

# Shells for T<sub>E</sub>X

## Phons Bloemen

Information & Communication Theory  
 Dept. of Electrical Engineering  
 Eindhoven University of Technology  
 The Netherlands  
 phons@ei.ele.tue.nl

### Abstract

This is a review of two ‘shell’ programs for emT<sub>E</sub>X, the PC-T<sub>E</sub>X implementation of Eberhard Mattes. The ‘shells’ provide an integrated development environment for making T<sub>E</sub>X documents. The two ‘shells’ presented here are T<sub>E</sub>XSHELL by Jürgen Schlegelmilch, and *AT<sub>E</sub>X* by Wietse Dol, Erik Frambach and Maarten van der Vlerk.

## 1 Introduction

The well-known emT<sub>E</sub>X package of Eberhard Mattes has brought T<sub>E</sub>X and its companions into the reach of the PC-compatible computers. Besides a good implementation of T<sub>E</sub>X and METAFONT (even for various processors like 8088, 80168 and 80386, which make use of additional features like ‘protected mode’, and special ‘big’ versions), emT<sub>E</sub>X offers an excellent previewer, which works on various graphic screens from Hercules to (S)VGA. There are also printer drivers for HP Laserjet and for nearly all dot matrix printers (by making a little configuration `.dot` file). A dozen of handy utilities come with the package, like BiB<sub>T</sub>E<sub>X</sub>, MakeIndx, T<sub>E</sub>Xcad, mf job.

Because this is a collection of various programs, some way of integration must be found, especially for users who want to make T<sub>E</sub>X their main document preparation system, and do not want to be bothered by MS-DOS commands. Furthermore, an integrated environment consisting of an editor with function keys for the various T<sub>E</sub>X operations and utilities would greatly improve the acceptance of T<sub>E</sub>X by a large group of users, especially in ‘WORDPERFECT country: The Netherlands’.

I reviewed the two environments by installing them on my home emT<sub>E</sub>X installation, and on a ‘clean 1990 emT<sub>E</sub>X distribution’. The latter is the distribution of 6 high-density disks of emT<sub>E</sub>X, as it was released in 1990. Since, there have been various updates on several programs, but more about that later. My own emT<sub>E</sub>X distribution runs on a 386/25Mhz clone, and is up-to-date, with the beta-test releases of T<sub>E</sub>X, dvidrv and METAFONT installed. I also installed babel and NFSS, and a large collection of fonts, and the AM-SPELL spell checker. The packages were also tested on a 286/8Mhz clone with Hercules display. After installation, I used the two environments to make some documents (including this one).

## 2 T<sub>E</sub>XShell

T<sub>E</sub>XSHELL is an integrated environment for the emT<sub>E</sub>X family, authored by Jürgen Schlegelmilch. It is based on Borland’s Turbo Vision, and it looks a lot like the integrated environment of Borland’s programming languages like TURBO C and TURBO PASCAL. Version 2.6 comes in a .ZIP archive of 619 kilobytes. It can be obtained from various FTP-servers, like ftp.uni-stuttgart.de. The archive contains a German and an English version. Version 2.6 is released in september 1993. Most of the evaluation is based on version 2.5.2.

### 2.1 Installation

The installation is simple: unzip the archive and the files will be placed in the correct directory in the `\emtex` hierarchy. Then the file `\emtex\TEXSHELL.CFG` must be edited to record the drive on which it is installed. It is possible to install T<sub>E</sub>XSHELL in a user-defined directory, but this is not recommended. No files of the already installed emT<sub>E</sub>X need to be changed. Start the program with

```
c:\emtex> texshell
```

Now the T<sub>E</sub>XSHELL manual advises to check the *Options* menu, to examine and adjust the other options to your own needs. The *Options* menu has a consistency check built in: if you enter something impossible (a reference to a program you do not have, or an incorrect path name), the input window will not disappear, and the cursor keeps blinking on the error. The pre-loaded configuration is sufficient for a base installation of emT<sub>E</sub>X (i.e. the ‘1990’ distribution, with the german format files).

### 2.2 Customization

If you want to change the external commands, or install new ones, you have to dig into the manual. T<sub>E</sub>XSHELL is very powerful in calling external programs, but it

takes some study to understand how to make such a call.

