

The Kile Handbook

**Jonathan Pechta, Federico Zenith, Holger
Danielsson, Thomas Braun, and Michel Ludwig**



The Kile Handbook

Contents

1	Preface	1
1.1	Requirements	1
1.2	Intended Audience	1
2	Introduction	3
2.1	Basic facts	3
2.1.1	About Kile	3
2.1.2	Kile and Kate	3
2.1.3	What is L ^A T _E X	3
2.1.4	How do you pronounce it? Why that strange typesetting?	4
2.2	L ^A T _E X 101	4
2.3	Kile's Main Features	5
2.3.1	QuickStart Wizard	5
2.3.2	Predefined Templates	5
2.3.3	Syntax Highlighting	6
2.3.4	Auto-Completion of Environments	6
2.3.5	Jump to Structure Element	6
2.3.6	Inverse Search	7
2.3.7	Forward Search	7
2.4	The Toolbar	7
3	Quickstart	11
3.1	Writing a L ^A T _E X Document with Kile for Beginners	11
3.2	Environments	12
3.3	Using Kile	12
3.4	DVI Files	13
3.4.1	Viewing a DVI	13

The Kile Handbook

3.4.2	Printing a DVI	13
3.4.3	Converting DVI files	14
3.5	Forward Search between Kile and Okular	14
3.6	Inverse Search between Kile and Okular	14
3.7	Resolving Errors	15
4	Starting a New Document	16
4.1	Templates	17
4.1.1	Create a New Template	17
4.1.2	Configuring Automatic Substitutions	17
4.1.3	Create a Template from the Wizard	17
4.1.4	Creating a Template from any File	18
4.1.5	Removing a Template	18
5	Editing L^AT_EX Documents	19
5.1	The L ^A T _E X Reference	19
5.2	Cursor Movements	19
5.3	Brackets	20
5.4	Highlighting	20
5.5	Bullets	20
5.6	Select	21
5.6.1	Select L ^A T _E X commands	22
5.7	Delete	22
5.8	Environment	23
5.9	T _E X Group	23
5.10	Double Quotes	24
5.11	Smart Newline	25
5.12	Smart Tabulator	25
6	Code Completion	26
6.1	Automatic Environment Completion	26
6.2	L ^A T _E X Commands	26
6.3	Environments	28
6.4	Abbreviations	29
6.4.1	Abbreviations	30
6.5	Autocompletion Modes	30
6.5.1	L ^A T _E X Commands	30
6.5.2	Document Words	31
6.6	Writing Own Completion Files	31

7	Wizards and Dialogs	32
7.1	Include Graphics	32
7.2	Array Wizard	35
7.3	PostScript® Utilities	35
7.4	Document Statistics	39
8	Special Tags in L^AT_EX	40
8.1	Using the L ^A T _E X Tag Library	40
8.2	Using Bibitems	43
8.3	User-Defined Tags	44
8.3.1	Placeholders in User-Defined Tags	46
9	The Build Tools	48
9.1	Compiling, converting and viewing	48
9.1.1	BibT _E X	48
9.1.2	makeidx	48
9.1.3	MetaPost and Asymptote	49
9.1.4	PDFL ^A T _E X	49
9.1.5	L ^A T _E X to Web	49
9.1.6	Passing Command-Line Parameters	49
9.2	Quick Preview	49
9.2.1	Selection Mode	50
9.2.2	Environment Mode	51
9.2.3	Subdocument Mode	51
9.2.4	Mathgroup Mode	51
9.2.5	Quick Preview in Bottom Bar	51
9.3	Graphic File Formats	52
9.3.1	L ^A T _E X and PDFL ^A T _E X	52
9.3.2	Graphics Conversion	52
9.3.3	Use the right File for the right Graphic	53
9.4	EPS Graphics	54
9.4.1	L ^A T _E X and EPS Graphics	54
9.4.2	The PostScript® Way of Kile	54
9.4.3	The PostScript® Way and Bitmap Graphics	55
9.4.4	PDFL ^A T _E X and EPS Graphics	56
9.5	Master Document	57
9.6	Error Handling	57
9.7	The Watch File Mode	58

The Kile Handbook

10 Navigating the L^AT_EX Source	59
10.1 Using the Structure View	59
10.1.1 Using the Context Menu	60
10.1.2 Updating the Structure View	61
10.2 Bookmarks	62
11 Projects	63
11.1 Working with Projects	63
11.2 Creating a Project	64
11.3 The Files and Projects View	64
11.4 Adding and Removing Files	65
11.5 Project Options	65
11.5.1 Archiving your Project	66
11.6 Closing a Project	66
12 Document Encoding	67
12.1 The ucs Package	68
12.2 XeLaTeX	69
12.3 CJK Support	69
12.3.1 CJK Troubleshooting	70
12.3.2 How do I input CJK in Unicode?	71
13 Scripting	72
13.1 Scripting in Kile	72
13.2 API Reference	72
14 Help	74
14.1 Help Documents	74
14.2 Context Sensitive Help	74
14.3 Searching for Keywords	75
14.4 User Defined Help	77
15 Credits and License	80

Abstract

Kile is a T_EX and L^AT_EX source editor and shell.

Chapter 1

Preface

1.1 Requirements

To Run Kile, you will be required to have the following components installed on your system:

- [K Desktop environment \(KDE\)](#): KDE is a popular open-source desktop environment.
- [Qt](#): Qt™ is a C++ GUI and network library needed to compile Kile.
- [LATEX](#): high-quality document typesetting program. Most likely you want the TeX Live (or on older systems the teT_EX) package, if you are on a Unix-like system.

Most of these items might be included in your Linux® distribution; please refer to your distribution documentation, or refer to your installation CD or DVD, for adding these packages to your computer.

The Kile project includes several binary packages of Kile for different distributions that can be easily installed and run without having to compile; check out the [Kile homepage](#).

1.2 Intended Audience

This manual is intended for any individual, regardless of her or his experience with L^AT_EX, KDE, Kile or Linux®.

Advanced users are not likely to read this manual, but all suggestions on documentation will be considered. If you would like to contribute to this project or the documentation, please consult the [Kile web page](#).

The Kile Handbook

Need answers about Kile? Are you stuck with your compile? Want to see a new feature? The preferred way to ask technical questions or start a discussion is to use our mailing list: kile-devel@lists.sourceforge.net.

Chapter 2

Introduction

2.1 Basic facts

2.1.1 About Kile

Kile is an integrated L^AT_EX environment for the KDE desktop. Kile gives you the ability to use all the functionalities of L^AT_EX in a graphical interface, giving you easy, immediate, and customized access to all programs for L^AT_EX codecompletion, compiling, postprocessing, debugging, conversion and viewing tools; you also get very handy wizards, a L^AT_EX reference and a powerful project management.

2.1.2 Kile and Kate

Kile is based on Kate, so a lot of editing capabilities come from Kate and Kile extends these capabilities to edit L^AT_EX documents. To learn more about Kate and its capabilities, see the [Kate Handbook](#). Kile users should start reading from the chapters ‘Working with the Kate Editor’ and ‘Configuring Kate’.

2.1.3 What is L^AT_EX

L^AT_EX is a text-processing system derived from T_EX, a program developed originally in 1977 by Donald Knuth to help him layout his work professionally, obtaining a work similar to a typesetter’s; the typesetter is the professional that styles a document’s look according to specifications for the type of document. L^AT_EX was created by Leslie Lamport to give authors an automatic typesetter, especially when it came to the expensive and painstaking typesetting of mathematical formulas and expressions, that *by no chance* are enclosed in dollar signs

in L^AT_EX. Today, word-processing programs let any user be the typesetter; but what you often want is a document that looks good, not one that you spent hours on to make it look good. L^AT_EX takes that burden, and lets you think of the document, not of the layout. And yes, it *will* look good!

2.1.4 How do you pronounce it? Why that strange typesetting?

There is a funny traditions of T_EX-related packages to have the strangest pronunciation and typesetting possible. T_EX was supposed to be brought in from the Greek *τεχνη*, in Latin letters *tech*. There are a lot of explanations why, but most likely it is because T_EX was originally conceived for technical reports, and indeed its foremost ability was the correct and easy typesetting of mathematical formulae, then an extremely expensive, time-consuming and frustrating business.

The pronunciation is supposed to be as follows: *T* as you would expect, *E* as in *get*, and *X* as in the German *ich*. If you do not know what *ch* sounds like, it is more or less as an hissing cat; the IPA symbol is /ç/. Many people report the different pronunciation of *ach* (IPA symbol /x/), but I have personally asked some Greeks, and can confirm the first version. You should be aware that a lot of people mispronounce /teks/ or /tek/.

Last, in L^AT_EX the first L^A is pronounced as *lay*: the idea is that, while raw T_EX is difficult, even a *layman* can use L^AT_EX macros. A less inspiring, but more realistic explanation is that it stems from the surname of Leslie Lamport, the creator of L^AT_EX. Now you know!

2.2 L^AT_EX 101

The L^AT_EX typesetting system is similar to other markup languages as XML, used for many types of documents (including the one you are reading), or HTML, used for web pages. The general idea is about having specially wrapped keywords, called *tags*, that tell a program (a word processor, a web browser, or the L^AT_EX compiler) how to present text. Kile offers you a good number of such tags in the menu LaTeX in the menu bar.

While we will try to give you a good idea of what L^AT_EX is, this is not The Definitive Book on L^AT_EX. If you want to learn L^AT_EX well, you may want to loan a book in your library. The writer has had a good experience with A Guide to L^AT_EX by H. Kopka and P. W. Daly, and still keeps it on the shelf.

As other markup languages, L^AT_EX contains a *preamble*, which sets global commands, such as paper size, page numbering, dimensions of the text on the page, and a *body*, that contains the document text; the preamble contains at least the `\documentclass` command, and precedes the body, that starts with the command `\begin{document}`, and is quite predictably concluded by the command `\end{document}`.

2.3 Kile's Main Features

2.3.1 QuickStart Wizard

The QuickStart wizard built into Kile is a useful feature to quickly start creating documents in Kile. Choosing the wizard from the menubar gives you several choices for the creation of your document. You can also specify some options related to the document right away.

Class options:

- Document Class: choose the type of document you want to create: article, book, letter, report, scrartcl, scrreprt, scrbook, prosper, beamer or other custom-defined.
- Typeface Size: tell Kile what point size (pt) you want to use.
- Paper Size: choose the size or style of sheets.
- Encoding: In general it is a good idea to use your systems standard encoding. Modern systems now move more and more to UTF-8 as the standard encoding. If you can, use utf8x (which is indeed the correct spelling for L^AT_EX documents).
- Other options: this allows you to set further options such as printing, draft, and others.

Packages

This lists some of the most common additional packages used in L^AT_EX. Select the check box to include it.

Document Properties

- Author: put your name here.
- Title: add the document title here.
- Date: specify the date.

2.3.2 Predefined Templates

The predefined templates in Kile are:

- Empty document: real freaks start from scratch!
- Article: sets the article format, for a document short enough not to be broken down to chapters.
- Report: sets the report format, for a middle-sized document, with for example page numbering on the page's outer edge.

- **Book:** sets the book format, a full-fledged flavor, so powerful that it is used to write many university textbooks.
- **Letter:** sets the letter format, that can automatically do those nasty indentations that nobody really remembers.
- **Beamer,HA-Prosper:** create nice presentations in PDF with a superior look and all L^AT_EX power.
- **Powerdot:** Powerdot is the follower of the packages `seminar` and `HA-Prosper`. It hasn't so much options like Beamer, but it is easy to use and creates really nice presentations in PDF.
- **Scrartcl,Scrbook,Scrpert,Scrltr2:** the KOMA-Script document classes, especially adapted to german typography. Use them whenever you write german texts.
- **PDF:** a modified `Article` template to use with PdfLaTeX.
- **Xelatex:** an modified `Article` template to use with XeLaTeX.

Remember that you can take all these templates as starting point for your own requirements.

New users need not to worry, this list is just a brief description of features, and more attention will be paid to complete these tasks in detail later in chapter 3.

2.3.3 Syntax Highlighting

Kile is similar to programs that deal with source code and editing, and will automatically highlight commands, options and items that are used (and abused). Kile makes it possible to spot easily problem areas: for example, if you see major areas of text turned green, it is likely that you forgot closing a math environment somewhere; you would have noticed anyway by how crappy the output file would have looked, but highlighting really saves you time and frustration.

2.3.4 Auto-Completion of Environments

The auto-completion of environments means that, when you begin a new environment by typing `\begin{environment}`, Kile will automatically insert a matching `\end{environment}` command, with a line in between them for your text. You can of course deactivate it if you want in Settings → Configure Kile... → LaTeX+Environments.

2.3.5 Jump to Structure Element

All documents are normally structured in a hierarchy of some type. L^AT_EX allows you to break up documents into the following hierarchy (part being highest in the hierarchy, and subparagraph being lowest):

- `\part`
- `\chapter`
- `\section`
- `\subsection`
- `\subsubsection`
- `\paragraph`
- `\subparagraph`

When viewing a document in the Structure view, you can jump between elements by clicking on the element you would like to view.

2.3.6 Inverse Search

When creating your own L^AT_EX files, inverse search can be very helpful. Once you have created a DVI file (DeVice Independent File) or PDF file, you can click the left mouse button while pressing **Shift** in the viewer and Kile will jump to the corresponding line in the L^AT_EX source code.

A DVI is a type of file containing a description of a formatted document, along with other information including character font, and is besides PDF the usual output of T_EX or L^AT_EX. A number of utilities exist to view, convert and print DVI files on various systems and devices.

2.3.7 Forward Search

When using inverse search, the selection of items in the DVI or PDF file is associated with the editor, so when you click on the DVI or PDF file, the main window jumps to the corresponding section of L^AT_EX code in the editor. Forward search is the exact opposite of this. Forward search will allow you to click on a specific section of text in the L^AT_EX code, and jump to the associated position in the viewer window.

2.4 The Toolbar

- New: begin a new document.
- Open: open a new document.
- Close: close your document.
- Define document as master: this is used when working with multiple files. Having a master document will let you work more easily with other `.tex` files included in your document. If you are using projects, you can also set in Project-> → Project Options a project-wide master document.

The Kile Handbook

- Quickbuild: compiles your L^AT_EX source code and displays the results automatically, unless you have code errors.
- Watch file mode: this mode will "watch" the DVI file for changes, and will not launch a new session of Okular after Quickbuild.
- View logfile: views the `.log` file, so you can spot errors.
- Previous error: jumps backward through the `.log` file and highlights errors in source.
- Next error: jumps forward through `.log` file and highlights errors in source.
- Stop: halts current tool.
- LaTeX: runs L^AT_EX on the active document.
- ViewDVI: launches DVI viewer.
- DVItO_{PS}: converts a DVI to a PostScript® (PS).
- ViewPS: launches PostScript® (PS) viewer.
- PDFLaTeX: runs PDFL^AT_EX on the active document.
- ViewPDF: launches the PDF viewer.
- DVItO_{PDF}: converts a DVI to a PDF.
- PStO_{PDF}: converts a PS to a PDF.
- ViewHTML: views HTML created.
- ForwardDVI: jump to page of the DVI file that corresponds to the current line in the editor.
- ForwardPDF: jump to page of the PDF file that corresponds to the current line in the editor.

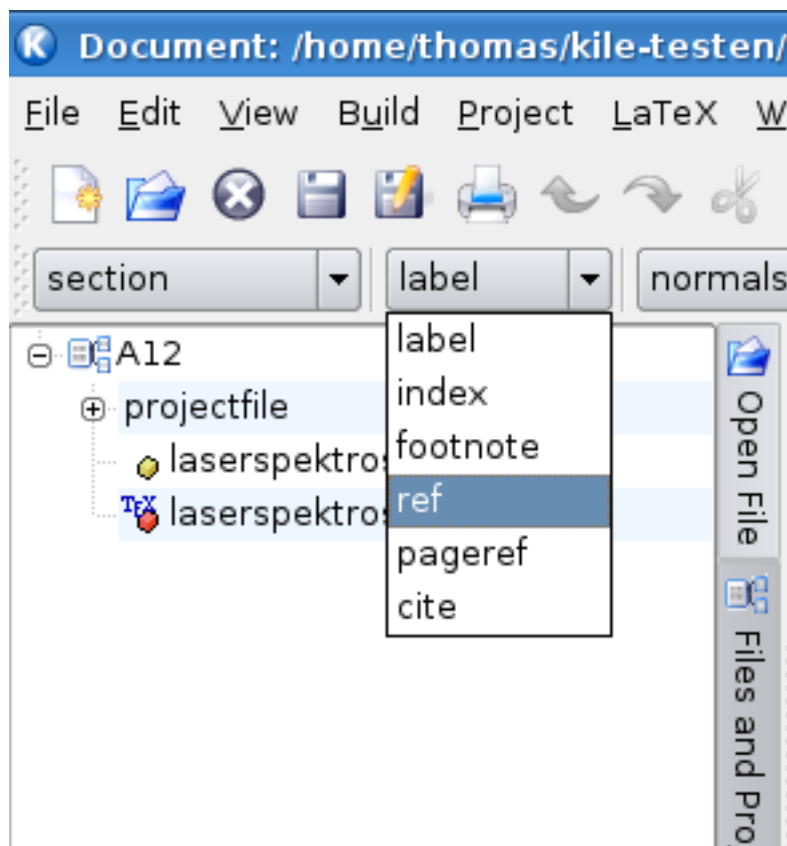
If you look at the Edit toolbar, you will notice three large drop-down menus. The drop-down menus were designed for you to be able to quickly add certain common features into your document. The first drop down box is used for quickly dividing your document by parts, chapter, sections and so on; the available commands to add segments to your L^AT_EX source code are:

- part: highest level of sectioning for a document.
- chapter: starts a new chapter.
- section: create a new section.
- subsection: create a new subsection.
- subsubsection: a secondary section between subsection and paragraph.
- paragraph: create a new paragraph.
- subparagraph: create a new subparagraph.

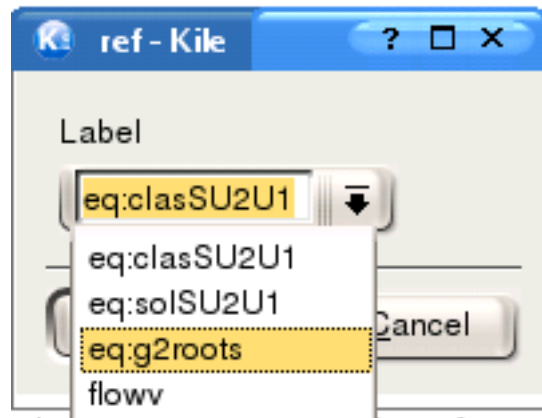
The Kile Handbook

The drop down box named label is used to insert items to your document such as indexes, footnotes, and references; the available commands are:

- label: a command that produces a label for a chapter, a figure or another element.
- index: creates an entry for the index.
- footnote: creates a footnote in your document.
- ref: used to refer to a predefined label, which you can choose from a drop-down list.
- pageref: just like ref, but refers to a page instead of a structure element.
- cite: create a reference with data from a bibliography.
- cite from ViewBib: ask the ViewBib tool for all selected references and insert them. Currently this is only available with [KBib](#) as ViewBib tool.



