

# Ltxshell v1.12 – Shell for L<sup>A</sup>T<sub>E</sub>X Commands

Volker Kiefel

September 15, 2006

## Contents

<b>1</b>	<b>Installation</b>	<b>1</b>
1.1	Win32-Systems . . . . .	1
1.2	Building and installing <code>ltxshell</code> on Linux systems . . . . .	2
<b>2</b>	<b>The main menu</b>	<b>3</b>
<b>3</b>	<b>Use of <code>ltxshell</code></b>	<b>3</b>
3.1	Selection of text files . . . . .	3
3.2	Shell commands . . . . .	3
3.3	Support of forward search (dvi-viewer) . . . . .	4
3.3.1	MikT <sub>E</sub> X . . . . .	4
3.3.2	Linux-Systems . . . . .	5
3.3.3	Details of <code>ltxshell</code> 's support for forward-search . . . . .	5
3.3.4	Forward-search – a solution for <code>ltxshell</code> and the vim text editor . . . . .	6
3.4	Additional functions . . . . .	7
<b>4</b>	<b>Changes</b>	<b>12</b>

## Short description

Support by text editors for calling external T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X-related programs ranges from “good” (Emacs + AucTeX) to “nearly absent” (gVim). `Ltxshell` provides a simple interface L<sup>A</sup>T<sub>E</sub>X-related tools for those who write L<sup>A</sup>T<sub>E</sub>X-documents. Among the programs supported by `ltxshell` are (PDF)L<sup>A</sup>T<sub>E</sub>X, dvi-viewer (`yap`, `xdvi`), `dvips`, `dvipdfm`, `gv/gsview32` and others on win32- and Linux-systems. `Ltxshell` is intended to run in a separate console window independent from the L<sup>A</sup>T<sub>E</sub>X text editor. Depending on the L<sup>A</sup>T<sub>E</sub>X-user's writing habits it may be useful to be used together with any text editor.

`Ltxshell` is provided as win32 console program. For its use on Linux-systems, the binary executable file has to be compiled from the sourcecode. `Ltxshell` can easily be installed manually and it is configurable for various (La)TeX-implementations: MikTeX, TeTeX, T<sub>E</sub>XLive.

`Ltxshell` supports single-file L<sup>A</sup>T<sub>E</sub>X-projects and multiple-file projects, but its full support requires that all L<sup>A</sup>T<sub>E</sub>X-files currently edited are in one directory/folder.

## 1 Installation

### 1.1 Win32-Systems

Extract the contents of the archive with the binary files (`ltxshell_nnn_w32.zip`<sup>1</sup>) into a temporary directory. It is recommended to copy these files into `c:\ltxshell`, since settings in the configuration

---

<sup>1</sup>with `nnn` for the version number

file `ltxshell.cfg` have default values<sup>2</sup> assuming installation in this folder/dictionary. Add an environment variable named `LTXSHELL_CFG_PATH` and assign the name of the directory of the configuration file to it:

```
LTXSHELL_CFG_PATH=c:\ltxshell\
```

Now you should add the name of this directory to the string assigned to the `PATH` environment variable.

An example of a configuration file is shown in figure 1, the variable names are explained in Table 1.

It is assumed that while you edit the files of your  $\text{\LaTeX}$ -project with a large and comfortable editor (gVim, Gnu Emacs, Xemacs, jedit, SciTE<sup>3</sup>) and open a console window from which you call `ltxshell`. The text editor called by `ltxshell` should be light and simple and start quickly as e.g. Windows' `notepad` or `win32pad`<sup>4</sup>, for details see section 3.

Each line of `ltxshell.cfg` consists of the name of the configuration variable, '=' indicating assignment and the value on the right side. An example is `PATH_TO_TEMPLATES`: if you have a directory with empty templates for various  $\text{\LaTeX}$  documents you may copy them from the directory assigned to `PATH_TO_TEMPLATES` into the current working directory.

If you wish to check syntax of  $\text{\LaTeX}$ -files with the command `m c l p`, you will have to install a copy of the program `lacheck`<sup>5</sup>.