There are a few predefined menu options for the ‘standard’ emT<sub>E</sub>X utilities such as `dvidrv`, BiB<sub>T</sub>E<sub>X</sub>, T<sub>E</sub>Xcad, `MakeIndx`, these also have their own predefined ‘hotkeys’. With respect to the dvi-drivers: it is possible to install a screen previewer, a draft and a final printer (in most cases: `dviscr`, `dvidot`, `dvihp1j`). This flaw has been corrected in version 2.6: here you can install up to 32 printing devices, and a screen previewer. There is also a *User* menu, where you can install your own programs: I tested this with AMSPELL.

For all external programs, the installation procedure is the same: you can choose from a series of *templates* to construct a command-line to call external programs like T<sub>E</sub>X itself, `dviscr`, `dvidot` and so on. These ‘templates’ depend on the filename being edited, the format file used (T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X) and various directory names you can install in the *Options* menu. Furthermore, you can assign an *environment* to each call. Settings in this environments are provided as DOS environment variables to the program when it is called from T<sub>E</sub>XSHELL. If you call a batch file as external command, the COMSPEC variable must be set in the environment (this is not mentioned in the manual)! The process of installing your own tools is tedious, and there are a few caveats.

For the ‘bare metal hackers’: the configuration file `TEXSHELL.CFG` is a plain ASCII file. It looks a lot like a Windows `.INI` file, and you can edit it to your wishes.

### 2.3 Editor

The editor provided by T<sub>E</sub>XSHELL is much like a standard TURBO PASCAL/TURBO C editor. The command structure is like old Wordstar with its Ctrl-K and Ctrl-Q commands, it works very quickly. The editor has a limit of 64 kilobytes per file, but it may have several files opened at the same time. It is possible to transfer text from one file to another using a clipboard and cut-n-paste. There are 10 function keys to put macros under. The editor provides wordwrap, but there is no way to make this the default. It also provides full mouse support for pointing the cursor, moving, sizing, closing windows and menu bar selection.

The package looks very colorful. On the slow machine with the Hercules display, the default ‘colors’ do not look very well. A little playing with the ‘color customization’ menu improved things a lot, but the program does not recognize monochrome displays.

There is a nice edit file selection scheme, which again much resembles the one used in TURBO PASCAL. The editor provides a ‘primary’ T<sub>E</sub>X file: when T<sub>E</sub>X is run, it starts with this file. T<sub>E</sub>XSHELL records its configuration together with the primary file, by writing a `.DSK` file.

### 2.4 Compile and preview

There is a menu *TeX* on the menu bar which gives access to these functions. Most of them have a ‘hotkey’. A very nice feature is an separate menu to view the log file of the command just executed in a *Log Window*, varying from T<sub>E</sub>X to previewer and BiB<sub>T</sub>E<sub>X</sub>. The ‘e’ option of T<sub>E</sub>X (provided when there is an error) is fully implemented, and with Ctrl-Q E / Ctrl-Q W you can go to the next/previous T<sub>E</sub>X error or warning! You can also use the *Log Window* to find out from which file an error originated, by point-and-shoot on the errors in the log file. Great if you are editing a big document, or are playing around with user-written style files.

The previewer can be operated by a ‘hotkey’ without any problems (only the installation of `mfjob` took some time, but everything works fine).

### 2.5 Help system

F1 gives context sensitive help: it provides help for the function you have selected. There is ‘help on Help’ and with Sft-F1 you can browse through an index and view the whole help file. The help files are in English or German. There is also online T<sub>E</sub>X help: in an edit window you can put the cursor on a T<sub>E</sub>X command, and get help on that command with Ctrl-F1.

### 2.6 Manuals

The T<sub>E</sub>XSHELL manual seems to be ‘not yet ready’. It is more a listing of all the features of every window. It contains a little section on what files you should have, and how to install T<sub>E</sub>XSHELL. Then there is an extended explanation of the use of *templates* and *environments*, and how to install new commands. The last section of the manual explains how to use T<sub>E</sub>XSHELL: it is a list of command grouped per menu on the menu bar, and a list of the editor functions. Everything is documented, but there is no such thing as a ‘guided tour’. There is an appendix containing a revision history.