The Label drop-down menu



Selecting the label for a reference

When using cite, you are presented with a drop-down list of bibitems, but if you are using Bib_T_EX this will only work if the file belongs to a Project. For editing Bib_T_EX files the usage of specialised editors is recommended. The author has made good experience with KBib_T_EX and KBib. Of course you can also write the Bib_T_EX files by hand inside Kile.

The last drop-down box labeled tiny is used to set the size of the text. You can set the size of the main text, of footnotes, and so on. The available commands are:

- tiny: smallest.
- scriptsize: very small.
- footnotesize: smaller.
- small: small.
- normalsize: normal.
- large: large.
- Large: larger.
- LARGE: even larger.
- huge: still larger.
- Huge: largest.

Chapter 3

Quickstart

3.1 Writing a L^AT_EX Document with Kile for Beginners

Users of Kile have two choices when starting a new document: they can use the Wizard to begin a new document, select the type of document they would like to create and options such as font size, paper size, and so on; otherwise, they can write the code by hand.

```
\documentclass[12pt]{article}
\begin{document}
Here is a bunch of text coded in \LaTeX.
\end{document}
```

Every document in L^AT_EX begins with the command `\documentclass[optional argument]{class}`, where `class` specifies the document type.

Typing in the code example above from the text box gives you the following output:

Here is a bunch of text coded in L^AT_EX.

Compiled text in DVI output

The brackets that come after the command `\documentclass` contain the options for the command. The option `[12pt]` sets the size of the font for your article; if you do not set the font size in the beginning, you can set it later in the text.

Once you have typed in the code example from the box above, you will need to compile your L^AT_EX source code. The easiest way for you to compile L^AT_EX is to use the Build menu, or using the Quickbuild button.

Alt-2 is the keyboard shortcut to compile your source code.

You have to save your source code before you can compile; Kile will do this automatically for you.

If your document did not compile, check the log for errors. When using the Quickbuild key, the Okular viewer should be launched automatically; if it does not, look at the log.

3.2 Environments

An environment is a segment of text that is managed differently than the rest of the document. For example, you create a report with font size 12, but you need to change your font size for a few sentences. The commands `\begin{environment}`, `\huge` and `\end{environment}` will let you temporarily alter the text inside the environment commands to be size huge.

Changes are only effective from `\begin{environment}` to `\end{environment}`. There are no limits as how many changes you can make inside an environment.

There are many features you can add to your document that will make it more readable and user-friendly. You can add features such as specific fonts, bold, italics, underline etc. to your document, and these commands will end with either an `\end` command, or at the end of your environment.

- `\begin{emph}`: this command makes text italicized, and is valid until the code comes across a `\end{emph}`, or another environment. To italicize one word in a sentence, you can use the syntax: this is `\emph{my}` sentence.
- `\textbf{I am making this text inside the brackets bold}`: this command makes your text bold.
- `\quote`: to create a quote inside your document; begin your quote with `\begin{quote}` and end it with `\end{quote}`.
- `\center`: centers the text.
- `\verse`: creates offset text for poems.
- `\itemize`: makes an itemized list.

3.3 Using Kile

Now that we have given you some background about how to write code using the L^AT_EX markup language, we will show you how to create a document using Kile step-by-step.

1. Start Kile.
2. Select Wizard → Quick Start, then choose a format, and set your preferences in the wizard.
3. Once the wizard has entered text, do some customization to make the document more readable, add a minimum of one quote, some bold text, italics, and a verse to see the difference between the commands.
4. Save your file, and give it the name `intro.tex`.
5. Build your document using Alt-2, or the button labeled LaTeX.
6. Select ViewDVI.
7. Check out all your new text.
8. When you are done viewing your document, click the Editor View button or press Ctrl-E to return to the editor if you are using the embedded viewer, or close the viewer window if you are using a separate viewer.

That's it! You have just created your first L^AT_EX document!

Once you have created your DVI, you will be able to print your document, or change it into a PostScript® or PDF file if you want. Experiment and have fun!

3.4 DVI Files

DVI stands for *DeVice Independent* file. These files are produced by T_EX or L^AT_EX to be read by a driver of some sort on your computer. There are many different types of output that a `.dvi` can be sent to, such as a printer, PostScript® or PDF file converter, or your computer screen.

3.4.1 Viewing a DVI

You have already seen how to view a DVI file on the screen by using the ViewDVI button in the toolbar.

3.4.2 Printing a DVI

To print a DVI, you can use the same process that you used to create your document earlier (see Section 3.3). At step 7, after clicking ViewDVI, select File → Print in the viewer, and if you have your printer properly configured, you will be able to print the DVI.

3.4.3 Converting DVI files

The toolbar gives the options of Converting a DVI to other formats. Once you have created a DVI from your L^AT_EX source code, you will be able to export it to a format of your choice using the toolbar buttons.

3.5 Forward Search between Kile and Okular

The forward search functions allow you to jump from your editor directly into the associated position of the DVI or PDF file.

Kile offers a configuration with this option for all L^AT_EX binaries. Go to Settings → Configure Kile... → Tools+Build and always choose the Modern configuration.

To execute a forward search, position the cursor on a line of source code, and click Forward Search to jump to the associated position in the DVI or PDF viewer window.

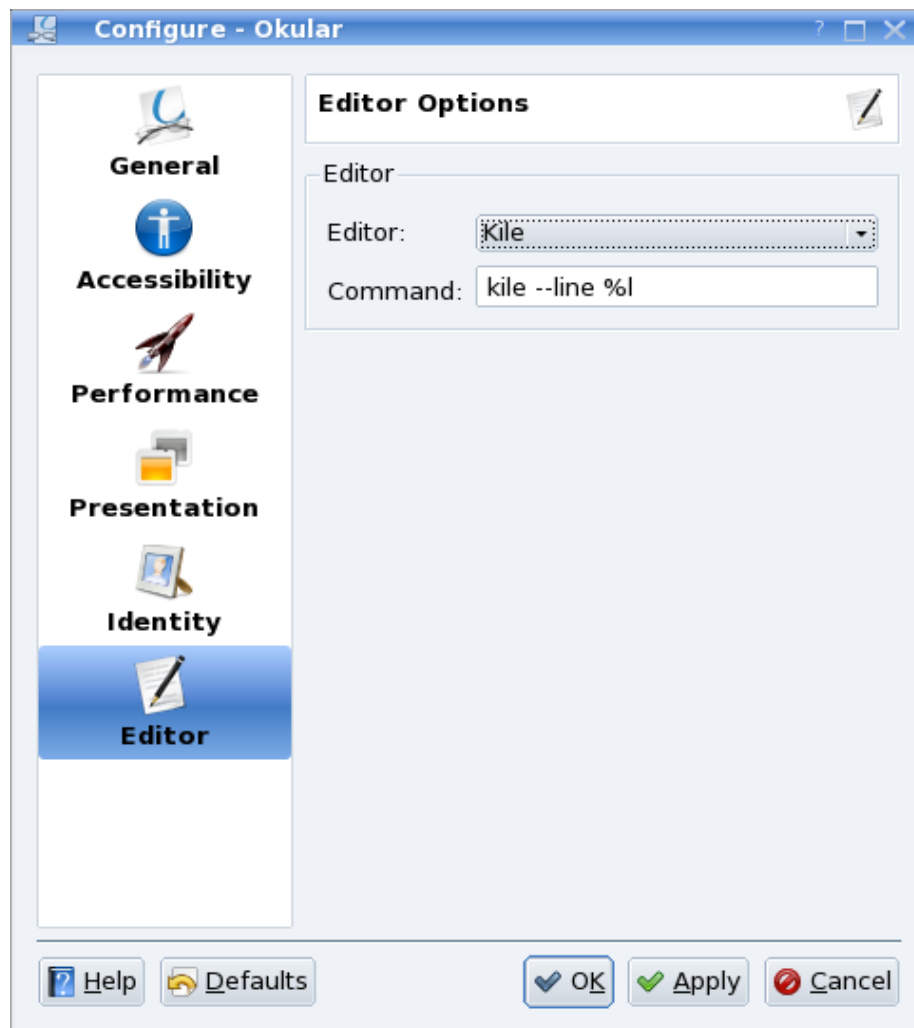
3.6 Inverse Search between Kile and Okular

Inverse search is a very useful feature when you are writing a L^AT_EX document yourself. If everything is set up properly, you can click into Okular's window with the left mouse button while pressing **Shift**. After that Kile loads the L^AT_EX source file and jumps to the proper paragraph. To use inverse search, you have to compile your L^AT_EX file with the Modern configuration.

Inverse search cannot work unless:

- The source file has been compiled successfully.
- Okular knows which editor you would like to use.

With this feature of Okular, a left mouse click while pressing **Shift** in the DVI or PDF document will result in Kile opening the corresponding L^AT_EX document and attempt to go to the corresponding line. Remember to tell Okular to use Kile as a text editor, in Okular's menu item Settings → Configure Okular... (on the page Editor).



Configuring Okular

3.7 Resolving Errors

If you are trying to use quickbuild, and the DVI viewer does not open, chances are you have an error. If you have an error, it will be visible in the log file / message area, and the summary of the error will be given.

The log file will explain the source of the error in your code. In the editor, you can use the buttons in the toolbar labeled Previous LaTeX Error and Next LaTeX Error to jump to and from errors. The log file always states in which line the error occurred. To view the line where an error occurred, click on the error in the log window, and Kile will take you to error's line.

Chapter 4

Starting a New Document

When you click the button in the toolbar to begin a new document a dialog appears, asking which type of template you would like to use to write your document. The default choices are:

- Empty document
- Article
- Beamer
- Book
- HA-Prosper
- Powerdot
- Letter
- Report
- Scrartcl (from the KOMA-Script package)
- Scrbook (from the KOMA-Script package)
- Scrlttr2 (from the KOMA-Script package)
- Scrreprt (from the KOMA-Script package)
- PDF
- XeLaTeX

If you selected an Empty document, you can either start writing a document from scratch, or you can use the wizard to quickly start a new document (see [Section 2.3.1](#)).

4.1 Templates

Frequent users of L^AT_EX typically use the same preamble for almost every document they use. Templates can be created, saved and loaded within Kile to make it easier to start a new document.

4.1.1 Create a New Template

To create a new template, you must first either open a T_EX / L^AT_EX file, or create a file of your own. Kile can generate a template from an existing document by opening the desired document and selecting File → Create Template from Document.

4.1.2 Configuring Automatic Substitutions

When creating a new document by selecting a template from File → New, certain character combinations will be replaced by data such as your name, or the character encoding you are using. These variables can be configured in Settings → Configure Kile... → Settings+General.

When designing your own template, it is useful to know which character combinations are replaced by which template variables:

- `$$AUTHOR$$`: This string will be replaced by the author variable.
- `$$DOCUMENTCLASSOPTIONS$$`: This string will be replaced by the document-class options variable. Typically this is used as follows: `\documentclass[$$DOCUMENTCLASSOPTIONS$${article}`
- `$$INPUTENCODING$$`: If the `inputencoding` variable is set to, say, `latin1` this string is replaced by `\input[latin1]{inputenc}`.

4.1.3 Create a Template from the Wizard

The easiest way to create a new template is to start the wizard, and then add commands in the editor. Once you have your document set up the way you like:

1. Save your file;
2. Go to File;
3. Choose Create Template from Document;
4. Make any corrections necessary to the template;
5. Enter a name for your new template;
6. Click OK to add your template to the menu.

Next time you start up a new document, you will be able to choose your customized template instead of the default ones.

4.1.4 Creating a Template from any File

A template can be created from any L^AT_EX file. If you are looking for an easy way to configure a template, go find one you like on the Internet and follow the same steps listed in Section 4.1.3.

For instance, you may want to create a full-fledged A0 poster. These posters are usually seen at scientific conferences, and L^AT_EX will help you making an attractive, catchy poster. You can get a template for A0 posters at [Jonathan Marchini's home page](#), but many more are available. Remember that you will need the `a0poster` package, which is normally not in standard tex distributions included. Download it from [here](#) and place it in the same folder as your L^AT_EX file.

4.1.5 Removing a Template

To remove a template from Kile, do as follows:

1. Go to File → Remove Template...;
2. A dialog box will appear with all templates listed: select a template;
3. Click OK, and your template will be removed.

Templates marked with an asterisk (*) cannot be removed without the proper permission.

Chapter 5

Editing L^AT_EX Documents

The internal editor that Kile uses is Kate. Kate is a text editor created for programmers, which incorporates the ability to read and highlight many different types of text files, among which are L^AT_EX and BibT_EX; you can access many options for Kate directly from Kile's Tools menu.

To learn more about Kate and its capabilities, see the [Kate Handbook](#). Kile users can start reading from the chapter 'Working with the Kate Editor'.

5.1 The L^AT_EX Reference

Kile features a very practical L^AT_EX tag reference, which you can access by choosing Help → LaTeX Reference. It contains a thorough description of almost all the commands that you may use in L^AT_EX and their syntax.

5.2 Cursor Movements

To select text, you have the following options:

- Hold left mouse button, and drag mouse to highlight text.
- Click once on a word to move the cursor to a new area.
- Click twice on a word to select the whole word.
- Click twice on a word and pressing Ctrl to select the whole tex word. This means clicking in this way on `\par` from `\par\bigskip` only select `\par`.
- Click three times to select the whole sentence.

Holding the left mouse button, and dragging the text you want to select, automatically copies the selected text to the clipboard.

Holding **Shift** and using the arrow keys allows you to select portions of the source code in the editor window.

5.3 Brackets

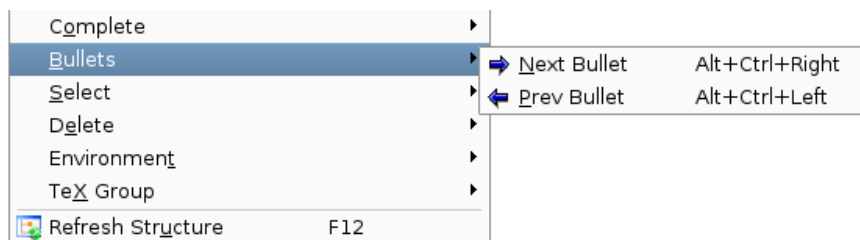
Bracket completion is a visual tool that the editor view uses to indicate you which bracket matches which. If you open any `.tex` file, and select any bracket, whether it be a parenthesis `()`, square brackets `[]` or braces `{}`, the editor will highlight the bracket and its match in yellow (this default color can be changed). So, for example, if you position the cursor on the braces in `\section{Introduction}`, you would see `\section{Introduction}` in the default yellow highlight, showing you the location of the beginning and ending brackets.

5.4 Highlighting

Kile has the ability to look for and highlight different types of code. For example, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ commands are distinguished from normal text, and math formulas are highlighted also in a different color.

5.5 Bullets

Many wizards can insert optional bullets, a special kind of bookmarks within the text. The menu entries `Edit → Bullets` or the corresponding keyboards shortcuts will allow you to jump to the next or last bullet. This will also highlight this bullet so that it will be deleted automatically, when you enter your first letter.



Next Bullet (Ctrl+Alt+Right) Jump to the next bullet in the text if there is one.

Last Bullet (Ctrl+Alt+Left) Jump to the previous bullet in the text if there is one.

5.6 Select

Editing is of course one of the main aspects when you use a program like Kile. Although Kate already has great capabilities, Kile adds some important features, which are especially needed to write L^AT_EX source. L^AT_EX always needs a lot of environments and groups, so Kile supports very special commands to select them. Under Edit → Select you will find the following commands to select text.

Bullets		
Select		
Delete		
Environment		
TeX Group		
Refresh Structure	F12	
	Environment (inside)	Alt+Ctrl+S,E
	Environment (outside)	Alt+Ctrl+S,F
	TeX Group (inside)	Alt+Ctrl+S,T
	TeX Group (outside)	Alt+Ctrl+S,U
	Math Group	Alt+Ctrl+S,M
	Paragraph	Alt+Ctrl+S,P
	Line	Alt+Ctrl+S,L
	TeX Word	Alt+Ctrl+S,W

Environment (inside) (Ctrl+Alt+S,E) Select an environment without the surrounding tags. If this command is called, when an environment is already selected, Kile will expand the selection to the next surrounding environment.

Environment (outside) (Ctrl+Alt+S,F) Select an environment including the surrounding tags. This selection can also be expanded with a second call of this command.

TeX Group (inside) (Ctrl+Alt+S,T) Select a TeX group inside the surrounding braces.

TeX Group (outside) (Ctrl+Alt+S,U) Select a TeX group including the surrounding braces.

Math Group (Ctrl+Alt+S,M) Select the current math group including the math commands.

Paragraph (Ctrl+Alt+S,P) Select a whole paragraph, i.e. a group of text lines separated on both sides by empty lines. A paragraph does not mean just continuous lines of text, as it is in other text editors. This extended meaning also includes tables, L^AT_EX commands and all other lines of source. The only important thing for Kile is that kind of paragraph is separated by two empty lines.

Line (Ctrl+Alt+S,L) Select the text line of the current cursor position.

TeX Word (Ctrl+Alt+S,W) Select the word under the current cursor position. This selection has also an extended meaning, because this command can also select L^AT_EX commands, which begin with a backslash and may also have an optional star at the end.

5.6.1 Select L^AT_EX commands

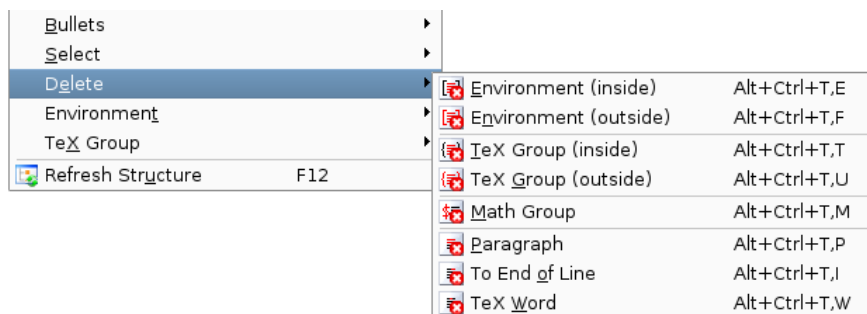
Kile has an extended feature to select L^AT_EX commands. If you for example have typed

```
text \bfseries\itshape more text
```

and double click on one of the L^AT_EX commands, both will be selected. But often you only want to select one of two or more commands. This can be done using the Ctrl key. You only have to press the Ctrl key and a double click will only select the desired command.

