain T_EX.

Leslie Lamport. *L^AT_EX: A Document Preparation System*. Addison-Wesley, second edition, 1986. The basis of everything, covering L^AT_EX and L^AT_EX2_ε.

Helmut Kopka and Patrick Daly. *A Guide to L^AT_EX2_ε*. Addison-Wesley, fourth edition, 2003. ISBN 0-321-17385-6. Covers core L^AT_EX from the ground up, and discusses commonly used packages for graphics, web integration, and more.

Frank Mittelbach, Michel Goossens, Johannes Braams, and Chris Rowley. *The L^AT_EX Companion*. Addison-Wesley, second edition, 2004. ISBN 0-201-36299-6.
Covers core L^AT_EX, plus a vast array of additional packages.

David Griffiths and Desmond Higham. *Learning L^AT_EX*. SIAM, second edition, 1997. ISBN 978-0-898713-83-1. A short example-based book covering core L^AT_EX and a few packages.

Herbert Voß. *Typesetting Mathematics with L^AT_EX*. UIT Cambridge, 2010. ISBN 978-1-906860-17-2. A practical book on typesetting mathematics with L^AT_EX, covering many packages.

Herbert Voß. *Typesetting tables with L^AT_EX*. UIT Cambridge, 2011. ISBN 978-1-906860-25-7. A practical book on typesetting tables with L^AT_EX, covering many packages.

George Grätzer. *First Steps in L^AT_EX*. Birkhauser, 1999. ISBN 0-8176-4132-7.

George Grätzer. *Math into L^AT_EX*. Birkhauser Boston and Springer Verlag New York, 2000. ISBN 0-8176-4131-9 and 3-7648-4131-9.

George Grätzer. *More Math into L^AT_EX*. Birkhauser, 2007. ISBN 978-0-387-32289-6.