
The EPM T_EX Front End for V_TE_X/2

Version 3.0-V
2000-04-19

Contents

1 Overview	2
2 Installing	2
2.1 Prerequisites	2
2.2 Installing the files	2
2.3 Initial configuration	2
2.3.1 REXX profile	2
2.3.2 Configuring the PS and PDF viewers	3
2.3.3 V _T E _X options	3
3 Usage	3
3.1 The TeX menu	3
3.2 The hot-keys	4
3.3 L _A T _E X and other T _E X formats	4
3.4 Master file support	4
3.5 TeXing a marked region of a file	5
3.6 L _A T _E X syntax assist	5
3.7 Syntax highlighting	6
3.8 File types	6
4 Advanvced configuration	6
4.1 User-defined menu items	6
4.2 The PS and PDF viewers	7
4.3 The configuration tool	7
5 Known bugs	7
6 Distribution and support	8

1 Overview

The EPM T_EX Front End is an additional module for the OS/2 'Enhanced Editor' (EPM), which turns the editor into an integrated V_TE_X environment:

- V_TE_X, the PDF viewer and other programs can be started from an editor menu.
- T_EX errors are marked and explained in the source code window.
- L_AT_EX syntax assist is optionally provided.
- A marked region of a file can be TeXed.
- Multi-file documents are supported.
- V_TE_X options can be specified interactively.

The T_EX Front End can easily be added to an existing EPM. Alternatively you can merge its E language source code into a customized EPM version; see the archive `source.zip`.

2 Installing

2.1 Prerequisites

T_EX Front End requires version 6.03b of the 'Enhanced Editor', which is part of Warp 4. It will *not* work in conjunction with EPM 5.51, that was shipped with Warp 3. However, v6.03b can also be installed within Warp 3, and it is available for free, see

```
ftp://ftp.leo.org/pub/comp/os/os2/leo/editors/epm603b.zip
```

or

```
ftp://ftp.software.ibm.com/ps/products/os2/fixes/v3.0warp/english-us/  
epmbbs/*
```

In contrast to Warp 4, this so-called 'BBS distribution' contains also the source code of the EPM and the related compiler *ETPM*.

Furthermore, the T_EX Front End requires a ready-to-run V_TE_X/2 system.

2.2 Installing the files

Unpack the zip archive `tfe.zip` to the directory, where your EPM resides. In case you have installed it in conjunction with Warp, this should be the directory `\OS2\APPS` on the boot drive.

In case you have already installed a previous version of the T_EX Front End, you should *not* replace your configuration file `tfe.cfg` during this step.

In the EPM directory, you will also find an icon `TEXFE.ICO` then, which can be used to designate the 'Editor for T_EX'.

2.3 Initial configuration

2.3.1 REXX profile

Using the T_EX Front End with the editor requires the module to be loaded into the EPM each time it starts up. This is accomplished by suitably editing the file `profile.ern`, which resides in the EPM directory; add the following line:

```
'link texfe.ex'
```

Notice that the quotes are part of the information!

In case the file `profile.ery` does not yet exist, you have to create it. The distribution contains a file named `profile.smp`, which can be used as a model. Beside loading the T_EX Front End, it will apply various useful changes to the EPM configuration; see the comments in the file.

Make sure the EPM option *REXX profile* is active; see the settings notebook of the EPM. After closing and restarting the editor, the T_EX Front End will be ready.

2.3.2 Configuring the PS and PDF viewers

When starting the T_EX Front End for the first time, the commands for running the PDF and PostScript viewers are likely to need change. The default setting is to start a GSView, which has been installed on drive C: 'as usual'.

In case you have installed GSView on a different drive or in a different directory, or if you want to use a different program, the T_EX Front End can be configured as follows:

1. Open the menu *TeX/Settings*;
2. select the item 'PDF Preview';
3. click on *change*;
4. edit the name and path of the `.exe` file, do *not* change the other parts of the command;
5. OK
6. edit the item 'PDF preview' accordingly.

2.3.3 VT_EX options

In the *Settings* menu you can also change the options which are passed to the VT_EX program, for PDF as well as PostScript mode. Usually the options should be the same as in the CMD scripts `vlatex.cmd` and `vlatexp.cmd`.

The default settings fit VT_EX/2 v7.00.

3 Usage

3.1 The TeX menu

TeX→*PDF* runs VT_EX in PDF mode on the current document. The document will be saved, if necessary.

TeX→*PostScript* ditto, PostScript mode.

quick opens a submenu, where VT_EX can be started with the 'quck run' option; either in PDF or PS mode.

Locate next error analyzes the log file of the last T_EX run and marks the location of the next error in the source file. In case the line number of the error cannot be determined or is not found in the source file, the log file will be displayed.

View log file switches to the log file of the last TeX run. Executing this action while the log file is already being displayed will reset the error browsing facility to the first TeX error.

Start PDF viewer starts the PDF previewing program.

Start PS viewer starts the PostScript previewing program.

Settings opens the configuration window. Settings changed here are saved immediately. Notice that you can also change the options for calling VT \TeX here.