5.7 Delete

To delete some parts of a document you can of course select them, and then use the Delete key. Kate also offers the command Ctrl-K which deletes the whole line. But Kile offers a faster way with its own delete commands. Under Edit → Delete you will find following commands to delete text.



Environment (inside) (Ctrl+Alt+T,E) Delete an environment without the surrounding tags.

Environment (outside) (Ctrl+Alt+T,F) Delete an environment including the surrounding tags.

TeX Group (inside) (Ctrl+Alt+T,T) Delete a TeX group inside the surrounding braces.

TeX Group (outside) (Ctrl+Alt+T,U) Delete a TeX group including the surrounding braces.

Math Group (Ctrl+Alt+T,M) Delete the current math group including the math commands.

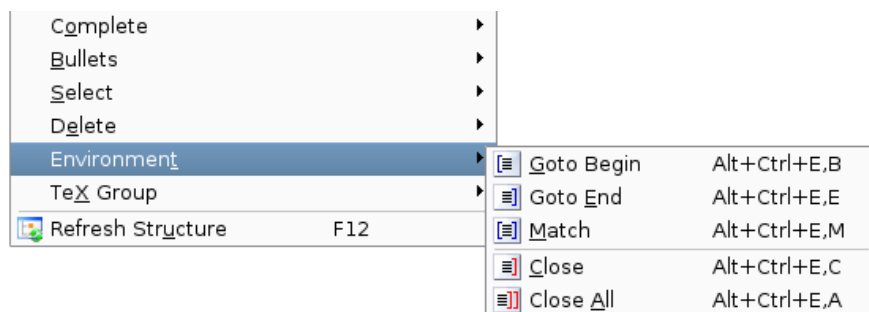
Paragraph (Ctrl+Alt+T,P) Delete a whole paragraph. Look at the Select → Paragraph command, how a paragraph is defined in Kile.

To End of Line (Ctrl+Alt+T,I) Delete the text from the current cursor position to the end of the line.

TeX Word (Ctrl+Alt+T,W) Delete the word or L^AT_EX command under the current cursor position.

5.8 Environment

It was already mentioned that environments are a central point in L^AT_EX. So Kile offers five other commands to make the work with L^AT_EX as easy as possible under submenus Edit → Environment.



Go to Begin (Ctrl+Alt+E,B) This command will jump to the beginning of the current environment, wherever your current position is. The cursor will be placed directly in front of the opening environment tag.

Go to End (Ctrl+Alt+E,E) This command will jump to the end of the current environment, wherever your current position is. The cursor will be placed directly behind the closing environment tag.

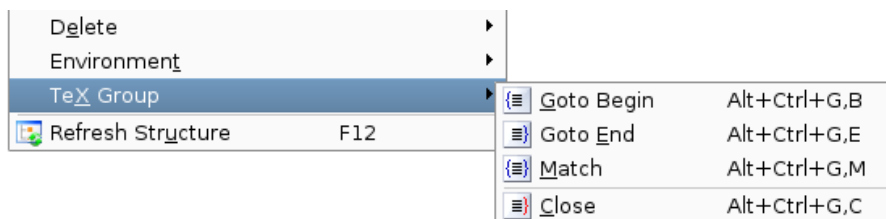
Match (Ctrl+Alt+E,M) When your cursor is placed in front of or above the `\begin{environment}` tag, it will be moved to the opposite end of the environment and vice versa.

Close (Ctrl+Alt+E,C) Typing a lot of nested environment tags, you may lose control of all those environments. This command will close the last opened environment, so that the nested structure of environments will not be broken.

Close All (Ctrl+Alt+E,A) This closes all open environments, not only the last opened environment.

5.9 T_EX Group

Kile also offers some special commands for L^AT_EX groups, which are determined by braces `{...}`. In submenu Edit → TeX Group you will find some important commands, which correspond to those from Edit → Environment.



Go to Begin (Ctrl+Alt+G,B) This command will jump to the beginning of the current group, wherever your current position is. The cursor will be placed directly in front of the opening brace.

Go to End (Ctrl+Alt+G,E) This command will jump to the end of the current group, wherever your current position is. The cursor will be placed directly behind the closing brace.

Match (Ctrl+Alt+G,M) When your cursor is placed in front of or behind an opening brace of a T_EX group, he will be moved to the opposite end of the group and vice versa.

Close (Ctrl+Alt+G,C) Typing a lot of nested group braces may be hard work. This command will close the last opened group, so that the nested structure of T_EX groups will not be broken.

5.10 Double Quotes

In L^AT_EX, two single quotes are used as double quotes. To help you insert these efficiently, Kile allows you to press " to insert two opening single quotes. Furthermore, if you want to close a quotation, you also have to press ". Kile will be smart enough to recognize this case and inserts two closing quotes for L^AT_EX.

To get a literal double quote on the other side, press " twice.

You can enable or disable this auto insertion of opening and closing double quotes in section Settings → Configure Kile... → LaTeX.



If you also include language-specific options like `ngerman` or `french`, you will also be able to use German or French double quotes. Many more languages are available.

5.11 Smart Newline

If you press Ctrl-Return, Kile inserts an intelligent newline. If your current position is inside a list environment, like `enumerate` or `itemize`, Kile will not only insert a newline, but also add a `\item` command.

If you are inside a tabular environment, Kile will finish the current line with `\\`, followed by the newline.

If you are inside a L^AT_EX comment, Kile will start the next line with a `%`.

Even better, Kile is smart enough to support predefined L^AT_EX and user defined environment, which can be added in section Settings → Configure Kile... → LaTeX.

5.12 Smart Tabulator

Some users like to arrange columns in tabular environments and put all ampersand characters `&` beneath each other. Kile tries to support this. If you press Alt-Shift-`&`, Kile will look for the next tab in the row above. Although this tab may not be the corresponding tab, Kile will add some spaces to adjust the column position with the current tab.

Chapter 6

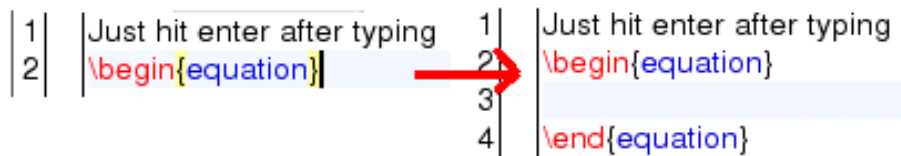
Code Completion

Although Kate already offers a good completion mode, Kile extended code completion to support some special methods especially for L^AT_EX. Five different modes are integrated. Three of them work on demand, the other two are autocompletion modes. All modes can be configured to work very differently at Settings → Configure Kile....

6.1 Automatic Environment Completion

When you begin a new environment, typing `\begin{environment}`, Kile will automatically add an `\end{environment}` command, with a line in between for your text.

Autocompletion can be turned off in the L^AT_EX section of Settings → Configure Kile... → LaTeX+Environments.

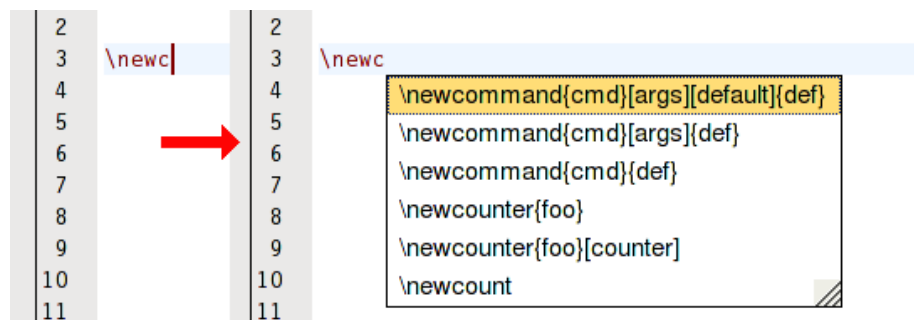


Completing an Equation Environment

6.2 L^AT_EX Commands

When you type some letters, you can activate this completion mode for L^AT_EX commands and normal words with Edit → Complete → (La)TeX Command or

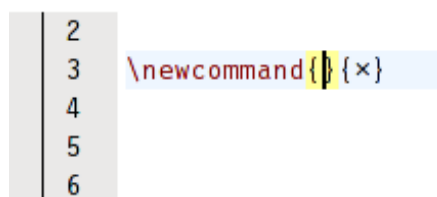
the keyboard shortcut Ctrl-Shift-Space. Kile first reads the letters from the current cursor position to the left and stops at the first non-letter character or a backslash. If this pattern begins with a backslash, Kile will enter completion mode for T_EX or L^AT_EX commands. Otherwise it enters normal dictionary mode, where you will not find any L^AT_EX commands. Depending on the chosen mode, a completion box will be opened. You will see all commands or words whose beginning matches the current pattern. You can navigate with the cursor keys through this list and select one entry with **Enter** or a double click with the mouse.



When you push the **Backspace** key, the last letter of your pattern will be deleted, and the completion list may grow. On the other hand, if you type another letter will expand the pattern and the visible word list may shrink.

If you decide not to select any of the suggestions, you can leave this dialog with **Esc**.

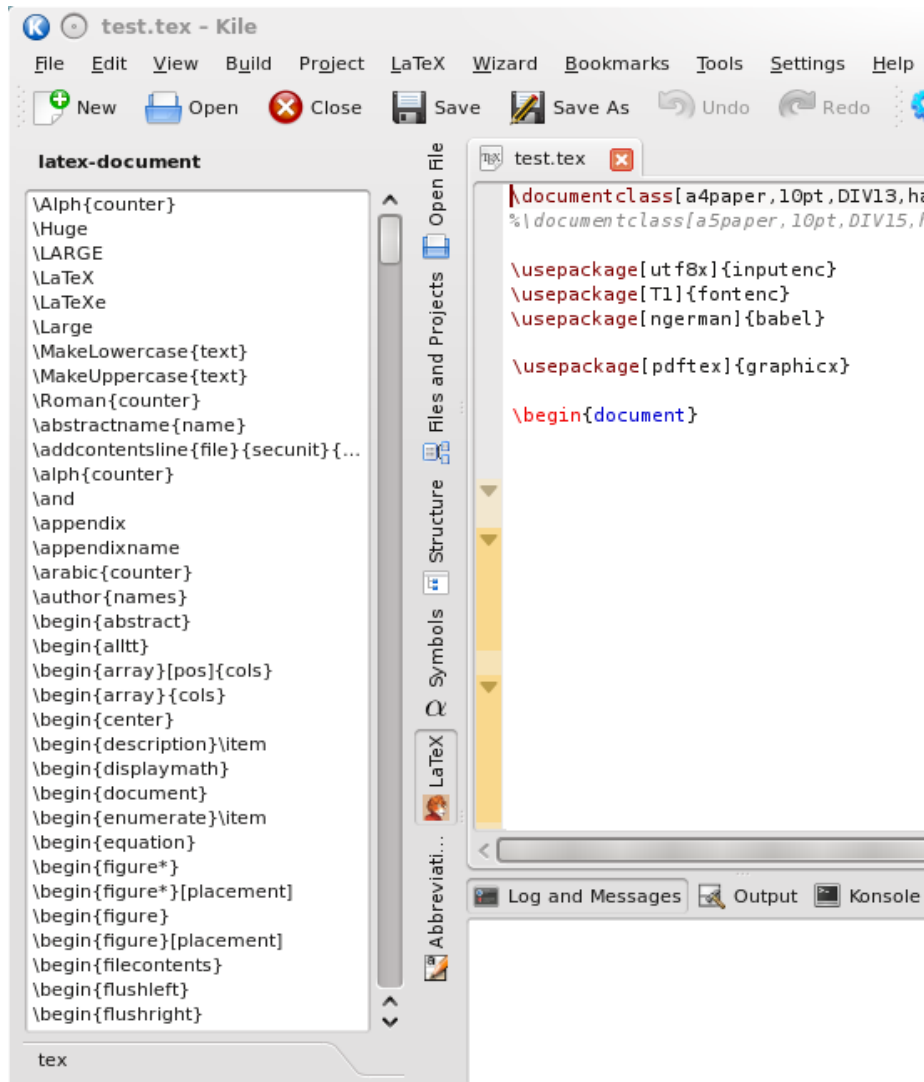
You will see that all commands are written with a short description of their parameters. These descriptions are of course stripped when you select a command. Optionally you can let Kile insert bullets at these places, so that you can easily jump to these positions with Edit → Bullets → Next Bullet and insert the parameter you want.



Go to Settings → Configure Kile... → Kile+Complete to configure one or more of these lists. You can choose different word lists for T_EX and L^AT_EX commands and dictionary mode for normal words.

If you choose the option Show Latex commands, the entries of all chosen cwl files for L^AT_EX command completion are shown in a separate view of Kile's sidebar. So you will have a good survey, which commands are available and what parameters and options must or can be given for a completion. You can also simply select one entry with a mouse click and it will be inserted into the document, where all named parameters and options are stripped.

The Kile Handbook



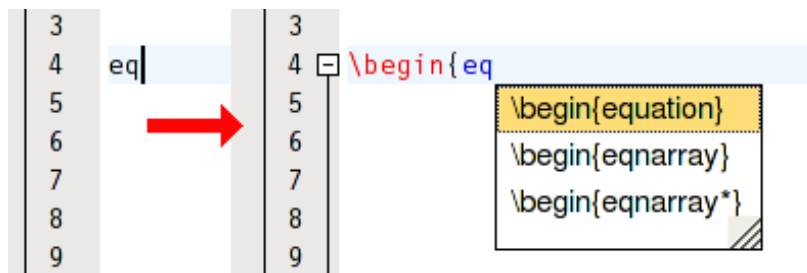
As each chosen wordlist will be shown in a separate view for its own, there could be too much views, so that Kile's main window may be greater than small screens. As this looks very ugly, Kile works with a maximum number of allowed views, which is set to 10. If this value is too big for your screen, you should reduce this number.

6.3 Environments

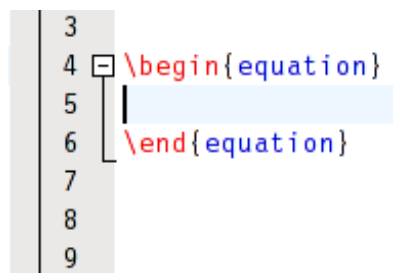
The *command mode* is not useful to complete environments. You always have to type some letters of `\begin`, and invoking the completion mode will result in a huge list of environments tags. On the other hand, environments are so

often used that Kile offers a special mode to complete environments. Forget the opening tag and write, for example, `eq`.

When you call the completion mode with `Edit → Complete → Environment` or keyboard shortcut `Alt-Shift-Space`, the opening tag is automatically added and you will see `\begin{eq}`. After this change, the completion list is much less cluttered.



Now select an environment, and you will see that it is also automatically closed. Even more, if Kile recognizes it as a list environment, it will also insert a first `\item` tag.



Go to `Settings → Configure Kile... → Kile+Complete` to configure one or more of these lists. This mode uses the same word lists as the completion mode for `TEX` and `LATEX` commands.

6.4 Abbreviations

Kile supports user defined lists of abbreviations, which are replaced on demand by longer text strings. Look at `Settings → Configure Kile... → Kile+Complete` to configure one or more of these lists. For the example given here, the abbreviation list in `example.cwl` must be chosen. In this file you will find for example the entry `L=\LaTeX` for example.

For example, type only the letter `L`. Now invoke the abbreviation mode of word completion with `Edit → Complete → Abbreviation` or keyboard shortcut `Ctrl-Alt-Space`, and the letter `L` is replaced by the string `\LaTeX`.

Abbreviation completion also supports newline `%n` and `%C` to place the cursor, if these characters are present in the expansion text. So if you have the entry

```
enl=\begin{enumerate}%n\item %C%n\end{enumerate}%n
```

in the completion file, and invoke the abbreviation completion, the expansion looks like below, where x shows the final cursor position.

```
\begin{enumerate}
\item x
\end{enumerate}
```

6.4.1 Abbreviations

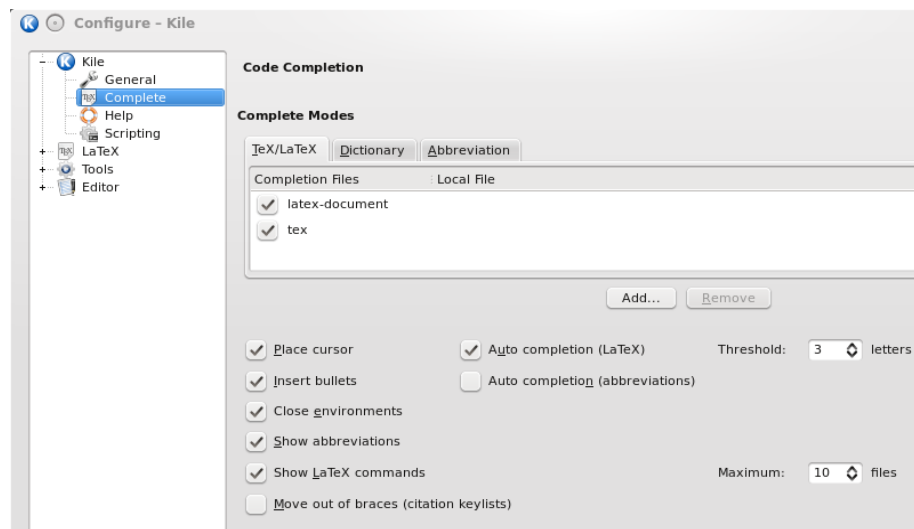
If you choose Show abbreviations, all possible abbreviations are shown in a view of the sidebar. So you will have a good survey of all possible abbreviations.

6.5 Autocompletion Modes

6.5.1 L^AT_EX Commands

You can also enable an autocompletion mode for L^AT_EX commands. When a given threshold of letters (default: 3) is entered, a popup window opens with a list of all matching L^AT_EX commands. You can select one of these commands, or ignore this window and type further letters. The entries of the completion box will always change and match your currently typed word.

Go to Settings → Configure Kile... → Kile+Complete to enable or disable this mode or to change the threshold.



6.5.2 Document Words

Large dictionaries are not useful in autocompletion mode. But, we have seen that a lot of words in a document are typed more than once. So Kile offers a completion for all words from the document that the user has already typed before. You can manually invoke this completion, if you press Ctrl-Space. Note that this mode is different from completion mode of L^AT_EX commands.

If you want to turn this mode on or off, go to Settings → Configure Kile... → Editor → Editing+Auto Completion. In this configuration dialog you can configure, if completion mode for document words should be enabled. There is also an additional autocompletion mode, where a completion box pops up, when a certain threshold is reached.

6.6 Writing Own Completion Files

The latest specification of the completion file format can found in the [CWL file format specification](#).