## 1.2 Building and installing `ltxshell` on Linux systems

Compilation requires that you have the `gcc` (the GNU compiler collection) installed on your computer. Create a temporary directory and unzip the contents of `ltxshell_nnn_src.zip`<sup>6</sup> into this directory. Open the `Makefile` with your text editor. The first lines following the note on licensing conditions must have the format:

```
## win32 systems
#CFLAGS=-c -Wall -DLTXSHELL_USES_MINGW
#LTXSHELLEXE := ltxshell.exe

## Linux systems
CFLAGS=-c -Wall -DLTXSHELL_USES_LINUXGCC
LTXSHELLEXE := ltxshell
```

i. e. the commands for the compilation process on win32-systems must be commented out<sup>7</sup>.

Copy the `ltxshell` executable file into a directory included in the `PATH` environment variable. Write your configuration file `.ltxshellcfg` in your home directory (i. e. create `~/.ltxshellcfg`) with your text editor.

A good configuration for starting would be:

```
TEXT_EDITOR=vim
PRJNAME_FILE=prj.txt
```

---

<sup>2</sup>which - of course - can be changed

<sup>3</sup>This will be called  $\text{\LaTeX}$ -editor in this documentation

<sup>4</sup><http://www.gena01.com/win32pad/>

<sup>5</sup>available through CTAN: `support/lacheck`. An executable file is also available on the website <http://people.freenet.de/vkiefel/compiled-SW.html>

<sup>6</sup>with `nnn` for the version number

<sup>7</sup>If you wish to compile `ltxshell` on a win32-system, uncomment the two lines following the line "`## win32 systems`" and comment out the two lines following the line "`## Linux systems`"

```

DVI_VIEWER=xdvi -geometry 1150x900 -s 5
DVI_VIEWER_OPTIONS_FILE=/home/username/ltxshelldvi.txt
DVI_VIEWER_OPTIONS_MODE=0
LATEX_OPTIONS=-src
PROMPT="--> "
SHELLSC_FILE=/home/username/ltxshellsh.txt

```

where `username` is the name of your user account.

## 2 The main menu

Enter the menu commands in brackets, e.g. `[l]`<sup>8</sup> to  $\LaTeX$  the project, `[l+v]` to (1)  $\LaTeX$  the project and (2) view the generated dvi-file, `[menu]` shows the menu again. Enter an empty string to activate the default visible in brackets. The option `[m]` opens the submenu (figure 3).

## 3 Use of ltxshell

If you use ltxshell, you will probably start the text editor for editing  $\LaTeX$ -files (the  $\LaTeX$ -editor, e. g. Vim, GNU Emacs, SciTE, jedit) independently from ltxshell which should run in its own window. The text-editor called by ltxshell<sup>9</sup> should be a “small” one which requires only short start times (e. g. notepad, win32pad (win32 systems); vim, joe (Linux)). It is not necessary that this small text editor has  $\LaTeX$  syntax highlighting, whereas this is recommended for your text editor which you use for editing text files. After starting ltxshell, select `[np]` for a new project<sup>10</sup>. You will then be prompted to enter (the) first character(s) of the project file to be selected, if you enter an empty string, all `.tex` files in the current directory will be listed. Please select a  $\LaTeX$  main project file (the file of a project containing “`\documentclass{}`”). Alternatively you may start ltxshell with the project name as command line parameter:

```
ltxshell myproject
```

if `myproject.tex` is the main project file. The name of the current project is printed by ltxshell with the command `[prj]`. With `[l]`, you will process the project  $\LaTeX$ , `[v]` calls the dvi-viewer with the project dvi-file. `[l+v]` first  $\LaTeX$ es and then shows the project. Most of the other commands of the menu (figure 2) are self-explaining. If you wish to work on a new project, select the next project file with `[np]`. To change the directory, you may type `[cd]` [Return], enter the name of the new directory, and [Return].

### 3.1 Selection of text files

The menu options `[np]` and `[ct]` open a file selection screen (figure 4). If you call `[np]` or `[ct]`, you are prompted for the first character(s) of the file name, this allows you to restrict the number of files listed. If you call `[np t]` in the directory of figure 4, the list of figure 5 will appear.

### 3.2 Shell commands

With `[sh]` you can call external programs: if you have assigned the name of a file in the configuration file to `SHELLSC_FILE`, the `[sh]` command calls the text editor with the shell command file. You may enter

<sup>8</sup>`[l]` means: enter “l” and press the return key

<sup>9</sup>to be defined in the configuration file

<sup>10</sup>A project for ltxshell is the name of the “main” file except the `.tex` filename extension.

lines with commands which will be executed after closing the text editor. The first empty line will stop the execution of the command file<sup>11</sup>.