### 2.7 Miscellaneous

This section contains a few remarks on other features of the system.

- Installation of new format files is simple when you have generated the appropriate `.FMT` files. However, it is not possible to assign emT<sub>E</sub>X command line options to each format file. Version 2.6 has a new option to do this, but you can also work around it with batch files.
- T<sub>E</sub>XSHELL provides good printing support. Before printing there is a menu to enter first and last page, and the number of pages to print. The ‘complete’ print job is given as a default. You can enter your own `dvidrv` options (if you want to print two A5 sheets sideways).
- T<sub>E</sub>XSHELL provides no spelling support. With some effort, I managed to install AMSPELL in T<sub>E</sub>XSHELL, using the *User* menu and a couple of batch files. AMSPELL provides all spelling checking you want.

- T<sub>E</sub>X<sub>SHELL</sub> does not support METAFONT when run 'stand-alone'. You then have to install it in a batch file in the *User* menu. METAFONT is supported through `mf job` for automatic font generation.
- I had to change the 'default' `TEXSHELL.CFG` to customize it to my needs (installed `AMSPELL`, `METAFONT`, `mf job`, `BiBDB`). Contact me for copies of the `TEXSHELL.CFG` file.

### 3 4T<sub>E</sub>X

4T<sub>E</sub>X is authored by Wietse Dol, Erik Frambach and Maarten van der Vlerk, from the University of Groningen. This integrated environment is based on JP Softwares 4DOS and SemWares QEDIT. The system is a large collection of 4DOS 'batch' .BTM files, which perform the various file-handling actions and call the programs from the emT<sub>E</sub>X collection. To make the system work nice, the authors have written several little utility programs to perform selection functions. The system was written to operate in networked environments, where the emT<sub>E</sub>X files are placed on a Novell network server, and the users run the programs from the network disk. The review is based on version 2.15 of 4T<sub>E</sub>X. I will make some references to version 2.20: this is a 'βtest-version' for the upcoming version 3.0 of 4T<sub>E</sub>X.

#### 3.1 Installation

4T<sub>E</sub>X was founded as a 'complete' system, containing the complete emT<sub>E</sub>X package, `AMSPELL`, `GHOSTSCRIPT` and so on. The archive `obelix.icce.rug.nl` in Groningen has organized the package in several parts: take a look at the various .TXT files. The shell itself hides in `4TEX.ARJ`. This package is intended for use on an existing 'standard' emT<sub>E</sub>X installation. It is about 3 megabytes, and contains the 4DOS .BTM files, updates to various emT<sub>E</sub>X utilities, and the shareware distributions of `PKZIP`, `QEDIT` and 4DOS version 4. As a bonus, over a 100 new style files will flood your `\emtex\texinput` directory. The 4T<sub>E</sub>X shell itself is about 300 'compressed' kilobytes, and the 'isolated' shell can be installed on an existing emT<sub>E</sub>X installation.

Installation is done as follows:

- Unpack the archive in the root of your emT<sub>E</sub>X drive with `c:\> arj x -v a:4texupgr.arj`. The use of ARJ is necessary because the archive spans more than one floppy (and ARJ compresses better than `PKZIP 1.1`). The whole package is put into the right directories.
- Some emT<sub>E</sub>X files must be deleted or moved: the 'compilers' `tex.exe`, `btex.exe` etc. must be moved to `c:\emtex\compilers`, and the utility programs `bibtex.exe`, `texchk.exe`, `makeindx.exe`, `texcad.exe`, `texchk.exe` to `c:\emtex\utils`.
- You must put 'LASTDRIVE=T' in your `CONFIG.SYS`
- You must edit `c:\emtex\btm\system.set`

and `c:\emtex\btm\texuser.set` to configure the system to your needs. If you have a 'standard' emT<sub>E</sub>X, you don't have to touch `system.set`. In `texuser.set` you have to set your personal defaults. Both files provide extended comments on each of the options, but they are quite long at first sight.

The system is started with  
`c:\emtex\btm> tex`

There are two batch files `tex.bat` and `tex.btm`: if you are already running 4DOS, it will not load it again (However, there are some restrictions to your personal `4DOS.INI` file, and you must have loaded `KSTACK`). If you are running MS-DOS or DRDOS, a new 4DOS command shell will be loaded with the right settings. After a while, the main menu of 4DOS is presented.