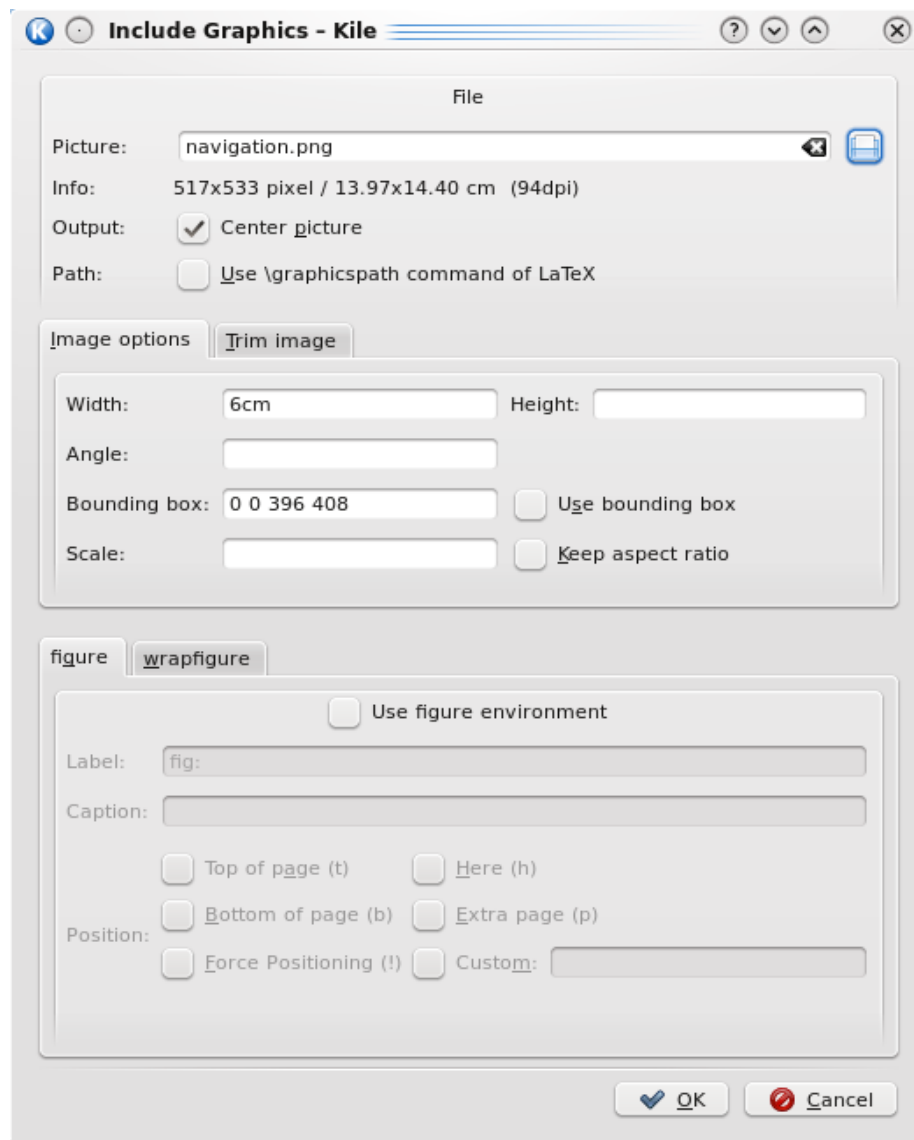
Completion files can be installed in a user's home folder under the `/.kde/share/apps/kile/complete/<mode>/` subdirectory, where *<mode>* either stands for abbreviation, dictionary or tex.

Chapter 7

Wizards and Dialogs

7.1 Include Graphics

The Include Graphics dialog makes insertion of graphics as easy as possible. Please take a look at [Section 9.3](#) and [Section 9.4](#) to get an overview of some basic facts concerning graphic formats.



1. Choose a graphics file. This can be a JPEG, PNG, EPS or even a zipped or gzipped EPS file. If you have installed [ImageMagick](#) and also configured Kile to use it (Settings → Configure Kile... → LaTeX+General), the width and the height of the graphic is automatically shown. If ImageMagick can determine a resolution, the size of the graphics is also shown in centimeters.
2. Decide whether your image shall be centered on the page.
3. Choose either traditional L^AT_EX or PDFL^AT_EX. Please remember that PDFL^AT_EX can also create DVI output, not only PDF.

4. You can choose whether the filename should be taken relative. This is the preferred way, when you use the `\graphicspath` command. By default graphics files have to be in the same folder than your master document. However it is possible to put them in other folders to make things tidier. Without a `\graphicspath` command, Kile would include the path for the graphics file. But if you use `\graphicspath`, like:

```
\graphicspath{{/path/to/my/graphics}{other/path/to/ ↵
more/graphics}}
```

and check this option, Kile will only use the base name of the graphics file. Another example: if you set `\graphicspath` command like:

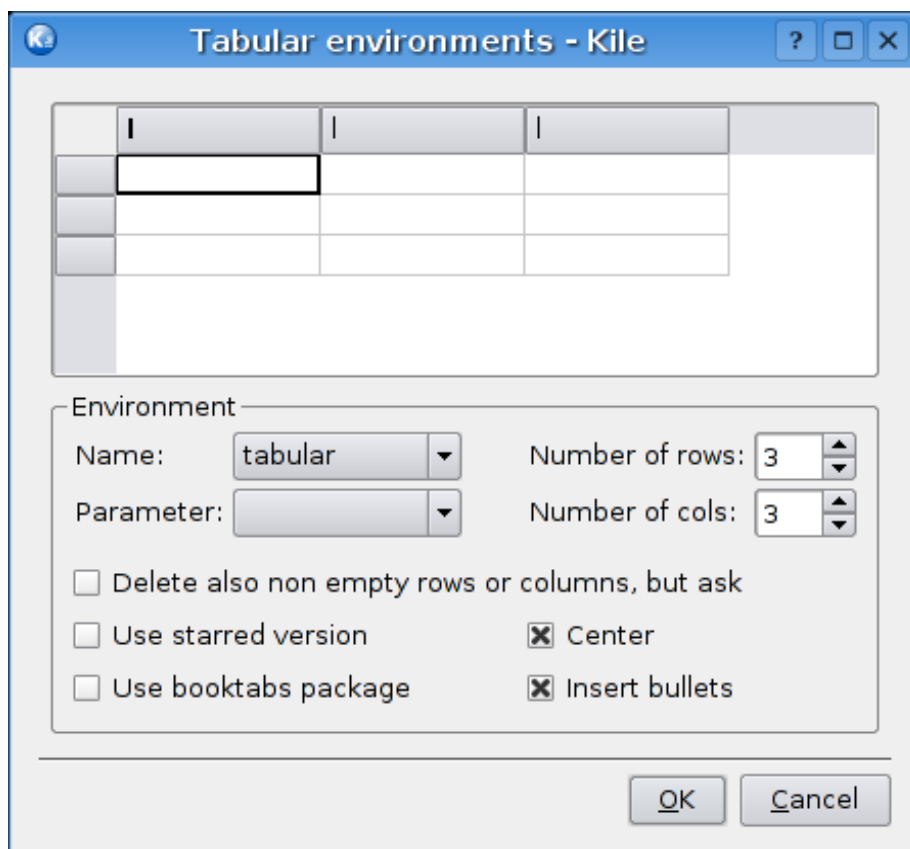
```
\graphicspath{{./}{camera/}{images/}}
```

L^AT_EX will search in the current folder, then in `camera` and finally in `images` to find your graphics file.

5. If you choose either a width or a height, the whole graphics will be proportionally scaled. If you set two values for width and height at the same time, width and height may be scaled with different factors, and this could not be what you want. See also the information near the top of dialog to know the original size of the graphics.
6. Insert an angle by which to rotate the graphics counterclockwise.
7. The bounding-box information is set automatically when you choose a graphics file. This information is only needed when you work with traditional L^AT_EX and bitmapped graphics. See the discussion of [EPS graphics](#). If L^AT_EX needs a bounding box and you don't want to generate a `bb` file, Kile supports this option. On the other side, PDFL^AT_EX will give a warning, when you want to include a `png` or `jpg` graphics with this option. This checkbox enables or disables the bounding box option.
8. Scale the image by the desired scale factor. e.g, 0.5 to reduce by half, or 2 to double.
9. The trim option will crop the imported image by `l` from the left, `b` from the bottom, `r` from the right, and `t` from the top. Where `l`, `b`, `r` and `t` are lengths.
10. Your last choice is whether to embed this graphics into a figure environment. If you decide to do so, you can also insert a caption and a label. It is a good idea to add a different prefix to each kind of label. It is common to use the prefix `fig:` for images.
11. If may prefer that some floats do not break the flow of text, but instead allow text to wrap around it, you have another choice here. To use `wrapfig`, you must first add `\usepackage{wrapfig}` to the preamble and set some of the parameter here.

7.2 Array Wizard

One of the most boring jobs one can do in L^AT_EX is to write a matrix or a tabular environment. One has to keep track of all the elements, ensure that the environment is well formed, and that all things are where they are supposed to be. Good indentation helps, but there is a simpler way: using Kile's Wizard menu. It contains Tabular and Array (used in math environments). You will then have a matrix-formed input form that you can easily fill in with your entries. This dialog also offers some options to typeset the tabular material.

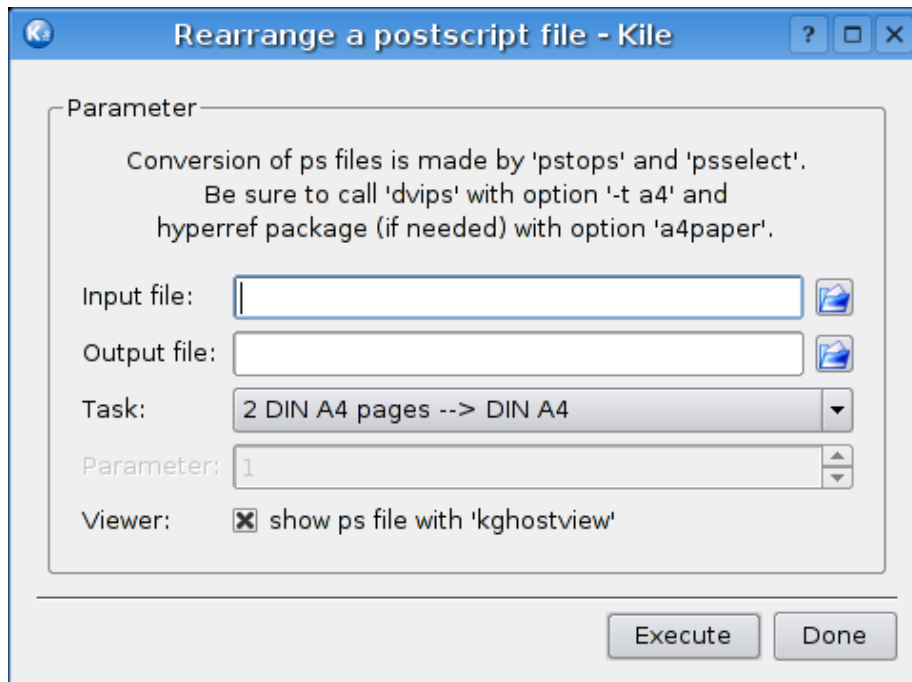


The Tabbing option will display a simpler menu to set up a tabbing environment. In all these environments, you can easily set the number of rows and columns, along with other specific options.

7.3 PostScript® Utilities

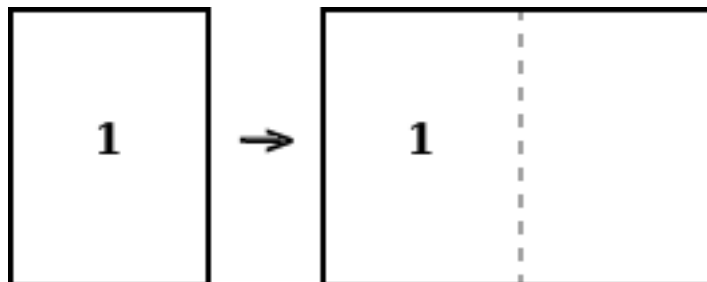
PS files are not so popular as PDF files, but are an excellent base for manipulations and rearrangements of pages. If you need PDF output, you can rearrange

pages with some PostScript® utilities and then convert it to PDF with **ps2pdf**. The *PostScript® Wizard* will suggest the most popular rearrangements, but you are free to do your own choice. Work is done by the programs **pstops** and **psselect**, which you will find in most distributions in the package `psutils`. If one of these programs is not available, the corresponding item will not be visible.

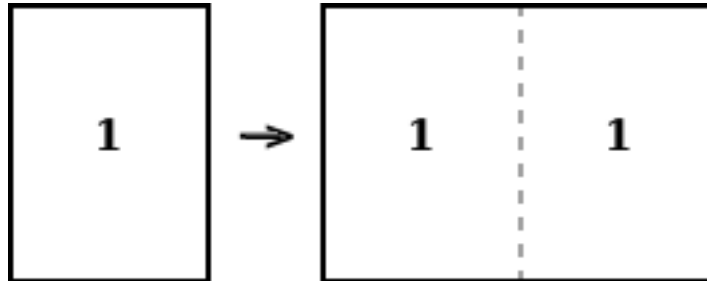


First choose your input file. If Kile finds a PS file corresponding to your current master document, it is already filled in as input file, but you are also free to choose another file. Then choose an output file, and select one of the tasks. Finally, you have to decide whether you want to do the conversion only, or also invoke Okular to view the result.

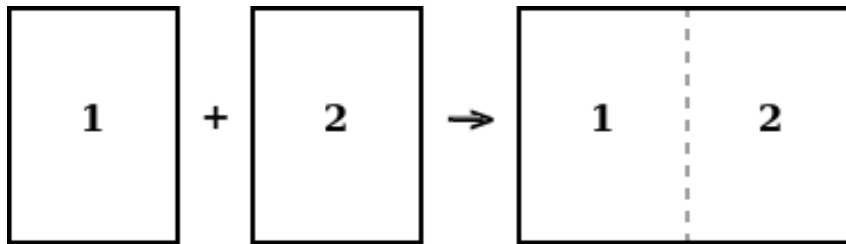
1 A5 page + empty page --> A4 Combine one A5 page together with one empty page on one A4 page. Whenever two A5 pages are combined together, they are rotated 90 degrees and will be arranged on an A4 page in landscape mode.



1 A5 page + duplicate --> A4 Put one A5 page and a duplicate together on one A4 page.

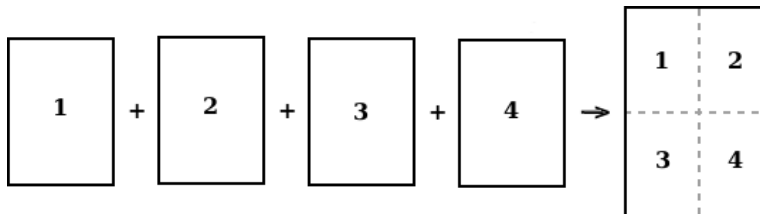


2 A5 pages --> A4 Put two consecutive A5 pages together on one A4 page.



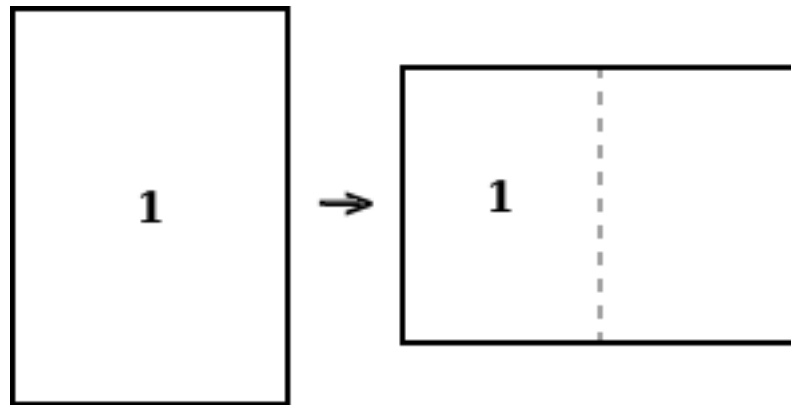
2 A5L pages --> A4 Put two consecutive A5 pages in landscape mode together on one A4 page.

4 A5 pages --> A4 Combine four consecutive A5 pages together on one A4 page. The A5 pages have to be scaled with factor 0.7 to fit on the page.

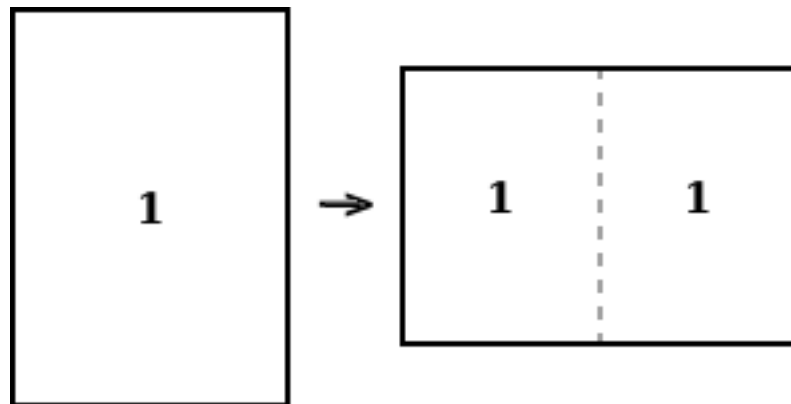


1 A4 page + empty page --> A4 Combine one A4 page together with one empty page on one A4 page. Whenever two A4 pages are combined together on one resulting A4 page, they have to be scaled with factor 0.7 and will be arranged in portrait mode.

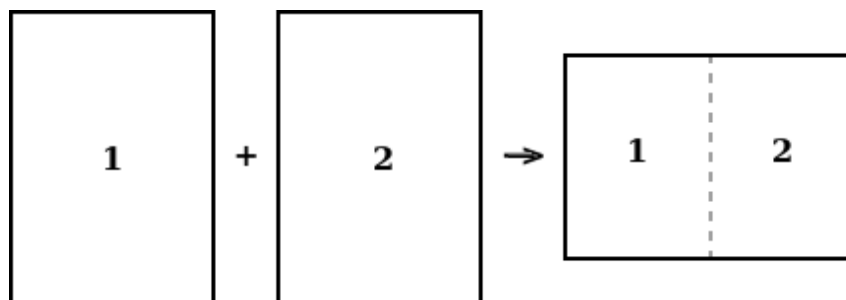
The Kile Handbook



1 A4 page + duplicate --> A4 Put one A4 page and a duplicate together on one A4 page.



2 A4 pages --> A4 Put two consecutive A4 pages together on one A4 page.



2 A4L pages --> A4 Put two consecutive A4 pages in landscape mode together on one A4 page.

select even pages Select all even pages of a document.