A shell command file is useful for processing a series of commands on a  $\text{\LaTeX}$  project. As an example, if all temporary files (including the `.aux`) files of a project (e. g. `myproject`) have been deleted, the project can be compiled with the following group of commands:

```
latex myproject
bibtex myproject
makeindex myproject
latex myproject
latex myproject
yap myproject
```

If you have already selected the project name with `[np]`, you may enter

```
latex %p
bibtex %p
makeindex %p
latex %p
latex %p
yap %p
```

as “`%p`” will be expanded to the current project name. You may append filename extensions as in

```
...
latex %p
dvips %p
gsview %p.ps
```

Similarly, “`%e`” is expanded to the name of the text editor defined in the configuration file. The keyword `stop` interrupts the sequence of shell commands with a prompt. The group of shell commands

```
lacheck %p.tex > out.txt
stop
%e out.txt
stop
del out.txt
```

checks the files of your current  $\text{\LaTeX}$ -file, opens the results file, and deletes it.

If a program called by `[sh]` requires that the `HOME` is assigned a value, this can be done with the `ENV_HOME` configuration variable<sup>12</sup>.

### 3.3 Support of forward search (dvi-viewer)

#### 3.3.1 MikTeX

This section should be introduced by a definition of “forward-search”. The following definition is taken from the documentation of Yap (MikTeXs dvi-viewer):

---

<sup>11</sup>i. e. everything following the empty line is ignored. This feature can be used to store “old” commands for later use in the same file

<sup>12</sup>The program author encountered this problem with the `ispell` program

“While editing a TeX source, issue an appropriate command that invokes Yap to display the TeXed document. Yap will indicate its idea of the current edit position by drawing a small circle. This is called forward search.”

A prerequisite for forward searching using MikTeX is that the L<sup>A</sup>T<sub>E</sub>X-source has been compiled with the `-src` command-line option. The appropriate command for yap is:

```
yap --find-src-special=srcspecial project_name  
  
yap -s srcspecial project_name
```

where `srcspecial` means `NNCURRENT_FILENAME` with `NN` for the line number in the current file (you are just editing) and `CURRENT_FILENAME` the name of the current file. An example: You are working on a L<sup>A</sup>T<sub>E</sub>X-project `myproj` with the main file:

```
\documentclass{article}  
...  
\begin{document}  
...  
\include{subfile1}  
\include{subfile2} ...  
...  
\end{document}
```

If you are just editing line number 255 of `subfile2.tex` and wish to jump to the corresponding text in yap, you will have to call

```
yap -s 255subfile2.tex myproj.dvi
```

or

```
yap --find-src-special=255subfile2.tex myproj.dvi
```

Ltxshell supports you in generating such calls to a dvi-viewer.

### 3.3.2 Linux-Systems

If you use `xdvi` on a Linux system, “forward-search” works similarly as described in section 3.3.1 with call formatted:

```
xdvi -sourceposition 255subfile2.tex myproj.dvi
```

### 3.3.3 Details of ltxshell’s support for forward-search

With the variable `DVI_VIEWER_OPTIONS_FILE` (in the configuration file) you may define a text file, where you can enter file name and line number of cursor position of the currently edited file. Ltxshell interprets the first line which is not empty as the filename of the currently edited file and the second non-empty line as the line number. If the variable `DVI_VIEWER_OPTIONS_MODE` has been assigned the value “1” ltxshell starts the text editor with `DVI_VIEWER_OPTIONS_FILE` if you select  or . You may then enter the file name and line number. After saving the file ltxshell generates a command

line with the appropriate additional command line option required for forward search. The format for this additional command line options is defined by the variable `DVI_VIEWER_OPTIONS` in the configuration file. Here you may enter the currently edited file name with `%f` and the line number of the current cursor position with `%l`. For MikTeX's `yap` this is:

```
DVI_VIEWER_OPTIONS=-s %l%f
```

or

```
DVI_VIEWER_OPTIONS=--find-src-special=%l%f
```

For `xdvi` on Linux/Unix-system you should enter:

```
DVI_VIEWER_OPTIONS=--sourceposition %l%f
```

If the variable `DVI_VIEWER_OPTIONS_MODE` has the value “0”, the file with the name of the currently edited file and line number is ignored, with the value “2” the file is read, but the text editor is not started. This is preferred, if the text editor for editing the  $\LaTeX$  files is able to generate the file with file name and line number. Details of this are described in section 3.3.4. With the value “1”, `ltxshell` opens the text editor with the file `DVI_VIEWER_OPTIONS_FILE` so that you can enter the currently edited file and the line number at which you are currently editing. You may write the name of the currently edited file with the command `m lf`.