OS/2 Shell opens an OS/2 command shell window in the directory of the current file.

Folder opens the directory of the current file as a WPS folder.

Notice, that the menu items for the previewing programs and for the OS/2 command shell will always open a new window. They will *not* bring a window to the foreground which has been created before; doing so requires clicking on the particular window or using the window list.

3.2 The hot-keys

<i>TeX</i> → <i>PDF</i>	Ctrl-F9
<i>TeX</i> → <i>PostScript</i>	Ctrl-F10
<i>Locate next error</i>	Ctrl-F11
<i>View log file</i>	Ctrl-F12

3.3 \LaTeX and other \TeX formats

The default \TeX format is `latex`, unless a format is specified in the first line of the document, using the following syntax, e.g., for the 'plain' format:

```
% format: plain
```

This is compatible with E. Mattes' REXX program `text.cmd` and other \TeX IDEs. The default format `latex` can be changed in the configuration window.

3.4 Master file support

Long documents are typically spread over several files and included into a common 'master' file using `\input` or `\include` commands. To provide special support for this way of handling multi-file documents, you have to turn on the 'master file support' via the configuration menu. Then \TeX ing a document will cause all dependent files in the same EPM window (edit ring) to be checked for modification, and they will be saved if necessary. 'Dependent files' are those which are included via `\input` or `\include` commands. Nested dependencies are, however, not recognized.

Each of the dependent files should, in its first line, indicate the corresponding master file, e.g.:

```
% master: main
```

In this example the master file is `main.tex`. \TeX as well as the the previewing programs and the user-defined programs (see section 4.1) can be called then, while editing one of the dependent files, and the main document will be used for the action.

Disabling the master file support is not required when \TeX ing 'simple' documents. With long text files, however, it may slow down the \TeX Front End unnecessarily. Master file support should also be disabled when editing documented \LaTeX sources containing the commands `input` or `include` within their code sections.

3.5 TeXing a marked region of a file

It is possible to process a marked region of a file only, as opposed to formatting the complete file. This feature is enabled if TeXing is requested while there is a marked region in the file being edited. You will be asked to confirm that only the marked region is to be processed. This works with standard (CUA) marking mode as well as with the 'advanced' marking mode; block marks can, however, not be processed.

The marked region will be combined with the preamble (or header) of the document, i.e., the lines from the beginning of the file up to one containing `\begin{document}` or `"%**end of header`. In case the string `\begin{document}` is found, the corresponding `\end{document}` will be added to the end of the text to be formatted, otherwise `\bye`. The action stops with an appropriate error message, if the end of the header cannot be located. Please, note that a temporary file is created with the name of the current document and the extension `.uuu`.

The menu item *Locate next error* will display the log file only (neither the source file nor the temporary file), in case a marked region has been processed before.

3.6 L^AT_EX syntax assist

Environments: Typing `<return>` within a line that contains `\begin{...}`, causes the corresponding `\end{...}` statement to be added. In the new line the cursor is indented appropriately, except for the environments `document`, `verbatim`, `verbatim*` and `macrocode`. Furthermore, entering `\begin{macrocode}` will cause the current line to be replaced with

```
%uuu \begin{macrocode}
```

```
%uuu \end{macrocode}
```

Additionally, `\[` is recognized as an environment, too.

Keywords: The following commands and keywords are completed when entering a space after the first three characters:

<code>\(</code> → <code>\(\)</code>	<code>\la</code> → <code>\label{}</code>
<code>\be</code> → <code>\begin{}</code>	<code>\re</code> → <code>\ref{}</code>
<code>equ</code> → <code>equation</code>	<code>\pa</code> → <code>\pageref{}</code>
<code>eqn</code> → <code>eqnarray</code>	<code>\ci</code> → <code>\cite{}</code>
<code>\fr</code> → <code>\frac{ }{ }</code>	<code>\fo</code> → <code>\footnote{}</code>

Completion of environments and keywords is usually active with documents and style files, unless you disable the 'expand' facility of the Editor. You can override syntax assist once by typing the Ctrl key in conjunction with `<return>` or `<space>`.

The accelerator keys known from EPMT_EX 3.1 are available, too:

Alt-A → <code>eqnarray</code>	Alt-B → <code>\begin{}</code>
Alt-C → <code>\cite{}</code>	Alt-D → <code>description</code>
Alt-E → <code>equation</code>	Alt-F → <code>\frac{ }{ }</code>
Alt-H → <code>\hspace*{}</code>	Alt-I → <code>\item</code>
Alt-J → <code>itemize</code>	Alt-K → <code>tabular</code>
Alt-L → <code>\label{}</code>	Alt-M → <code>\mbox{}</code>
Alt-N → <code>enumerate</code>	Alt-Q → <code>equation</code>
Alt-R → <code>\ref{}</code>	Alt-S → <code>\sqrt{}</code>
Alt-V → <code>\vspace*{}</code>	Alt-W → <code>verbatim</code>
\$ → <code>\$\$</code>	{ → <code>{}</code>

These keys are optional: Their behaviour can be toggled via the configuration window, with the initial setting being 'off'. Enabling/disabling these keys will only become effective after changing the current file.