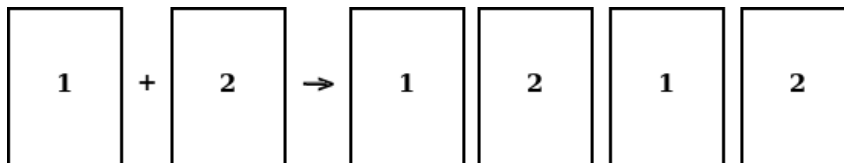
select odd pages Select all odd pages of a document.

select even pages (reverse order) Select all even pages of a document and reverse the order.

select odd pages (reverse order) Select all even pages of a document and reverse the order.

reverse all pages Reverse all pages of a document.

copy all pages (sorted) Copy all pages of a document. You have to choose the number of sorted copies.



copy all pages (unsorted) Copy all pages of a document. You have to choose the number of non-sorted copies.



pstops: choose parameter There are many options for PostScript® utilities **pstops** and **psselect**. If you need a very special one, you can invoke **pstops** with an option of your choice. Please read the manual for all possible options.

psselect: choose parameter You can invoke **psselect** with an option of your choice. Please read the manual for all possible options.

7.4 Document Statistics

The statistics dialog gives you an statistical overview for a selection, a document or an whole project. It includes the number of words, L^AT_EX commands/environments and also includes the number of characters for each type. The statistical numbers can be copied as text or as a nice formatted L^AT_EX tabular to the clipboard. If you want to get statistics for the whole project you can use File → Open All Project Files for an easy and quick way to open all source files of your project.

A note of caution has to be sounded about the accuracy of the numbers. We have included some logic to get a good estimate, e.g. K\"uhler gives one word and one command, with six resp. two characters. But there are other combinations in which parts of commands are counted as words and vice versa. It is also implied that the algorithm was developed and tested for languages similar to English or German. So don't take the numbers for granted. If you have to make an report with an exact numbers of words or characters, make some tests to check whether Kile's accuracy satisfies your needs.

Chapter 8

Special Tags in L^AT_EX

8.1 Using the L^AT_EX Tag Library

L^AT_EX has thousands of tags for symbols and special characters. The easiest way to insert these tags is to use the sidebar menu, left of the editor window.

Most Frequently Used	 Open File  Files and Projects  Structure  Symbols  Scripts  Abbreviation
≡ Relation	
⊃ Operators	
↑↑ Arrows	
∇ Miscellaneous Math	
☐ Miscellaneous Text	
⟨⟩ Delimiters	
λ Greek	
α β γ δ ϵ ε ζ η θ ϑ ι κ λ μ ν ξ $\omicron$ π ϖ ρ ϱ σ ς τ υ ϕ φ χ ψ ω $\mathcal{A}$ $\mathcal{B}$ Γ $\mathcal{I}$ Δ Δ $\mathcal{E}$ $\mathcal{Z}$ $\mathcal{H}$ Θ $\mathcal{O}$ $\mathcal{I}$ $\mathcal{K}$ $\mathcal{A}$ $\mathcal{A}$ $\mathcal{M}$ $\mathcal{N}$ Ξ Ξ $\mathcal{O}$ Π $\mathcal{H}$ $\mathcal{P}$ Σ $\mathcal{S}$ $\mathcal{T}$ Υ $\mathcal{Y}$ Φ $\mathcal{P}$ $\mathcal{X}$ Ψ $\mathcal{P}$ Ω $\mathcal{Q}$	
∅ Special Characters	
Ж Cyrillic Characters	
♪ User Defined	

The Kile Handbook

The following types are available:

- Most Frequently Used
- Relation
- Operators
- Arrows
- Miscellaneous Math
- Miscellaneous Text
- Delimiters
- Greek
- Special Characters
- Cyrillic Characters
- User Defined

The tooltips of the icons show the L^AT_EX commands and additionally needed packages.

Pressing Shift and clicking a symbol will result in `\symbolcmd` being inserted. Similar pressing Ctrl inserts it in curly brackets.

If you insert a command which requires a package which is not included in your L^AT_EX document, you will see a warning message in the logview window.

The first list of symbols holds the Most Frequently Used symbols. Inserted symbols will be added to this list, for quick and easy reference. The ordering of the symbols will not be changed upon addition of new symbols, instead a reference counter is incremented. If the number of items would exceed 30 items, the item with the lowest count will get removed.

The User Defined symbol list can hold your own symbols. To create your own symbols you need the program `gesymb` and the file `definitions.tex` from the kile source package. Additionally you need a L^AT_EX compiler (what a surprise) and [dvipng](#) (version 1.7 or later). The procedure is so that you create a L^AT_EX file with `\input{definitions}`, which makes the commands listed below available, and let `gesymb mysymbols.tex user` (which calls L^AT_EX and `dvipng`) create the icons. After copying them to `$HOME/.kde/share/apps/kile/mathsymbols/user/` and restarting kile you can use your own symbols.

The following commands are defined in `definitions.tex`:

- `\command[\optarg]{\symbol}`: Include the symbol `\symbol` in the symbol list, the optional argument `\optarg` specifies the command which kile should insert. If it is not given the command in the mandatory argument is used.
- `\mathcommand[\optarg]{\symbol}`: Same as above, except that the command in the mandatory argument is inserted in math mode.

- `\pkgsg[arg]{pkg}`: Declare that the command given in this line needs the L^AT_EX package `pkg` with the optional argument `arg`. This command has to be in front of the `\command` command and overrides any package specification by the `neededpkgs` environment.
- `\begin{neededpkgs}[pkgs-args]{pkgs} ... \end{neededpkgs}`: Has the same effect as above, but for all enclosed commands.

An example for completeness is given here:

```
\documentclass[a4paper,10pt]{article}
\usepackage{amssymb}
\input{definitions}
%
\begin{document}
\pagestyle{empty}
%
\begin{neededpkgs}{amssymb}
\mathcommand{\surd}
\pkgsg{amsmath}\mathcommand[\dddot{}]{\dddot{a}}
\mathcommand{\angle}
\end{neededpkgs}
\command{"A}
\mathcommand{\exists}
\mathcommand{\stackrel{}{}}{\stackrel{abc}{=}}

%\begin{neededpkgs}[russian,koi8-r,T2C,]{babel,inputenc, ←
fontenc,mathtext}
%
% \end{neededpkgs}
% this would need to include the packages
% \usepackage{mathtext}
% \usepackage[T2C]{fontenc}
% \usepackage[russian]{babel}
% \usepackage[koi8-r]{inputenc}
% just to explain the format
\end{document}
```

8.2 Using Bibitems

`\bibitem` is a command used to enter a reference in a thebibliography environment in your document. The syntax for using `\bibitem` is `\bibitem[label]{key}`.

The optional `[label]` is for you to add your own labeling system for the bibliography entry. If no label is set, the entries will be set in numerical order: [1], [2], [3], etc.

The argument `{key}` is used to reference and link the commands `\bibitem` and `\cite` to each other and the information they contain. The command `\cite`

contains the label associated with the intended `\bibitem`, which is located inside a `thebibliography` environment, and contains the reference data. Both corresponding `\bibitem` and `\cite` must have the same `{key}`; the easiest way to organize keys is by the author's last name. The secondary braces in the `thebibliography` environment denote the longest bibliography label you expect to have. So, inserting `{foo}` means you can have any label shorter or as large as the expression `foo`. Failure to set this parameter correctly may result in a not so attractive indentation of your bibliography.

The bibliography is a section apart from your main document, and an example of code for the bibliography would look like the following:

```
\begin{thebibliography}{50}
\bibitem{Simpson} Homer J. Simpson. \textsl{Mmmmm...} ↔
    donuts}.
    Evergreen Terrace Printing Co., Springfield, SomewhereUSA ↔
    , 1998
\end{thebibliography}
```

Then, your main source code would contain the location of the information relating to the `\bibitem` using `\cite`. That source code would look similar to this:

```
My thesis, about the philosophy of The Simpsons\copyright
comes from my favorite book \cite{Simpson}.
```

As it is often difficult to remember the exact citation key once you have many references, Kile provides an easy way to insert a citation. On the Edit toolbar click on the second drop-down box (usually it reads label) and select cite. A list with all the citation keys pops up: select the correct reference and a citation will be inserted into your document. To update the list of keys, either save the file, or Edit → Refresh Structure, or press **F12**.

The final product in your document's bibliography would then look like this:

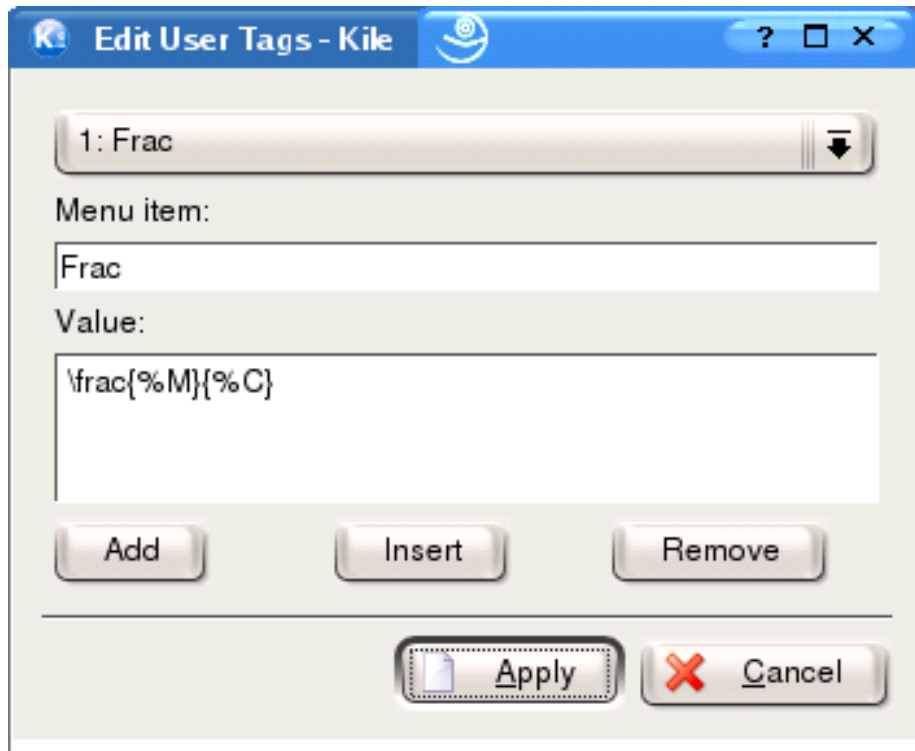
```
[1] Homer J. Simpson. Mmmmm...donuts. Evergreen Terrace Printing Co., Springfield, SomewhereUSA, 1998.
```

The easiest way to work with `\bibitem` and `\cite` is to use the toolbar drop-down box marked cite. When you select a citation to insert, you will be given the list of `\bibitem`s you have created so far, and will be able to select the reference from the list. Kile can also work together with Bib_{TEX} editor applications, such as KBib_{TEX} to help make citations easier.

8.3 User-Defined Tags

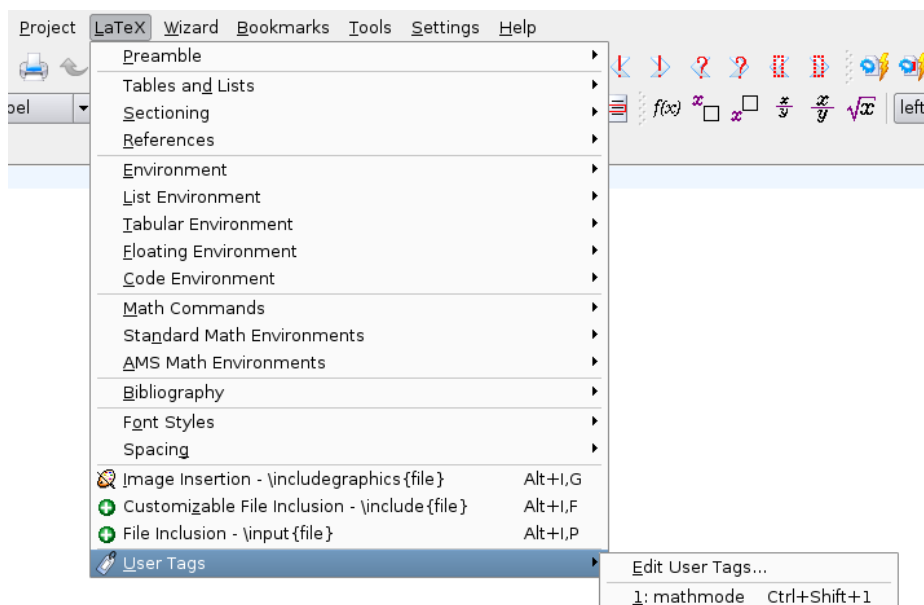
Kile gives you the ability to make your own tags. A tag is similar to a shortcut that launches some command or writes frequently-used text. For example, Joe Sixpack uses often the sentences I prefer `\LaTeX` to `\TeX` and What would I do without Linux®?. To create user-defined tags to write these sentences,

he would access LaTeX → User Tags → Edit User Tags...; this will present him a dialog where he can create his own user-defined tags.



The Edit User Tags Dialog

The Kile Handbook



Invoking a User Defined Tag

He would probably give each tag a name that can clearly identify it. The name you give your tag is entered in the section marked Menu item, and the text of frequently-used command should be entered into the section labeled Value. Once the commands are entered, he can use them quickly using the shortcut Ctrl-Shift-1 for the first tag to enter I prefer \LaTeX\ to \TeX\ and Ctrl-Shift-2 to enter What would I do without Linux®?.

8.3.1 Placeholders in User-Defined Tags

There are some placeholders you can use in user-defined tags: they are %B, %C, %M and %S.

- %B: will be replaced by a bullet.
- %C: this is where the cursor will be placed after the insertion of a user-defined tag.
- %M: this stands for marked text; the selected text is inserted in its place when inserting user-defined tags.
- %S: will be replaced by the source file's name without file extension.

To show you how this works let's say for example that we have a user-defined tag, which contains the value `\bfseries{%M}%C`, and I have a selection of text

highlighted in my document that we want to turn into bold text. So, we highlight the phrase `I love Fridays`, apply our user-defined tag by pressing Ctrl-Shift-1, and we get the text `\bfseries{I love Fridays}`, with the cursor placed at the end of the text.

Chapter 9

The Build Tools

9.1 Compiling, converting and viewing

To view the result of your work, you first need to compile the source. All the build tools are grouped closely together in the Build → Compile, Build → Convert, and Build → View menus.

To compile your source code for screen viewers like Okular or further conversion, you can use the shortcut Alt-2. Then you can view the DVI file using your default viewer with Alt-3, convert the DVI to a PS file with Alt-4, and view the PS file with Alt-5.

9.1.1 BibT_EX

If you are using [BibT_EX](#) for your bibliography entries, you usually have to follow a special compiling scheme. This means calling L^AT_EX and then BibT_EX and then L^AT_EX twice again. Fortunately Kile is clever enough to detect automatically if it is necessary to call additional tools like BibT_EX, makeidx and Asymptote. This logic is by default turned on and can be changed in Settings → Configure Kile... → Tools+Build in the General tab in the L^AT_EX and PDFL^AT_EX tools.

9.1.2 makeidx

If you are using the [makeidx](#) package to make a final, alphabetical index for your document, you have also to follow a certain compilation pattern or let Kile do this for you in the same way as with BibT_EX files.

9.1.3 MetaPost and Asymptote

If you want to compile your document with MetaPost or Asymptote, picture drawing programs, you can do it with Build → Compile → Metapost, or Build → Compile → Asymptote.

9.1.4 PDFL^AT_EX

There is also another way to compile your document, if you want a PDF: you can run PDFL^AT_EX, that will compile the source directly into a PDF file, with Alt-6: you can then view the compiled file pressing Alt-7.

Alternatively, you can convert a PS into a PDF with Alt-8, or directly a DVI into a PDF with Alt-9.

Using PDFL^AT_EX instead of L^AT_EX may be just a matter of simplicity or habit, but sometimes the behavior of the two program can differ.

9.1.5 L^AT_EX to Web

Finally, you may want to publish your work on the web and not just on paper. You may then use the latex2html program, that can be called from Kile's menu Build → Convert → LaTeX to Web. The result will be placed in a subfolder of the work folder, and you will be able to see the result of the conversion choosing the menu item Build → View → View HTML.

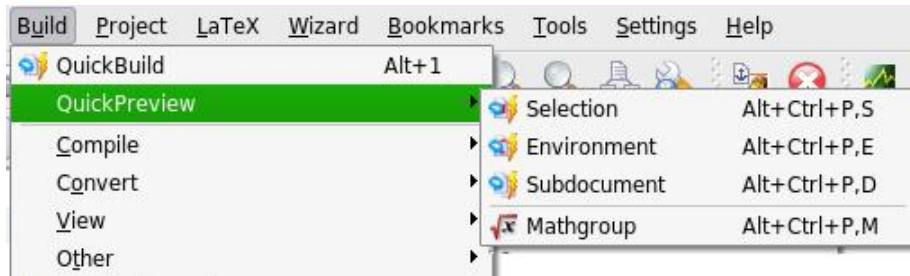
9.1.6 Passing Command-Line Parameters

If you want to pass some specific command-line parameters to the compile, convert or view tools, you can configure their call in Settings → Configure Kile... → Tools+Build.

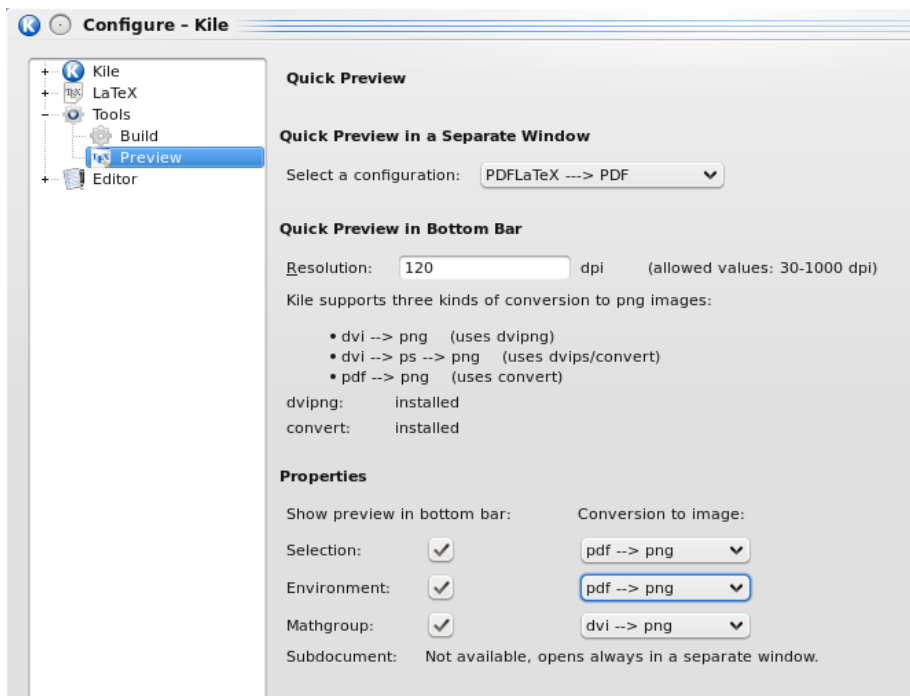
9.2 Quick Preview

You will always need some time to view the result, when working with L^AT_EX. L^AT_EX has to compile the source and the viewer has to be called. This can be annoying if you only changed some letters in an equation difficult to typeset. Kile offers a *Quick Preview* mode, where you can compile only a part of a document and save a lot of time. It supports four different modes, which can be combined with seven configurations.