### 3.3.4 Forward-search – a solution for `ltxshell` and the `vim` text editor

Enter the following code fragment in a script file read by `vim` at startup<sup>13</sup>:

```
:function PrintFileLineno()  
  redir! > c:\ltxshell\ltxshelldvi.txt  
  silent echo expand("%")  
  silent echo line(".")  
  redir END  
:endfunction  
  
au BufWritePre *.tex :call PrintFileLineno()
```

Whenever you edit a `.tex` file, `vim` writes name of currently edited file and line number into the file `c:\ltxshell\ltxshelldvi.txt`. You may edit this line in order to define another file in another directory. Now it is no more necessary to enter currently edited file and line number manually and you can assign the value “2” to the variable `DVI_VIEWER_OPTIONS_MODE` in the configuration file: the file `c:\ltxshell\ltxshelldvi.txt` will not be opened before the `dvi-viewer` is called by `ltxshell`, but `ltxshell` will read the file `ltxshelldvi.txt`. This is the most comfortable solution of forward-search with `ltxshell`.

---

<sup>13</sup> This code can be found in the program author's `vim` script `latex-for-vim.vim` which is available at <http://people.freenet.de/vkiefel/ownsoftware.html> in the archive [http://people.freenet.de/vkiefel/latex-for-vim\\_nn.zip](http://people.freenet.de/vkiefel/latex-for-vim_nn.zip). This script comprises additional support for entering  $\LaTeX$ -commands, environments and BibTeXreferences

### 3.4 Additional functions

Ltxshell writes the name of the main project `.tex` file into a small text file (e. g. `prj.txt`) if the name of this file has been assigned to the `PRJNAME_FILE` variable. This file can be included with:

```
{\tiny\raggedleft\input{prj.txt}\par}
```

or a similar construct, to add the path/name of a document to its printed version like this<sup>14</sup>:

`/home/kiefel/Documents/progr/TCPR/ltxshell/doc/ltxshell.tex`

This feature is similar to a commonly found practice with normal word processors.

---

<sup>14</sup>Ltxshell writes the path/filename with the `\verb++` command

```

LINES_OF_SCREEN=25 # number of lines in the console window
TEXT_EDITOR=notepad
PROMPT=": "
LATEX_OPTIONS="-src -quiet -halt-on-error"
PDFLATEX_OPTIONS="-quiet"
DVIPS_OPTIONS="-A"
DVIPDFM_OPTIONS="-p a4"

DVI_VIEWER=yap # MikTeX
PDF_VIEWER="C:\Programme\Adobe\Acrobat 5.0\Acrobat\Acrobat.exe"
PS_VIEWER=gsview32

BIBTEX8_OPTIONS=--huge
MAKEINDEX_OPTIONS="-c"

PATH_TO_TEMPLATES=c:\tex_pat # a directory with templates for LaTeX documents

TEMP_FILES=.log .dvi .ps .aux .pdf .bbl .blg .ilg .ind .lof .lot .toc
# files with these extensions belonging to the
# current project are removed with the [rm]
# command

PRJNAME_FILE=prj.txt

SHELLSC_FILE=c:\ltxshell\ltxshellsh.txt
# file for ltxshells 'shell scripts'

DVI_VIEWER_OPTIONS_FILE=c:\ltxshell\ltxshellldvi.txt
DVI_VIEWER_OPTIONS=-s %l%f # enables forward search for MikTeXs yap

## Please uncomment one of the following:
# DVI_VIEWER_OPTIONS_MODE=0 # ignore dvi-viewer options
# DVI_VIEWER_OPTIONS_MODE=1 # load text editor with DVI_VIEWER_OPTIONS_FILE
# containing name of currently edited file and
# line number within currently edited file.
DVI_VIEWER_OPTIONS_MODE=2 # process dvi-viewer options without
# calling the text editor. This option assumes
# that the LaTeX editor you use writes name of the
# currently edited file into
# DVI_VIEWER_OPTIONS_FILE

ENV_HOME=c:\ispell # may be useful with Luzius Schneiders
# ispell distribution
# (http://www.luziusschneider.com), otherwise
# please uncomment or remove this line

CHECK_LATEX_PROJECT=lacheck