3.7 Syntax highlighting

T_EX syntax highlighting can be toggled via the configuration window, too. Caution: Files already loaded are not affected. The initial setting is 'off'. Syntax highlighting is controlled by the file EPMKWD_S.TEX which does not belong to the T_EX Front End, but is part of the OS/2 EPM package.

3.8 File types

Certain features of the TeX Front End are enabled according to the type (extension) of the current file:

Running T_EX is possible for file types defined as TeX documents; initially this is .tex only .

changed (v3.0)

Syntax assist and highlighting should be provided when editing style files, too. Initially, only the type .sty is recognized.

The configuration window provides items for editing the file types, which are recognized as documents, style files or auxiliary files. The file types must be entered *without* the leading dot.

Notice, that documented L^AT_EX sources, i.e., the file types .dtx and .fdd, are never highlighted.

4 Advanced configuration

4.1 User-defined menu items

There is a file tfe.cfg in the EPM directory, where up to four user-defined programs can be specified; they can be run from the T_EX Front End menu then. The sample configuration file defines BibTeX as one of these programs:

```
USER1_CMD      = start /n /f bibtex %**N
USER1_EXT      = .aux
USER1_MENU_TEXT = BibTeX
USER1__MSG     = run BibTeX on current document
```

USER_n_CMD, $n = 1 \dots 4$, is the EPM command to be executed. The EPM starts external programs using the start command, which you know from the OS/2 command prompt. A binary .exe program is launched using the option /n, whereas a batch file is executed using /c. All programs will run in the directory of the current document.

Usually the name of the current document needs to be passed to the external program. You can specify where and how this is done via the strings %**N and %**F. When the command is actually executed, these strings will be replaced as follows:

```
%**N → file name without extension
%**F → file name with extension
```

Certain commands need a facility for extra options to be added manually when they are launched. If you include a string in square brackets to your command, then a window will pop up when the program is started and you will be asked to enter

additional options: 'Enter extra options for <string>. For instance, when *makeindex* is be called as

```
USER2_CMD = start /n /f makeindex [makeindex] %**N.
```

the EPM will prompt: Enter extra options for *makeindex*. The information you enter will be included into the command string just in the place of the square brackets.

If *USERn_EXT* is defined, running the program *#n* requires the file with the given extension to exist. E.g., BibTeX can be run only if the *.aux* file exists. In case no file type is specified, the program can always be started.

USERn_MENU_TEXT is the title of the corresponding menu item. A tilde *~* in this string makes the following character act as a menu shortcut.

USERn__MSG is an (optional) menu prompt.

You can add further programs simply by following the above model.

Having completed editing the configuration file *tfe.cfg*, you have to change to the EPM directory and run the command *tfconfig*, which will transfer the information from *tfe.cfg* to the binary EPM configuration file.

Further changes to *tfe.cfg* will become effective only after repeating this step! Notice that an existing entry cannot be deleted by simply omitting it from the next 'generation' of *tfe.cfg*. You must supply an "empty" value, e.g., *USER3_EXT=* instead.

4.2 The PS and PDF viewers

new description
(v3.0)

The commands for running the PS and PDF previewing programs, which can be entered in the settings menu, are evaluated in the same way as the above-mentioned commands for starting the user-defined programs.

4.3 The configuration tool

Invocation:

```
tfconfig [(file)] [(option)]
```

Options:

```
/N list contents of configuration file only  
/U read settings from configuration file and write them to the TEX Front End  
/V view current settings of the TEX Front End  
/D delete all TEX Front End settings from the EPM
```

/U is the default option. In case no file is specified, settings are read from the configuration file *.\tfe.cfg*. A given filename is ignored, if */V* or */D* is requested.

The settings, which are displayed via */V*, can be piped to a file; this file can be used as a valid configuration file then,

5 Known bugs

- The key Alt-T will always open the T_EX menu, regardless of the general setting of the 'menu accelerators'.
- The name of a T_EX file is required to have an extension.
- The master file dependency system recognizes only one *\input* or *\include* command per line.

- A drive letter in a master file specification is not always be evaluated properly. (There should, however, be no real need for specifying a drive letter there.)
- The error searching facility may sometimes show a wrong source file or none at all.

6 Distribution and support

For the current version of this software see

`ftp://dante.ctan.org/tex-archive/systems/os2/epmtfe/`

or any other CTAN host.

In case you encounter any problems upon installing or using the T_EX Front End, please, contact the author:

Walter Schmidt <`walter.schmidt@arcormail.de`>

Please, obtain my email address from the latest edition of this document!

Credits

Special thanks to Jon Hacker and Rodney Korte for providing the program EPMT_EX and for the permission to use large parts of their source code. Without the model EPMT_EX I would never have been able to write the T_EX Front End. Wonkoo Kim suggested and created the configuration menu, and he found various bugs.

Finally I would like to apologize for my bad English. English is not my native language. Suggestions for improving this document and the messages displayed by the program are welcome.