The Kile Handbook



All settings must be done in Settings → Configure Kile... → Tools+Preview.



9.2.1 Selection Mode

The user has to select a part of the document. Menu entry Build → QuickPreview → Selection or the keyboard shortcut Ctrl+Alt+P,S will start the selected programs. Kile takes the preamble of the original text, so that all packages and user defined commands are included. The user can choose one of seven predefined configurations:

- LaTeX+DVI (embedded viewer)
- LaTeX+DVI (Okular)
- LaTeX+PS (embedded viewer)

- LaTeX+PS (Okular)
- PDFLaTeX+PDF (embedded viewer)
- PDFLaTeX+PDF (Okular)
- PDFLaTeX+PDF (Okular)

This should be sufficient for all situations for which a quick preview is needed.

9.2.2 Environment Mode

Very often you want to preview the current environment, and especially mathematical environments, which sometimes may be difficult to write. Kile offers a very fast way to do this. No selection is needed, only choose Build → QuickPreview → Environment or the keyboard shortcut Ctrl+Alt+P,E and the current environment will be compiled and shown.

9.2.3 Subdocument Mode

If you have a large project with a lot of documents, compiling the whole project is not a great idea, if you have made changes only in one single document. Kile is able to compile and show a preview of the current subdocument. It takes the preamble from the master document and only compiles the current part when you choose Build → QuickPreview → Subdocument or the keyboard shortcut Ctrl+Alt+P,D.

9.2.4 Mathgroup Mode

The mathgroup preview mode allows you to preview the mathgroup you are currently editing. Kile takes the preamble from the master document and only compiles the mathgroup the cursor is currently in when you choose Build → QuickPreview → Mathgroup or the keyboard shortcut Ctrl+Alt+P,M.

9.2.5 Quick Preview in Bottom Bar

Instead of showing the preview in a new document Kile can also be configured to use the bottom bar for preview compilations. You can activate this feature in the quick preview configuration panel.

9.3 Graphic File Formats

9.3.1 L^AT_EX and PDFL^AT_EX

PDFL^AT_EX, when used with `graphics` or `graphicx` packages, can compile correctly PNG and JPG files into DVI or PDF, but is not able to handle EPS files. Conversely, the process of compiling with L^AT_EX to DVI and converting to PS and eventually PDF does support EPS, but does not support PNG and JPG.

A lot of users want to create PDF documents, but also want to use of the excellent Pstricks package to create PostScript® graphics, or they want to use the PostScript® output of mathematical and scientific software like Mathematica, Maple or MuPAD. These L^AT_EX users have to compile first in PostScript®, even if they want to create PDF documents, because these programs produce PostScript® code which cannot be managed by PDFL^AT_EX. However, it is not so hard as it may sound, because Kile will help.

9.3.2 Graphics Conversion

To overcome this frustrating loop, in case you want to include both PostScript® code and PNG or JPG files, you have a number of workarounds:

- If you need a file in PS format, but have JPG or PNG graphics, you can also simply use PDFL^AT_EX with DVI output first, and then run `dvips` to create the PS file. You see that PDFL^AT_EX is a very good choice, if your source contains no PostScript® code at all.
- You can convert EPS files to PNG or other formats with utilities as the [Gimp](#) or [ImageMagick](#) and use PDFL^AT_EX.
- A preferred way is to convert EPS graphics to PDF graphics with **epstopdf**, which comes with every T_EX distribution and then use PDFL^AT_EX. It produces high quality graphics, and you can even control the result with some of the following options:

```
-dAutoFilterColorImages=false
-dAutoFilterGrayImages=false
-sColorImageFilter=FlateEncode
-sGrayImageFilter=FlateEncode
-dPDFSETTINGS=/prepress
-dUseFlateCompression=true
```

Even better: if your system allows `shell-escape`, conversion can be done on the fly. All you have to do is to include the `epstopdf` package, which is part of all T_EX distributions, with command `\usepackage{epstopdf}`. Assuming that your code is

```
\includegraphics[width=5cm]{test.eps}
```

When you call PDFL^AT_EX with option `--shell-escape`, graphics `test.eps` is automatically converted into `test.pdf`.

This conversion will take place each time you run PDFL^AT_EX. If your graphics command is given implicitly:

```
\includegraphics[width=5cm]{test}
```

epstopdf checks whether `test.pdf` is already available, so that conversion step can be skipped.

- You can convert the other way around, and use L^AT_EX and PS-PDF conversion. This is not always a good idea, since EPS encapsulation of JPG or PNG can yield larger files, that in turn yield unnecessarily large documents. This is however *highly* dependent on the graphic utility that you use, since EPS can encapsulate other graphics, but not all applications support this perfectly. Some might actually try to build your JPG image with vectors and various scripting, which will result in gigantic files. Conversion of all graphics formats to EPS can be done by [ImageMagick](#). Another simple program that does this process correctly is [jpg2ps](#).
- You can also use an automatic conversion. All graphic files are converted on the fly to EPS, and inserted into the PS document. This is a comfortable way, but you have to set up your system properly. This is discussed in the section [EPS Graphics](#).

9.3.3 Use the right File for the right Graphic

- EPS is sort of a graphic vector scripting language, describing all the lines and dots the graph is made of; it looks good even when magnified beyond its default size, and suits best diagrams and vectorial graphic natively produced in EPS, which look very clear and sharp while maintaining a very small byte size.
- PNG (or the deprecated GIF) is a *non-lossy* file format, with good compression and quality. It is very good for diagrams, scans of drawings, or anything whose sharpness you do want to retain. It is sometimes overkill when used for photos.
- JPG is a *lossy* format, that compresses files better than PNG at the price of some loss in the picture detail. This is usually irrelevant for photos, but may cause bad quality for diagrams, drawings, and may make some thin lines disappear outright; in those cases use EPS or PNG.

But always remember: garbage in, garbage out! No conversion will make a bad picture good.

9.4 EPS Graphics

EPS graphics files are the traditional way to insert graphics files into L^AT_EX documents. As mailing lists are full with questions concerning EPS graphics, we will discuss some important aspects and demonstrate how Kile supports them.

9.4.1 L^AT_EX and EPS Graphics

If you decided to use the traditional L^AT_EX to produce PS or PDF output, you will probably run into some problems with graphics. You have to use EPS graphics (Encapsulated PostScript®), no JPEG or PNG files. This should be no problem, as there are a lot of [converters](#) like **convert** from the excellent [ImageMagick](#) package. But, it needs some time of course.

The EPS files are used by both L^AT_EX and the DVI-PS converter:

- L^AT_EX scans the EPS file for the bounding box line, which tells L^AT_EX how much space to reserve for the graphics.
- The DVI-PS converter then reads the EPS file and inserts the graphics in the PS file.

This has some implications:

- L^AT_EX never reads the EPS file if the bounding-box parameters are specified in the graphics-insertion command.
- Since L^AT_EX cannot read non-ASCII files, it cannot read the bounding-box information from compressed or non-EPS graphics files.
- The EPS graphics are not included in the DVI file. Since the EPS files must be present when the DVI file is converted to PS, the EPS files must accompany DVI files whenever they are moved.

Now you can call L^AT_EX, and a DVI-PS converter like dvips to create your PostScript® document. If your goal is a PDF document, you should run **dvips** with option **-ppdf** and then call **ps2pdf**. You will find a lot of documents describing this solution.

9.4.2 The PostScript® Way of Kile

Kile helps you to get the bounding-box information. If you have installed [ImageMagick](#) package, Kile will extract this information from the EPS file and insert it as an option. This is done automatically when you select the graphics file. There are two advantages to proceed like this:

- The information is already scanned in the dialog, and need not to be done by L^AT_EX later on.
- Even more important is that the width and height of the picture can be calculated, when the its resolution is known. This information will be shown near the top of the dialog, and may serve as a clue when you want to scale the graphics.
- Kile can also support zipped or gzipped EPS files, which are much smaller than uncompressed EPS files. But, this feature can only be used with a special system setup and a change of your local graphics configuration, like it is described in the [Bitmap Graphics](#) section.

9.4.3 The PostScript® Way and Bitmap Graphics

If your system allows `shell-escape`, Kile also supports an easy way to include bitmap graphics, if you set up your T_EX system properly. There is no need to convert JPEG or PNG graphics, this can be done automatically when the DVI file is converted to PS.

L^AT_EX needs some information about the file suffixes. The package `graphicx` looks for a file `graphics.cfg`, which must be somewhere in your search path for L^AT_EX documents. Search for entries like:

```
\DeclareGraphicsRule{.pz}{eps}{.bb}{}%
\DeclareGraphicsRule{.eps.Z}{eps}{.eps.bb}{}%
\DeclareGraphicsRule{.ps.Z}{eps}{.ps.bb}{}%
\DeclareGraphicsRule{.ps.gz}{eps}{.ps.bb}{}%
\DeclareGraphicsRule{.eps.gz}{eps}{.eps.bb}{}%
```

and replace these lines with:

```
\DeclareGraphicsRule{.pz}{eps}{.bb}{}%
\DeclareGraphicsRule{.eps.Z}{eps}{.eps.bb}{}%
\DeclareGraphicsRule{.ps.Z}{eps}{.ps.bb}{}%
\DeclareGraphicsRule{.ps.gz}{eps}{.ps.bb}{}%
% changed or new graphic rules
\DeclareGraphicsRule{.eps.zip}{eps}{.eps.bb}{`unzip -p ←
#1}% zipped EPS
\DeclareGraphicsRule{.eps.gz}{eps}{.eps.bb}{`gunzip -c ←
#1}% gzipped EPS
\DeclareGraphicsRule{.jpg}{eps}{}{`convert #1 eps:-}% ←
JPEG
\DeclareGraphicsRule{.gif}{eps}{.bb}{`convert #1 eps:-}% ←
GIF
\DeclareGraphicsRule{.png}{eps}{.bb}{`convert #1 eps:-}% ←
PNG
\DeclareGraphicsRule{.tif}{eps}{.bb}{`convert #1 eps:-}% ←
TIFF
```

```
\DeclareGraphicsRule{.pdf}{eps}{.bb}{`convert #1 eps:-}% ↵
PDF-graphics
```

You will find this file, for example in Debian, at `/etc/texmf/latex/graphics-.cfg`. The best way to proceed is to copy this file to your local `texpath` and then change this file. See the yours $\text{T}_\text{E}\text{X}$ distribution manual to learn how to get a list of your $\text{T}_\text{E}\text{X}$ folders.

With this configuration file you are able to insert bitmap graphics and zipped or gzipped EPS files in $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$. The command for conversion is given by **dvips**. When you look at the conversion command you will see that no extra file is created. The result of the conversion process is directly piped into the PS file. The only thing $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$ must know is the size of the graphics, and therefore we need the bounding box, which is provided by Kile.

Some say that this way is insecure; you have to decide on how to work. In any case, you need no bounding box, as Kile will extract this information from all types of graphics.

9.4.4 PDF $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$ and EPS Graphics

As already stated, PDF $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$ is not able to handle EPS graphic files, but converters like **epstopdf** will help. The best way is to include package `epstopdf`, which must follow the `graphicx` package.

```
\usepackage[pdftex]{graphicx}
\usepackage{epstopdf}
```

Now you can already include EPS graphics, if you run **pdflatex** with option `--shell-escape`, but we can make it even better and also handle zipped or gzipped EPS files. Again we have to change the graphics configuration file `graphics.cfg` like above. This time we search for:

```
% pdfTeX is running in pdf mode
\ExecuteOptions{pdftex}%
```

and simply add some lines.

```
% pdfTeX is running in pdf mode
\ExecuteOptions{pdftex}%
\AtEndOfPackage{%
  \g@addto@macro\Gin@extensions{.eps.gz,.eps.zip}%
  \@namedef{Gin@rule@.eps.gz}#1{{pdf}}{.pdf}{`gunzip -c #1 ↵
    | epstopdf -f >\Gin@base.pdf}}%
  \@namedef{Gin@rule@.eps.zip}#1{{pdf}}{.pdf}{`unzip -p #1 ↵
    | epstopdf -f >\Gin@base.pdf}}%
}%
```

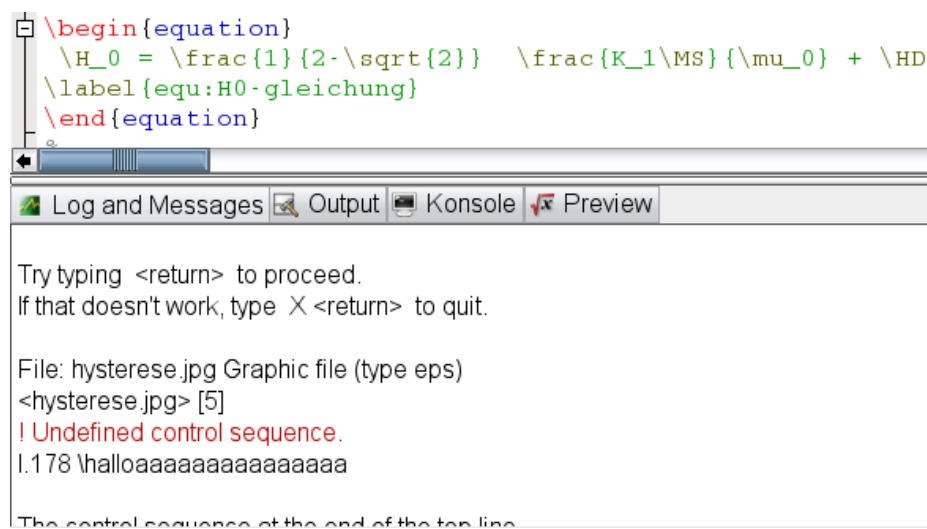
With these lines, PDF $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$ is able to handle EPS files, and hopefully there should be no more issues concerning graphics.

9.5 Master Document

Defining your document as a master allows you to work with separated files, which gives you a parent document (or Master document), and child documents that make up a complete work. After having defined your Master document, with the corresponding command in the Settings menu, all the commands of the Tools menu will apply only to this document, even when you are working on the child documents. You can even close the Master document.

9.6 Error Handling

After you have compiled something, Kile takes a look at the error messages that were generated. If there are any errors or warnings, they will be briefly reported in the Log and Messages window. One can take a closer look at the messages by selecting Build → View Log File, or by using the keyboard shortcut Alt-0. The generated log is then displayed in the Log and Messages view; errors and warnings are highlighted.



Viewing the log

You can easily jump from one message in the log file to another by using the Build → Next / Previous LaTeX Error / Warning menu items, or by using the corresponding toolbar buttons.

To jump to the line in the L^AT_EX source where the error or warning occurred, click on the error or warning in the Log and Messages view. Kile will take you automatically to the offending line.

9.7 The Watch File Mode

When you launch the Quickbuild command, a viewer of some sort will normally be called after the compilation. If you are not using an embedded viewer, a new window will be opened every time.

If you are adjusting the look of your document, you might launch Quickbuild very often, and have many viewer windows open on your desktop; to avoid this confusion, you can activate the Watch file mode, that will prevent Quickbuild from launching a viewer.

This mode is of course useless with the embedded viewers, as you have to close them anyway to get back to editing the document and recompiling.

Chapter 10

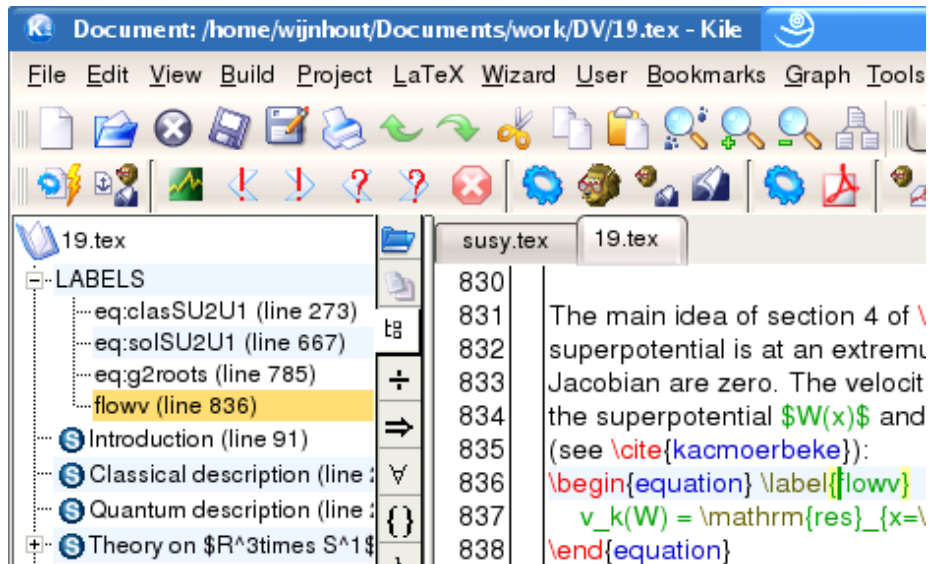
Navigating the L^AT_EX Source

10.1 Using the Structure View

The Structure view shows the hierarchy of the document being created in Kile, and allows you to quickly navigate it, showing its segmentation. To navigate around your document, all you need to do is to left click on any label, chapter, section, subsection, etc., and you will be taken to the beginning of the corresponding area.

If you included a separate L^AT_EX file in your source using the `\input` or `\include` tags, these files will be referred to in the Structure view; double-clicking on their names in it will make Kile bring up the included file in the editor window.

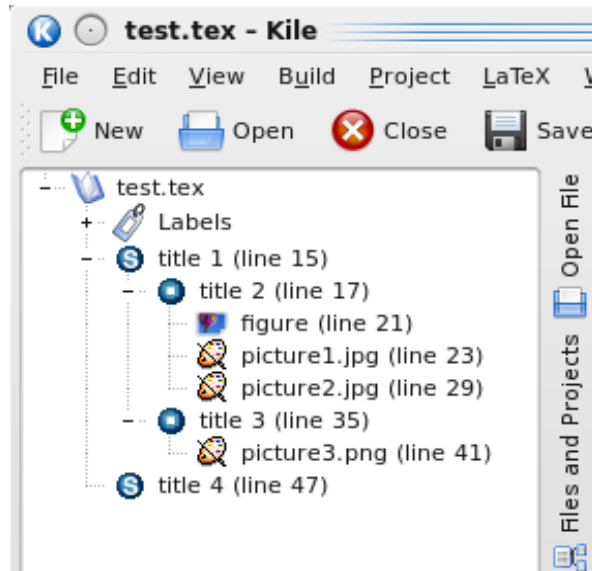
The hierarchy tree has also a separate branch for labels used in the text.



