

Preface to the 4th edition of the 4allT_EX CD-rom

It is with great pleasure and pride that we introduce to you the fourth edition of the 4allT_EX CD-rom.

When this project was started back in november 1993, nobody dared to envision the success this CD-rom has become. Back then we were afraid we would not be able to sell all of the 250 CD-roms we were planning to produce. Since then we have sold nearly 10 times as many! And we sold them all over the world; from Japan to Singapore, South Africa, Argentina, Canada, New Zealand, to name but a few countries where one or more CD-roms have ended up.

The third edition of the 4allT_EX CD-rom which was produced in 1995 was a double CD-rom because everything simply would not fit on one disk anymore. This double CD-rom broke all records. In two years time about 4000 copies were sold world-wide.

We are overwhelmed by the success of this project that started out as a service to the members of the NTG, the Dutch language oriented T_EX Users Group. We sincerely hope this fourth edition is going to be just as successful as the first three releases were.

We wish you a lot of pleasure and success with the tools and information provided on both disks.

Wietse Dol
Erik Frambach

About this manual

This manual probably contains more information than you will ever need to know. It may also contain information you already know. We do not expect you to read all of it, in stead we will supply a quick overview of the content of all parts. We hope this will enable you to find the parts that are of interest to you.

Part 1 is a survey of \TeX , its flavors, and its twin sister METAFONT, within the context of Electronic Publishing. Especially for those who are not familiar with \TeX (yet) this will be a good starting point. While reading it you may encounter terms of concepts you do not understand immediately. In that case do not worry, just read on. Details are not important at this level.

This part is an abbreviated version of a paper presented by Kees van der Laan at NLUUG meeting of 2 November, 1993 (NLUUG stands for the Dutch Unix Users Group).

Part 2 explains in just a few pages the principles of \LaTeX : what philosophy it is based on, what features are available, how to get it and what support is available. Those who want to get a raw idea of what \LaTeX is may want to read these pages.

This part is an adapted version of a paper presented by Wietse Dol and Erik Frambach at the European \TeX conference in Gdańsk, September 1994.

Part 3 explains how \LaTeX is setup and how it is installed. Installation from CD-rom are explained, and hints for network setup are given.

Part 4 gives a complete overview of all the parts of \LaTeX . All menus are discussed in detail and all functions described.

Part 5 is for those who want to know exactly what is going on. This part explains the special functions of two editors in combination with \LaTeX . The \TeX compilers are described in such detail that you should be able to run them ‘by hand’ from the commandline if you wish. However, if you stick to \LaTeX ’s menus you may never need to know any of these details.

Part 6 is for those who want to know ‘everything’ about the print and preview program that \LaTeX supports. Again, if you stick to \LaTeX ’s menus you may never need to know any of these details.

Part 7 explains the functions of a long list of utilities that are supported by \LaTeX . Although they are all integrated in \LaTeX ’s menus it may be worthwhile to read this. Some background knowledge of e.g., using graphics or bibliographies will prove to be helpful and time-saving when you are writing a book or article that includes pictures and bibliographic references.

Part 8 is an overview of all software that is used by \LaTeX . Most of the software is *free software* (i.e. you do not have to pay for it), but some of it is *shareware*. E.g., if you are using \LaTeX you are using 4DOS, even if you are not aware of it. 4DOS is shareware so you are supposed to pay the (rather small) registration fee. The listing of software is also of interest for those who want to use utilities in their own way or for other purposes.

Part 9 covers \TeX support. Many \TeX user groups are active all over the world, there are many mailing lists, bulletin boards and file servers dedicated to \TeX . If you want to know more about \TeX c.s. you will find several references here that will put you on track.

A **bibliography** and an **index** complete this manual.