```

Figure 1: An example of a ltxshell configuration file. The character # introduces a comment. Empty lines are ignored. This configuration file assumes that you have installed gVim and that you are using MikTeX. If you use gVim together with ltxshell and MikTeX, it is recommended to use the vim-script latex-for-vim.vim, cf. footnote 13 on page 6.

```

[ l] run LaTeX on project
[ v] view dvi file      [lv] l + v
[ ps] create PS file with dvips
[vps] view/print PS file  [lpv] l + ps + vps
[pdf] run pdfLaTeX on project
[ vp] view/print pdf file
[ dp] convert dvi to PDF with dvi2pdf
[ i] run makeindex on project
[ b] run BibTeX      [b8] BibTeX8
[log] show (pdf)LaTeX log file
[ e] open text editor
[cfg] edit configuration file
[ sh] run shell command(s)
[prj] show current project name
[pwd] print current working directory
[ cd] change working directory
[ np] select (new) project
[ ct] copy template into current directory
[ rm] remove unused, temporary files
[ m] more options    [q] quit

[menu]:

```

Figure 2: Main menu

```

[clp] check LaTeX files of current project
[ lf] select current LaTeX file
[ a] about ltxshell
[ q] quit, back to main menu

[menu]:

```

Figure 3: Menu opened with m

```

1: att.tex
2: dic.tex
3: einleitg.tex
4: grundl.tex
5: hemophil.tex
6: hemost.tex
7: hit2.tex
8: pl_funkt.tex
9: pli-verweise.tex
10: pli.tex
11: pltimm-allo.tex
12: pltimm-auto.tex
13: test.tex
14: thrombocytos.tex
15: thrombph.tex
16: tmiang.tex
17: unsort.tex
18: varia.tex
19: vwk.tex

Select file number: 1..19, [q] to abort:

```

Figure 4: File selection screen

```
Please enter initial character(s) of project name: t

  1: test.tex
  2: thrombocytos.tex
  3: thrombph.tex
  4: tmiang.tex

Select file number: 1..4, [q] to abort:
```

Figure 5: File selection screen: only files beginning with “t” selected

BIBTEX_OPTIONS	command line options for the <code>bibtex</code> program (default: empty string)
BIBTEX8_OPTIONS	command line options for the <code>bibtex8</code> program (default: empty string)
CHECK_LATEX_PROJECT	name of $\LaTeX$ syntax checker (default: <code>lacheck</code> )
CLP_OPTIONS	command line options for the $\LaTeX$ syntax checker
DVIPDFM_OPTIONS	command line options for the <code>dvipdfm</code> program (default: empty string)
DVIPS_OPTIONS	command line options for the <code>dvips</code> program (default: empty string)
DVI_VIEWER	name of the dvi-viewing/-printing program (default: <code>yap</code> (MikTeX), <code>xdvi</code> (Linux))
DVI_VIEWER_OPTIONS	pattern of additional command line options for the dvi-viewer ( <code>%f</code> : name of currently edited file, <code>%l</code> : line number in currently edited file), default: empty string
DVI_VIEWER_OPTIONS_FILE	name of file where <code>ltxshell</code> can read the name of the currently edited file and the line number, default: empty string
DVI_VIEWER_OPTIONS_MODE	mode how <code>ltxshell</code> and text editor treat the file addressed by <code>DVI_VIEWER_OPTIONS_FILE</code> possible values: 0, 1, 2, default: 0
ENV_HOME	<code>HOME</code> environment variable (visible for programs called by <code>sh</code> ) (default <code>c:\</code> )
LATEX_OPTIONS	additional command line options for call to $\LaTeX$ (default: empty string)
LINES_OF_SCREEN	number of lines of the screen of the console window (default: 25)
MAKEINDEX_OPTIONS	additional command line options for the <code>makeindex</code> program
OUTPUT_FILE	Name of output file for programs called by <code>ltxshell</code> , currently <code>m clp</code>
PATH_TO_TEMPLATES	path to the directory with $\LaTeX$ document templates (default: empty string)
PDF_VIEWER	path/name of the pdf-files viewing/printing program (default: <code>gsview32</code> : you may change this to the name of the Acrobat reader or the Acrobat program)
PDFLATEX_OPTIONS	additional command line options for call to <code>PDF<math>\LaTeX</math></code> (default: empty string)
PRJNAME_FILE	Name of file with the complete project <code>.tex</code> file and path (default: <code>prj.txt</code> )
PROMPT	command line prompt
PS_VIEWER	PostScript files viewing/printing program (default: <code>gsview32</code> )
SHELLSC_FILE	name of shell script
TEMP_FILES	list of temporary files extensions, extensions separated by spaces (default: <code>.log .dvi .aux</code> )
TEXT_EDITOR	name of text editor, called by <code>cfg</code> , <code>e</code> and <code>log</code> commands (default: <code>notepad.exe</code> : you should change this if you wish to use another file)