Using the Structure View

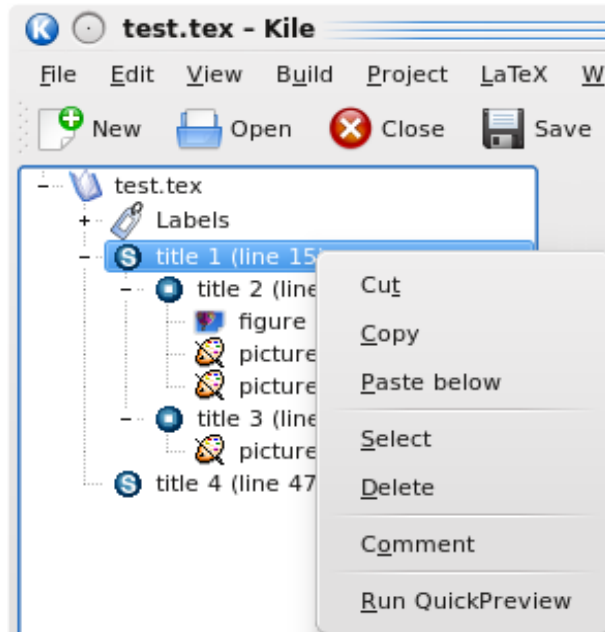
10.1.1 Using the Context Menu

Most of the entries in the structure view have a lot of entries in the context menu, which you can open with a right mouse click. So look at the structure view in the following picture.

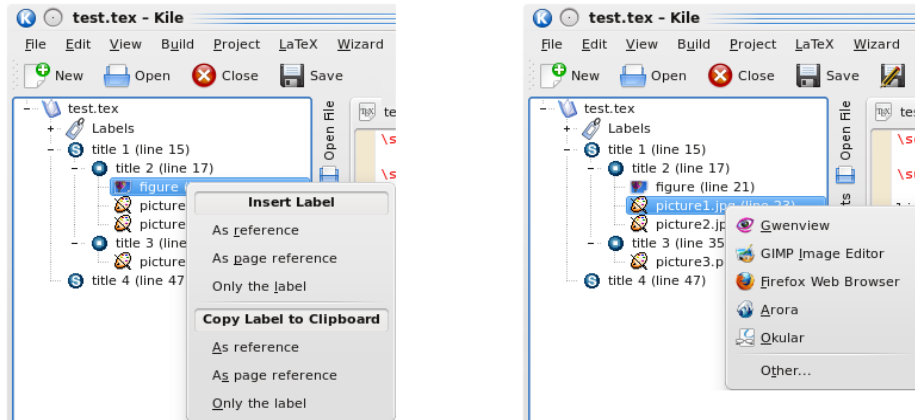


You will find some labels, some sectioning entries, one figure environment and three pictures. If you right click on one of the sectioning entries, a menu with

some useful commands will popup. All commands like Select, Delete or Comment will work with all lines, which belong to this section.



Clicking on a figure or table entry will offer some actions with respect to references and a right click on a graphics entry will offer some programs to open the graphics.



10.1.2 Updating the Structure View

To update your structure view you can either go to Edit → Refresh Structure, hit F12, or you can save your document, which will make Kile update its Structure view.

10.2 Bookmarks

Bookmarks are your reference to a segment of text or a line inside the Kile environment. To use a bookmark, select a specific line of your document you would like to return to, then press Ctrl-B, and Kile will add a bookmark to this line. Alternatively, you can also set a bookmark by highlighting a line and choosing the menu labeled Bookmarks → Set Bookmark.

To remove all your bookmarks, select Bookmarks → Clear All Bookmarks.

Please note that currently the bookmarks are not saved after exiting Kile.

Chapter 11

Projects

11.1 Working with Projects

In Kile you can create and work with *projects*. A project is a group of L^AT_EX, graphic, BibT_EX, or other files that contain all the information that is used to build your complete document. A typical project would be a document consisting of several chapters, written in different `.tex` files; all of them could be included in a project, to make the whole document easier to manage. The specifications of the project are stored in a special file, with extension `.kilepr`.

A Project adds the following functionalities:

- You need not set a master document, Kile does this automatically.
- Project files can easily be archived together by going to Build → Other → Archive
- The Files and Projects view shows which files are included in the project.
- After opening a project, any file that was previously opened will be restored with the original encoding and highlighting.
- Code completion works across all project files.
- Reference completion works across all project files.
- Citation completion works across all project files.
- Search in all project files.
- Specify custom quickbuild and makeidx command.

11.2 Creating a Project

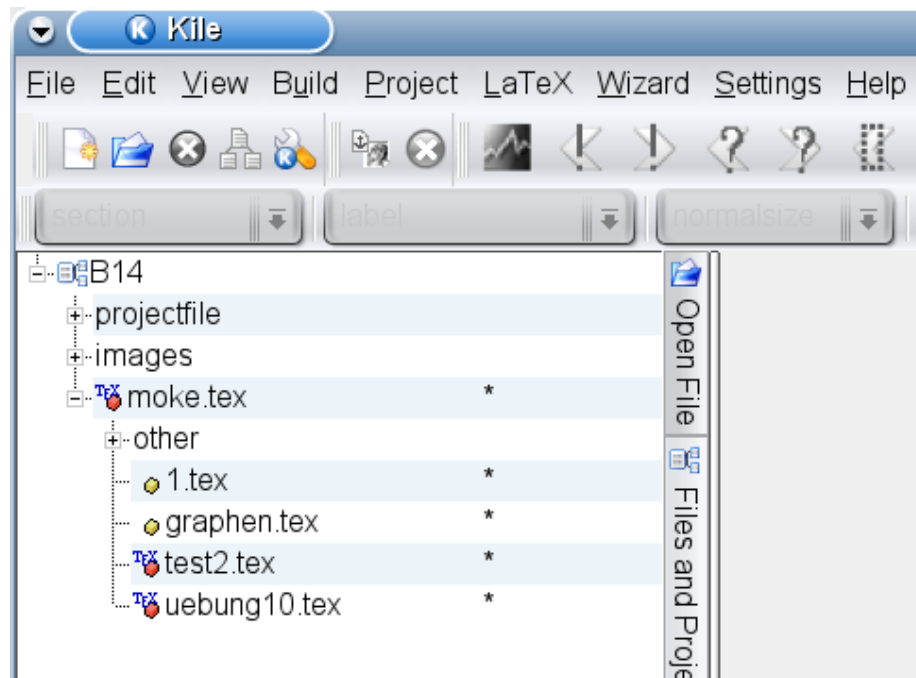
To create a project, select Project → New Project.... You will be asked to give the following information to create your project:

- Title of your project.
- Name of the project file with `.kilepr` extension.
- Filename.
- Type of file creating: Empty Document, Article, Book, Letter, Report,

When you fill out the filename box, you have to include a relative path from where the `.kilepr` project file is stored to the file.

11.3 The Files and Projects View

The Files and Projects view is a button of the sidebar menu. From this view, you can see the structure of your project, its files, and the name of the `.kilepr` file that stores the project information. Adding, removing, or changing options in your project is done via the Files and Projects view.

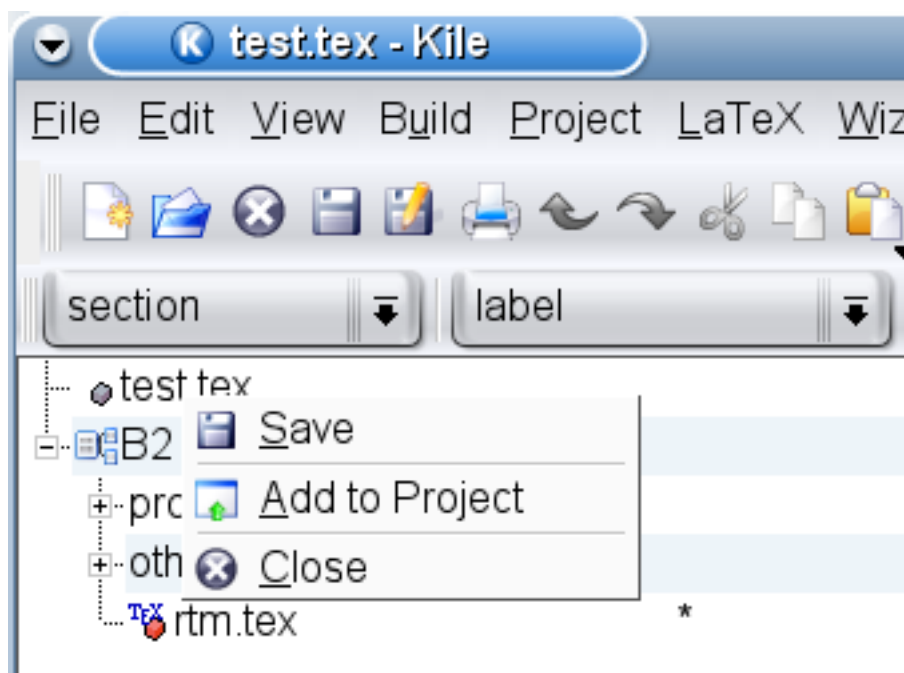


The Files and Projects View

11.4 Adding and Removing Files

To add a file to your project, open any T_EX file, right click on its name in the Files and Projects view, and select Add to Project. If you have multiple projects open, a dialog box will ask you which project the file has to be added to.

If you have multiple files to be added to a project, you can select the project from the Files and Projects view and right-click then select Add Files...; you will then be able to select your files in a dialog box.



Adding a file to a project

To remove a file from a project, right-click on it and select Remove File. This does *not* delete your file (and also does not close it), but only removes it from the list of files contained in the .kilepr extension.

11.5 Project Options

Kile has a few options related to your project that can be set. To change them, right-click on the title of your project and select Project Options, and you will have the option of changing:

- The title of your project.
- The Master document.

- The Quickbuild command.
- The makeidx options.

11.5.1 Archiving your Project

Kile allows you to easily backup your project by storing all its files into a single archive (often known as *tarball*). To archive your project right-click on its name in the Files and Projects view, or select Build → Other → Archive.

By default, all files in a project are added to the archive. If you do not want to include a certain file in the archive, right-click on it in the Files and Projects view, and uncheck the Include in Archive option.

Archive commands are simple shell commands that are executed from the project folder (where the `.kilepr` file is located).

11.6 Closing a Project

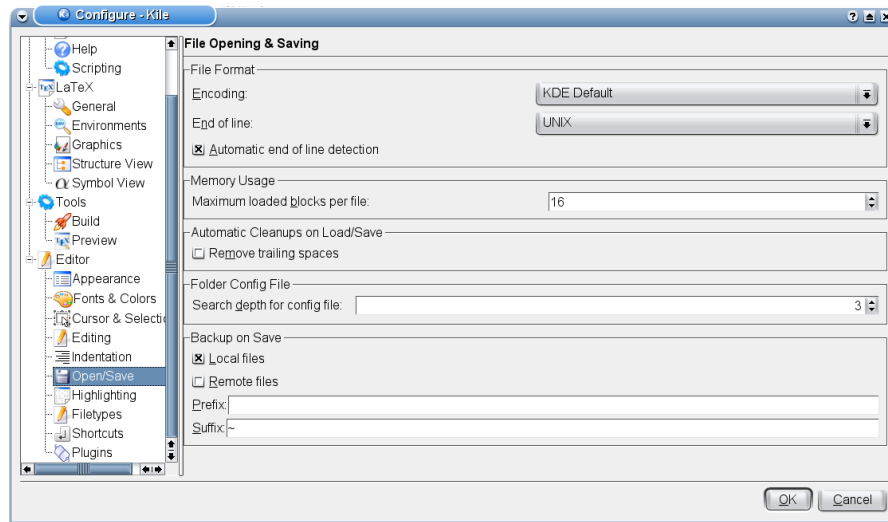
To close a project, select the Files and Projects view from the vertical toolbar, right click on your project title, and then select Close. This will close your project, all the files associated with your project, and will also add the name of the project you just closed to Open Recent Project... in the Project menu.

Chapter 12

Document Encoding

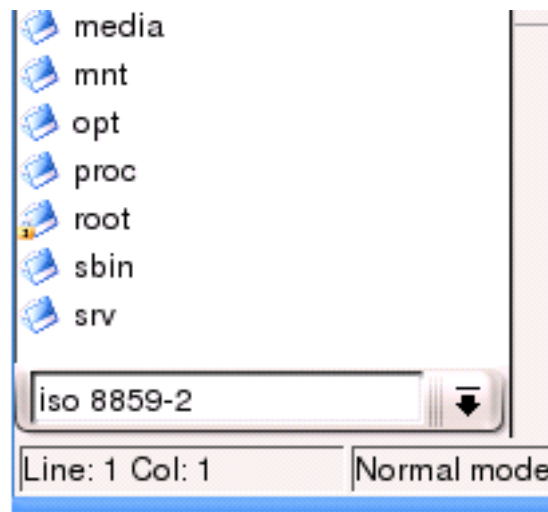
The Kile editor allows you to read, convert and set the text to the encoding your document needs. This allows you to use non-standard letters and symbols; you can use, for example, accented characters for Italian or French. Selecting the encoding for your document can be done in three ways:

- One way to set the encoding is to use the Set encoding combo box, located at the bottom of the Open File sidebar view.
- Another way is using the submenu Settings → Configure Kile... → Editor, where you can set the default character encoding for all files.



- A third way to set the encoding for a document is to set the option when you use the wizard to create a new document.

L^AT_EX itself understands only ASCII, a very limited set of characters, so you could not use accented or special letters directly. To use accented letters, a special syntax was created: such as for example `\"e` for `ë`. There is a package to help you with this, called `inputenc`, and is included in the preamble using `\usepackage[latin1]{inputenc}`, where the optional argument is the encoding you would like to use (nowadays in most cases `utf8x`). This tells L^AT_EX to translate all of the `ë`'s you wrote to `\"e`'s before compiling. Please refer to the `inputenc` documents directly for more information on `inputenc`. Last but not least: remember to make sure that your file is *actually* encoded in the same encoding you told `inputenc`!



Choosing the source file's encoding

This host of different character coding tables has been creating problems on many applications: for example, you cannot write a course of Turkish in French without losing one language's special characters. There is general agreement that, sooner or later, everybody will switch to [Unicode](#). There are many implementations of Unicode, and UTF-8 is the most successful in Linux®; Windows®(R) relies instead on the more cumbersome and less flexible UCS-2. Some distributions, as RedHat, have already begun setting their default encoding to UTF-8, and therefore you may be very interested in using of the `utf8x` argument to the `inputenc` package.

12.1 The ucs Package

If you don't have the `ucs` package installed, you can proceed as follows:

- Get the `ucs` package from the home page of [Unicode support for L^AT_EX](#), by Dominique Unruh from the University of Karlsruhe.

- To install it, unpack the downloaded file and place it in a folder listed in your `$TEXINPUTS` environment variable. This can also be set inside kile.

```
\usepackage{ucs}
\usepackage[utf8x]{inputenc}
```

12.2 XeLaTeX

If you are using XeLaTeX you can simply load the `xltxtra` package. It will load all required packages.

```
\usepackage{xltxtra}
```

12.3 CJK Support

Adding support for ideographic languages is quite tricky. However, once you are done with it, it will work quite well. Other than installing packages, there is some extra configuration work to do.

TIP

Your Linux® distribution might already have a CJK (Chinese, Japanese, Korean) package ready for you, so you might be saved the hassle of manually installing everything. Do check before going forward!

There is the possibility of using the `ucs` package in order to write short snippets of CJK text, but that option is seriously limited as it does not handle, among other things, newlines. We will instead install the complete CJK- $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ package and make it work for both $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ and $\text{PDFL}^{\text{A}}\text{T}_{\text{E}}\text{X}$. A lot of this material has been inspired by [Pai H. Chou's page about how to setup PDFLATEX](#).

1. Download the [CJK](#) package. Copy its unpacked files to an appropriate subfolder of `$TEXMF`, just like you did with the `ucs` package before (see [Section 12.1](#)). The files will be unpacked in a `CJK/X_Y.Z` folder, it is not important that you take them out, though it will probably be tidier for you to maintain.
2. Now you have to download a font that supports all the CJK characters you need. You can choose any `*.ttf` file that covers them, but in this walkthrough we will use [Cyberbit](#). Unzip the file and rename `Cyberbit-.ttf` to `cyberbit.ttf`, since uppercase might confuse your system.

Place `cyberbit.ttf` in a folder together with [Unicode.sfd](#), and generate the `*.tfm` and `*.enc` files with the command `$ ttf2tfm cyberbit.ttf`

-w cyberbit@Unicode@. For some reasons, sometimes this does not produce the hundreds of files it should. Should that be your case, you can download both [*.tfm](#) and [*.enc](#) files.

Place the [*.tfm](#) files in an appropriate folder, say `$TEXMF/fonts/tfm/bitstream/cyberbit/`; the [*.enc](#) files may be installed in `$TEXMF/pdftex/enc/cyberbit/`.

3. Now we need a map file to connect the [*.enc](#) files to the font. Download [cyberbit.map](#) and install it in `$TEXMF/pdftex/config/`.
4. Download another file, [c70cyberbit.fd](#), and place it in an appropriate folder. You may choose, for example, `$TEXMF/tex/misc/`.
5. The last file we have to generate is a PostScript® Type 1 font, necessary to read DVI files generated with L^AT_EX. Run the command `$ ttf2pfb cyberbit.ttf -o cyberbit.pfb`, and copy the resulting `cyberbit.pfb` in a folder like `$TEXMF/fonts/type1/cyberbit/`.
6. Let's now place `cyberbit.ttf` among the fonts where L^AT_EX can find it. You could place it in a folder named `$TEXMF/fonts/truetype/`.
7. Check the configuration file you find at `$TEXMF/web2c/texmf.cnf`, and make sure that the line mentioning `TTFONTS` is uncommented and points to the folder where you saved `cyberbit.ttf`.
8. To make it possible for PDFL^AT_EX to use your CJK fonts, it is necessary that you add a line in configuration file `$TEXMF/pdftex/config/pdftex.cfg`. Add `map +cyberbit.map` in the file to complete the configuration for PDFL^AT_EX.
9. To configure L^AT_EX so that you can produce DVI files with CJK characters, you have to add a line in file `ttfonts.map`. The file might be in a folder named `$TEXMF/ttf2pk/`, but you will probably have to look for it. Add the line `cyberbit@Unicode@ cyberbit.ttf` into it.
10. Now, you only have to run `texhash` and the system should be ready.