#### 3.2 Customization

All customization of 4T<sub>E</sub>X is done via the configuration files `texuser.set` and `system.set`. There is not much need for the possibility to install 'own' utilities, because there are a lot of utilities predefined, like `GHOSTSCRIPT`, various graphics utilities like `BM2FONT`, bibliography maintenance with `BiBTEX` and `BiBDB`, spelling check with `AMSPELL` and much, much more. Even some commercial software like `WORD FINDER` and `EUROGLOT` is supported. However, recent versions of 4T<sub>E</sub>X do offer the opportunity to install own utilities, in user-defined menus.

A system that is made out of 'human-readable' batch files (.BTM) is a paradise for 'hackers': if you study the manual of 4DOS (included in the distribution), you can add your own enhancements to the system. Not recommended to keep compatibility with the official release, but hard to resist.

#### 3.3 Editor

The editor of the 4T<sub>E</sub>X system is the shareware editor `QEDIT`. The authors have defined a set of macros to use with `QEDIT`. These extra macros provide 'hotkeys' to start the emT<sub>E</sub>X 'compiler' and the previewer, to start the `AMSPELL` spell checker, the `BiBDB` bibliography database program.

An editor help screen is brought up by F1, and it shows the standard `QEDIT` editor commands, as well as the special extensions. The `QEDIT` editor is capable of having more files open at the same time. Spell-checking is done by `AMSPELL` on a word basis in the editor, or on file basis from the main menu. Before `AMSPELL` is started, you can choose your language.

The mouse is supported in the `QEDIT` editor for cursor pointing, block definition and window switching. In the main menus, the mouse can be used to point to menu choices. 4T<sub>E</sub>X also provides the 'primary file' system as described in T<sub>E</sub>X<sub>SHELL</sub>. It also records its configuration together with the primary file, using the extension .OPT.

### 3.4 Compile and preview

The emT<sub>E</sub>X ‘compilers’ are started from the main menu in most cases. There are also ‘hotkeys’ to start them from the editor. A nice option is to start the emT<sub>E</sub>X compiler and previewer from the editor with a block selected: then only that block of text is T<sub>E</sub>Xed and shown.  $\mu$ T<sub>E</sub>X is intelligent about choosing the right document-styles for such a text block: it examines the primary file and puts its preamble together with a surrounding `\begin{document}` and `\end{document}` around the text block. The error processing is not as good as that of T<sub>E</sub>XSHELL: the ‘e’ option works, and you can view the log files, but that is all. The error checking will be improved in version 3, according to the authors.

The complete  $\mu$ T<sub>E</sub>X system comes with L<sup>A</sup>T<sub>E</sub>X format files with `babel` and `NFSS` installed. The format file contains hyphenation patterns for English, German, French and Dutch. In emT<sub>E</sub>X’s memory organization, they set aside 65K words for hyphenation patterns where 36K should be enough according to my information. This leaves little main memory to do the real T<sub>E</sub>X jobs in (like processing elaborate tables and figures). So the notorious message `TeX capacity exceeded` appears (too) soon.... Version 2.20 has the capability to generate new formats ‘on the fly’ where you can choose the languages to include from a menu. It then makes a format with just enough space for the hyphenation patterns you choose. A T<sub>E</sub>X with 7 languages loaded is possible!

### 3.5 Help system

$\mu$ T<sub>E</sub>X provides two different help systems. First, there is help available for each of the menus of the system explaining the various options in short. Second, there is a good ‘online’ T<sub>E</sub>Xhelp system called T<sub>E</sub>XHELP. This is a popup command which you can access everywhere with Alt-F2, Alt-F3, Alt-F4. It provides help on the T<sub>E</sub>X command the cursor is on, and you can navigate through its index. T<sub>E</sub>XHELP is a TSR program, taking 9 Kb of main memory, and 250 Kb EMS / disk swap space. The help is in English, but it is possible to install a help file in a different language.

### 3.6 Manuals

The  $\mu$ T<sub>E</sub>X manuals are very elaborate. In fact, they provide information on every aspect of the whole emT<sub>E</sub>X system, and all its utilities. Chapter 2 is fully devoted to using  $\mu$ T<sub>E</sub>X itself, and to install it. It contains a section of format-file generation, and installing `babel` and `NFSS`.