Table 1: Complete table of variables for the configuration file

## 4 Changes

**v0.3:** New menu option `[rm]`: calls the shell script to delete unused files. Selection of `[cd]` deletes the current project name. Menu option `[np]` now prompts for first letter(s) of the project name. Command line option `--help` now lists all variables for the configuration file. Documentation has been completely rewritten and is now provided as PDF-file.

**v0.4:** `[sh]` has now an enhanced function. If a filename is assigned to `SHELLSC_FILE`, `[sh]` calls the text editor for a shell script file.

**v1.0:** Shell commands files: “%p” is interpreted as the project name, “%e” as the text editor defined in the configuration file. The keyword `stop` interrupts a sequence of shell commands with a prompt. If a shell command returns 1 or higher, a message is issued. Ltxshell may be started with the project name as command line parameter.

**v1.1:** Command `[l+v]` renamed to `[lv]`. New command `[lpv]`: runs `LATEX`, `dvips` and the PostScript viewing/printing program. Layout of main menu changed. Bug removed: %p and %e can now be expanded more than one time in one command line (which was already intended for the previous version).

**v1.2:** Command `[rm]` now deletes temporary files of the current project, their filename extensions can be assigned to the variable `TEMP_FILES`. The configuration file variable `DELETE_UNUSED_SCRIPT` is no longer functional. New configuration file variable `ENV_HOME`, sets the `HOME` environment variable for shell scripts called with `[sh]`. Documentation updated.

**v1.3:** New configuration file variable `MAKEINDEX_OPTIONS`, sets options for the `makeindex` program. Variable `PRJNAME_FILE` added: ltxshell now writes name and complete path of the main project tex file into a text file. Documentation updated.

**v1.4:** Internal change – function listing files in a directory changed.

**v1.5:** Support for forward search in dvi-Viewer (e. g. Yap of Mik<sub>T</sub>E<sub>X</sub>) new configuration variables `DVI_VIEWER_OPTIONS`, `DVI_VIEWER_OPTIONS_MODE`, `DVI_VIEWER_OPTIONS_FILE`.

**v1.6:** Changes of the menu structure: `[m]` leads to a submenu with a new option `[clp]` (check `LATEX` files) of the current project. After calling the `[cfg]` command, changes become effective at once, e. g. without restarting ltxshell. New configuration variables `BIBTEX_OPTIONS`, `BIBTEX8_OPTIONS`, `CHECK_LATEX_PROJECT` and `CLP_OPTIONS`. Documentation completed (section 3).

**v1.7:** Bugfixes – ltxshell now reads variables `DVI_VIEWER_OPTIONS`, `DVI_VIEWER_OPTIONS_FILE` and `DVI_VIEWER_OPTIONS_MODE` correctly from the configuration file.

**v1.8:** Bugfix: uninitialized variables fixed.

**v1.9:** Documentation completed.

**v1.10:** Configuration variable `PDFLATEX_OPTIONS` added.

**v1.11:** The manual installation procedure now assumes that the software is installed in `c:\ltxshell` by default. Description of the installation procedure has been rewritten. The command `[m l f]` (select current `LATEX` file) has been added.

**v1.12:** Ltxshell can now also be used with Linux, therefore the manual (this file) has been extended, figures and the table have been moved to the end of the manual. Several bugs fixed.