To test whether your configuration is correct, you can try to compile [this test file](#).

12.3.1 CJK Troubleshooting

There are many things that can go wrong when setting CJK support manually. If something seems not to work, the following checklist might help you.

- Obviously, since you run L^AT_EX as a user and not as root, you must *allow* ordinary users to access the new files. Make sure all folders and files are accessible using the `chmod` command.

- If L^AT_EX writes a DVI without problems, but you cannot view it, it is almost certainly because of some problems in the automatic generation of `*-pk` fonts. They are supposed to be generated on the fly when viewing a DVI file, but this might fail for a number of reasons: double-check `ttfonts-.map` for your custom line first. However, it might happen that your **ttf2pk** command, which is usually invoked by the DVI viewer, has been compiled *without* support for the kpathsea libraries. If this is the case, **ttf2pk --version** will make no mention of kpathsea. As support for these libraries is necessary, you might have to find a new package, or recompile FreeType 1 by yourself.

12.3.2 How do I input CJK in Unicode?

There are a number of different input engines, and the choice can depend also on personal preference. The author uses [Skim](#), a port to KDE of the [Scim](#) engine. Refer to your distribution's documentation to learn how to install these programs. Configuration of such programs can be tricky too, in the case of Skim you will have to define an environment variable `XMODIFIERS="@im=SCIM"` *before* starting X.

Chapter 13

Scripting

13.1 Scripting in Kile

Kile's scripting feature allows for the execution of ECMAScript code. Scripts can be managed through the scripting panel in the sidebar.

13.2 API Reference

In this section we describe Kile's scripting programming interface.

IMPORTANT

Please note that the scripting API has not been finalized yet. The API described below might change in future versions of Kile.

First of all, Kile's script execution environment provides a global object called "kile", which owns the following methods:

(kile).currentTextDocument () Returns a `KileTextDocument` object which reflects the currently active text document. Returns `null` if no text document is active.

(kile).getInputValue ((caption,, label)) Opens a dialog with the given caption and label. Returns the value that user has entered.

Objects of the type `KileTextDocument` represent text documents in Kile. They have the following properties:

(KileTextDocument).backspace () Deletes the character that is located immediately before the current cursor position and moves the cursor one position backward in the text.

- (KileTextDocument).cursorLeft ()** Moves the cursor one position backward in the text.
- (KileTextDocument).cursorRight ()** Moves the cursor one position forward in the text.
- (KileTextDocument).insertText ((*text*))** Inserts the text contained in the variable *text* into the document at the current cursor location.
- (KileTextDocument).insertBullet ()** Inserts a bullet into the document at the current cursor position.
- (KileTextDocument).nextBullet ()** Selects the first bullet located in the document immediately after the current cursor location.
- (KileTextDocument).previousBullet ()** Selects the first bullet located in the document immediately before the current cursor location.
- (KileTextDocument).up ()** Moves the cursor one line up in the document.
- (KileTextDocument).down ()** Moves the cursor one line down in the document.
- (KileTextDocument).cursorLine ()** Returns the line which the cursor is currently located at.
- (KileTextDocument).cursorColumn ()** Returns the column which the cursor is currently located at.
- (KileTextDocument).setCursorLine ((*line*))** Moves the cursor to the line denoted by *line*.
- (KileTextDocument).setCursorColumn ((*column*))** Moves the cursor to the column denoted by *line*.

Chapter 14

Help

14.1 Help Documents

L^AT_EX is a rather sophisticated system, where the basic features can be expanded by a great variety of additional packages. Kile provides a lot of different help to support the user.

LaTeX Reference An alphabetical index of the most common L^AT_EX commands.

TeX Documentation teTeX/TeX Live comes with a huge amount of documents. This includes a documentation for all included packages and an additional L^AT_EX reference.

LaTeX A full reference for T_EX and friends. This is not only a description of all programs, some important packages are also mentioned. And it includes a full reference manual of L^AT_EX commands—ideal for looking up a particular piece of formatting while writing a document. As this document is really extensive, Kile separates it with three important bookmarks.

LaTeX Command Another alphabetical index of the most common L^AT_EX commands.

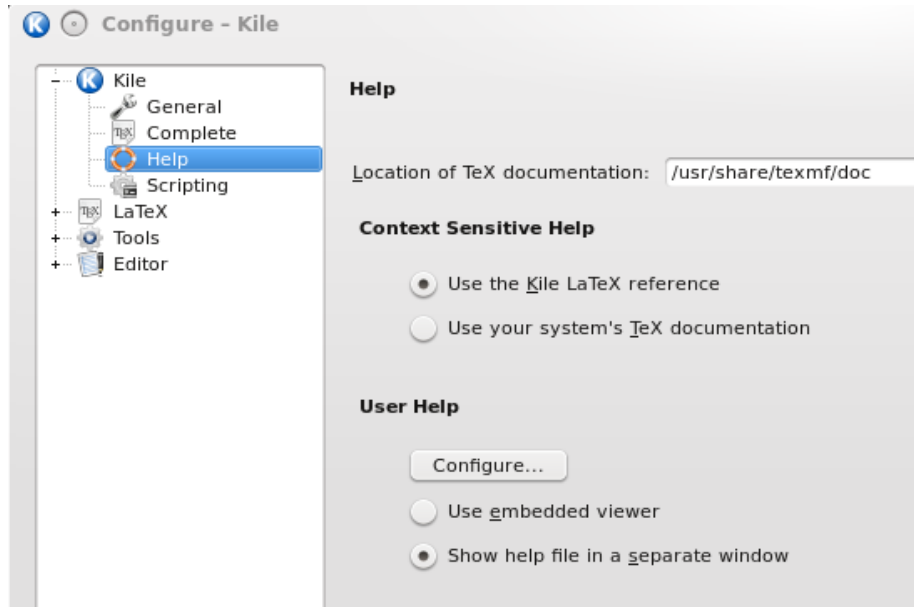
LaTeX Subject A description of important L^AT_EX subjects.

LaTeX Env An alphabetical index of the most common L^AT_EX environments.

14.2 Context Sensitive Help

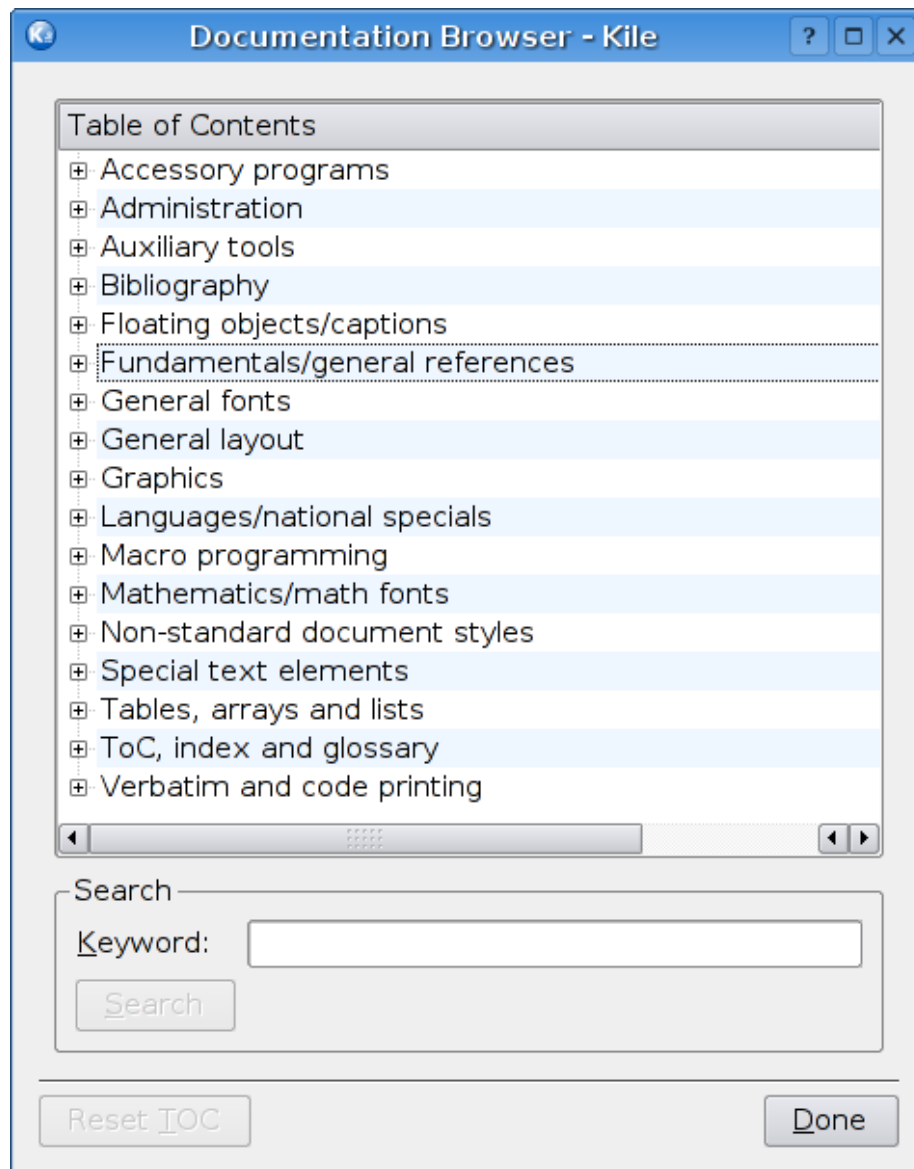
Kile also support a context sensitive help, which is called with Ctrl+Alt+H,K. In Settings → Configure Kile... → Kile+Help you can choose whether you want

to use Kile's L^AT_EX reference or the help system of t_EX/L^AT_EX Live, which is the default setting.

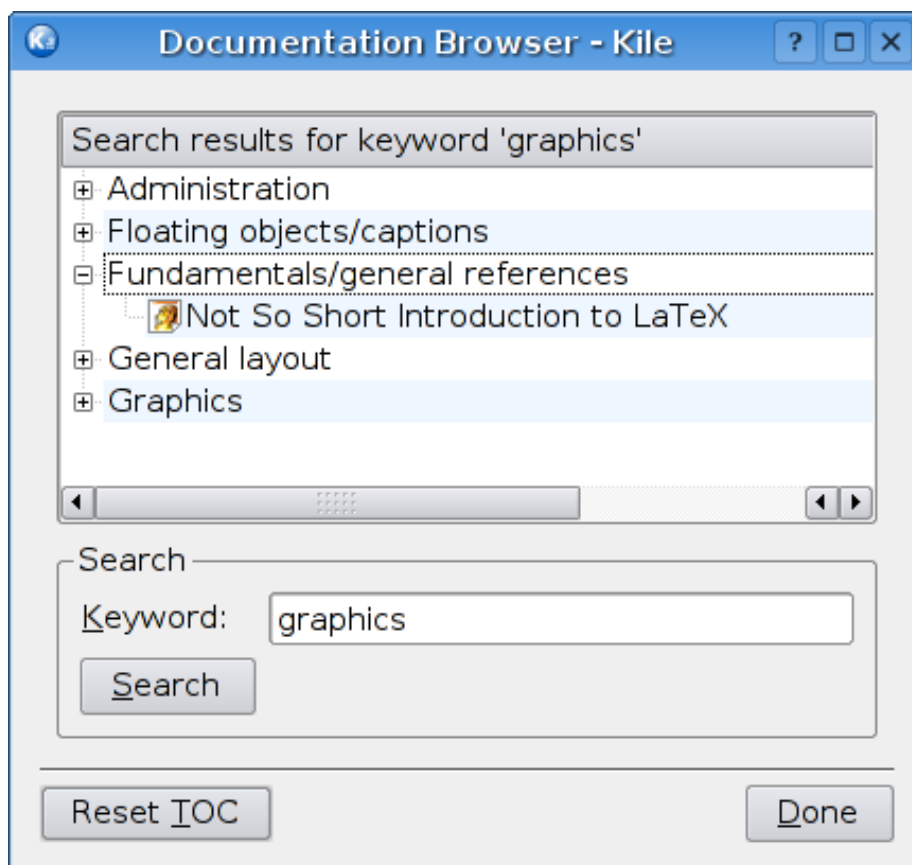


14.3 Searching for Keywords

It is not always easy to find the right document, as t_EX/L^AT_EX Live comes with a huge amount of documents. As one possible help, t_EX/L^AT_EX Live provides a tiny program `texdoctk`. It comes with a database of all documents that Kile uses to offer an interface to it.



All documents are grouped into some categories, and the main advantage is that you can search for packages names or keywords. Kile will then show only the results.

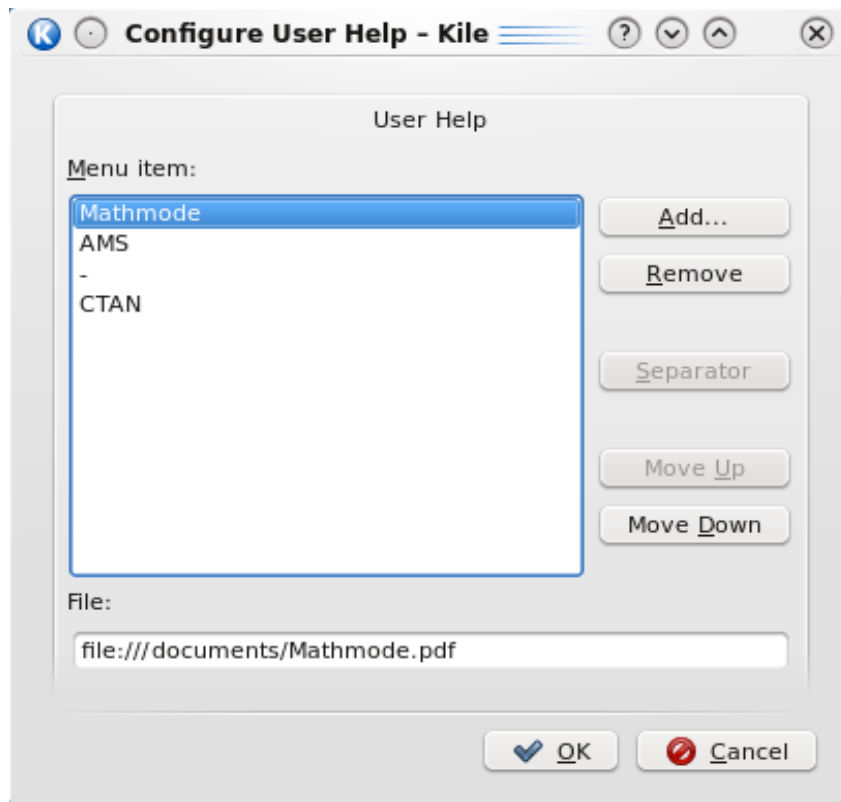


A mouse double click or the **Space** key will start the viewer for this document. This can be an arbitrary document, not only a DVI, PS, PDF or HTML document. Kile will take KDE settings to start an appropriate viewer.

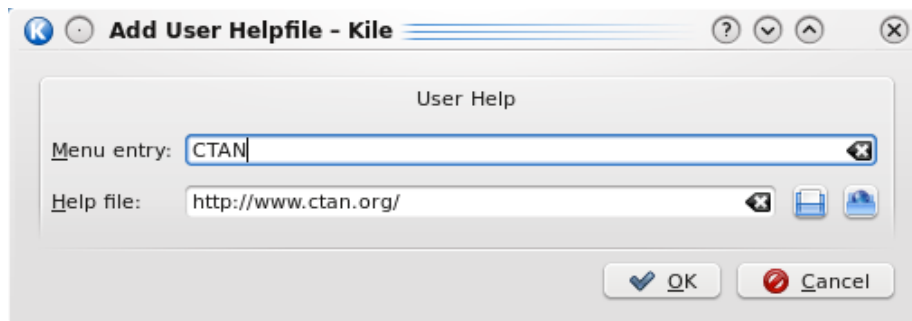
14.4 User Defined Help

Beside this static TeX /TeX Live documentation, Kile supports also another variable way for user-help documents. In Help menu Kile has a special User help submenu, where the user can add documents of his own choice. These can be the most important documents of TeX /TeX Live documentation, or even self written documents. It is even possible to choose some Web URLs.

Go to Settings → Configure Kile... → Kile+Help and choose Configure button to configure this User help menu. You can add, remove or navigate menu entries, and also insert separators to get a better structure.

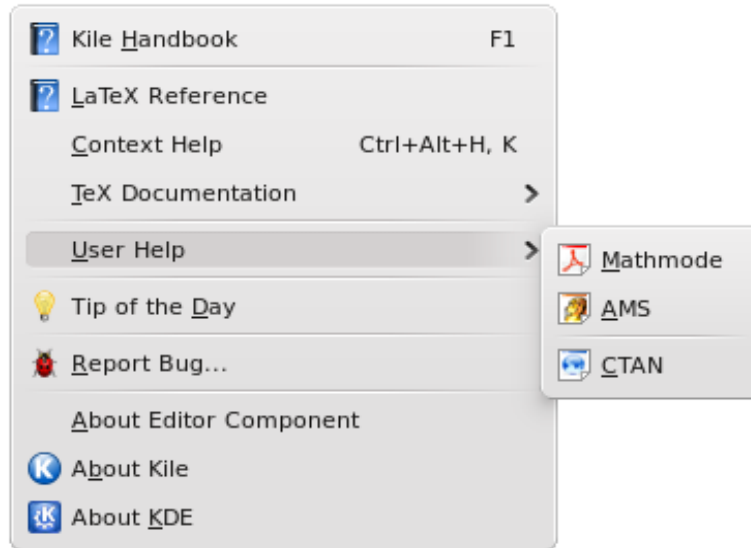


Pressing the Add button will open another dialog, where you must edit the name of the menu entry, and choose the corresponding file or Web URL. If you choose a Web URL, Konqueror is started and you should copy the final URL.



After configuration all chosen entries will appear in the Help menu of Kile as a special menu entry User help.

The Kile Handbook



Chapter 15

Credits and License

Kile is an open-source user-friendly L^AT_EX / T_EX source code editor. It will run on systems with the KDE Desktop Environment installed. KDE is available for several architectures with Linux[®] and other Unix-like systems installed. Kile is also part of the Fink project, which means you can run Kile on a Mac[®] with OS-X.

Many thanks are owed to those who strive to continue the Kile project, and the many hours of contributions made by those who sacrifice their time to develop tools we can all use under the GNU license. Up-to-date information about contributors can be found in the About Kile dialog from the Help menu.

Contributions among others from: Rob Lensen, Roland Schulz, Michael Margraf, Holger Danielsson

Many thanks to all those involved!

This documentation is licensed under the terms of the [GNU Free Documentation License](#).

This program is licensed under the terms of the [GNU General Public License](#).