Then the manual covers the following topics in brief, concise sections: the emT<sub>E</sub>X ‘compilers’ themselves, the `dvidrv` programs, Postscript, bibliography and

index support, the `AMSPELL` spell-checker, a section on importing graphics in various ways (T<sub>E</sub>Xcad, `BM2FONT`), some miscellaneous utilities like `detex`, the TSR programs like T<sub>E</sub>XHELP and commercially available translation programs, and finally something about fonts and `METAFONT`/`mFjob` font generation. This is more than a manual, it is an example of how a ‘local guide’ should look like.

The manual concludes with a list of sources for more information, like T<sub>E</sub>X users groups (TUG, NTG), and distribution lists (T<sub>E</sub>X-NL). An extended bibliography for further reading is added.

### 3.7 Miscellaneous

- $\mu$ T<sub>E</sub>X provides support for different types of printers. The printer type can be chosen from a menu in the *Output* menu. There it is also possible to enter the printer port, first and last page, and the number of pages to print. You can enter your own `dvidrv` options. Postscript printers are supported via `dvips`, and even the use of `GHOSTSCRIPT` to get the postscript files on a simpler printer is possible (not tested).  
In earlier versions, the printer support was poor, especially for matrix printers: only a 9-pin was supported, and there was no opportunity to install user-defined printers. This was reported to the author, together with a proposal for a better way to do it ( $\mu$ T<sub>E</sub>X hacking is easy). Version 2.20 contains much better printer support.
- $\mu$ T<sub>E</sub>X provides an extra *Graphics* menu where you can do all types of conversions of pictures to get them into your T<sub>E</sub>X documents, provided the necessary programs are installed. It also provides extra *BiB<sub>T</sub>E<sub>X</sub>* and *MakeIndx* menus to manage bibliography databases and automatic indexing.
- Because  $\mu$ T<sub>E</sub>X was initially developed to run on a network, there is an extra menu called *T<sub>E</sub>Xbatch*. Here you can send print jobs to network printers and batch T<sub>E</sub>X jobs to fast computers on the network.

## 4 Conclusions

To conclude this review I give a short ‘pro-and-con’ list for both systems. First I want to remark that the authors of both systems did a good job, and that the systems are still in a development phase. The use of a ‘shell’ around the emT<sub>E</sub>X system is very useful, and it can speed up things even for experienced T<sub>E</sub>X users. It is also a step on the way to make T<sub>E</sub>X more attractable to the ‘average WORDPERFECT user’, and to a T<sub>E</sub>X ‘beginners package’. Some attention must be given to the fact that T<sub>E</sub>X ‘beginners’ should not be bothered by lengthy installation procedures: it should be ‘plug-and-play’.

T<sub>E</sub>X<sub>S</sub>HELL in brief:

- + The use of Turbo Vision gives T<sub>E</sub>X<sub>S</sub>HELL a professional look.
- + There is an excellent system of tracing T<sub>E</sub>X errors in your file.
- + The package is rather small (about 300K).
- + Powerful mechanism to call external programs.
- + The package is reasonably fast, even on slow machines.
- Hard to configure. A major drawback, if there is a 'universal' T<sub>E</sub>X<sub>S</sub>HELL.CFG file which covers the complete emT<sub>E</sub>X distribution and some related things, this would ease the installation in a great way.
- Lacks 'integration' of some utilities like AMSPELL.
- The manual needs some rewriting (especially if it is going to be used as a beginners guide).

4T<sub>E</sub>X in brief:

- + The package covers almost everything in the emT<sub>E</sub>X package, and even more.
- + Excellent manual, discusses also the emT<sub>E</sub>X utilities.
- + The T<sub>E</sub>X help is given by a stand-alone TSR program.
- + Spell-checking with AMSPELL is an integrated part.
- The package is slow (annoying on the 286 clone).
- The package is rather bulky, because it is packed with a lot of other stuff. The shell itself is 300K.
- Installation procedure is elaborate.
- The use of 'human-readable' programs makes the system vulnerable to users who 'customize' it by changing the program itself. But this also may be an advantage...