History, copyrights and acknowledgement

\LaTeX started as a simple batch file for just doing the edit-compile-view cycle of a \LaTeX job on the 4th of February 1991. In 1992 the Department of Econometrics of the State University of Groningen slowly changed from χ -writer to \LaTeX/\LaTeX . As the group of users grew, many wishes and utilities were added to the menu. Thanks to the suggestions and patience of the early users, \LaTeX evolved into the product of today. In 1993 the first diskette versions of \LaTeX was distributed and many people from outside the Groningen State University started to use \LaTeX . The big boom came after an invitation to demonstrate \TeX and \LaTeX at the annual meeting of the NLUUG (Dutch Unix User group). At this meeting a CD-rom of useful Unix freeware was given as a present to all participants. Here we realized that a CD-rom would be the ultimate way to distribute a (bulky and complete) \TeX implementation. In November 1993 Wietse Dol and Erik Frambach started with the first preparations to make a turn-key MS-DOS \TeX system on CD-rom. Much time was spend to make it possible to run from CD-rom (i.e. one can not write on a CD-rom and a lot of programs needed rewriting). To finance the project the NTG was invited to participate. The board of the NTG did not hesitate and after a discussion about the number one could probably sell, we decided to have a first edition of 250 CD-roms. Soon after presenting the first CD-rom at the June 1994 meeting of the NTG (i.e. the first CD-rom was presented to Eberhard Mattes) we were sold out. In August 1994 we decided to make a second release not only with some minor updates but also with another 100 Mbytes additional \TeX ware. In March 1995 we sold the last of our 1 250 copies. Preparing a third edition we found that we had so much valuable stuff that it would no longer fit on one CD-rom, so we decided to make a double CD-rom. The third edition broke all records by selling about 4 000 copies in two years. Then we thought it was time for a new version, completely updated, enhanced and even more complete than before. We hope you will enjoy this product. Suggestions for improvements are more than welcome.

This documentation was last updated on November 12, 1997. It was written by

Wietse Dol	e-mail: dolly@pi.net
Erik Frambach	e-mail: E.H.M.Frambach@eco.rug.nl
Siep Kroonenberg	e-mail: N.S.Kroonenberg@eco.rug.nl
Maarten H. van der Vlerk	e-mail: M.H.van.der.Vlerk@eco.rug.nl

Wietse Dol, Erik Frambach and Maarten H. van der Vlerk are the authors of the \LaTeX Workbench.

Siep Kroonenberg designed the cover of this manual and the prints on the CD-roms. She also wrote parts of this manual, and developed a set of batch files (“Some \TeX ”) for those who prefer to run \TeX from the command line.

Kees van der Laan supplied the introductory text ‘What is \TeX and METAFONT all about?’.

The NTG, the Dutch language oriented \TeX Users Group was very helpful in supplying a financial basis for the \LaTeX project and promoting the 4all \TeX CD-rom with much enthusiasm.

Various subscribers to the \LaTeX mailing list and beta-testers made good suggestions for improvements and extensions to \LaTeX .

Many other people made valuable contributions to the content of 4all \TeX , this manual or the CD-roms.

We want to say Thanks! to all of you. Without your input \LaTeX could never have matured as it did—and still does.

Please stay in touch and let us know what you think of our product.

Contents

Part 1	What is T_EX and Metafont all about?	1
1.1	Introduction	1
1.2	T _E X etc. tools	2
1.3	Importance	4
1.4	T _E X's flavors	5
1.5	T _E X's drivers	6
1.6	T _E X and fonts	7
1.7	Descriptive markup	8
1.8	T _E X and its author	9
1.9	T _E X and its users	9
1.10	T _E X and the publishers	10
1.11	T _E X and other Electronic Publishing tools	11
1.12	Trends	15
Part 2	4T_EX: a workbench for ms-dos PCs	17
2.1	Introduction	17
2.2	Principles of 4T _E X	18
2.2.1	Free software, shareware and commercial software	19
2.2.2	Setting up 4T _E X	19
2.3	Features	19
2.3.1	Add, delete, modify	19
2.3.2	Help in 4T _E X	20
2.3.3	Editing	20
2.3.4	Block-View: quick partial compilation	20
2.3.5	Spell-checking	21
2.3.6	Extended graphics support	21
2.3.7	Printing	22
2.3.8	Conversions to and from...	22
2.4	Availability	22
2.5	Support	22
Part 3	Installing 4T_EX	25
3.1	Introduction	25
3.2	Typography conventions	25

- 3.3 \TeX and \LaTeX 26
- 3.4 A Historical Note, or ‘What’s in a Name?’ 26
- 3.5 Principles of \LaTeX installation from CD-rom 27
 - 3.5.1 Installation from CD-rom 27
 - 3.5.2 Installation on your hard disk 30
 - 3.5.3 Fine-tuning \LaTeX to your taste 31
 - 3.5.4 Network setup 32
 - 3.5.5 Directory set up 33
- 3.6 Starting \LaTeX 34

Part 4 \LaTeX menus 37

- 4.1 The main menu 37
 - 4.1.1 Shortcuts and extras 40
- 4.2 The output menu 42
- 4.3 The Bib \TeX menu 45
- 4.4 The graphics menu 47
- 4.5 The MakeIndex menu 50
- 4.6 The conversion menu 52
- 4.7 The Fontlib menu 55
- 4.8 The format menu 56
- 4.9 The Metafont format menu 58
- 4.10 The MFjob menu 59
- 4.11 The Metafont menu 60
- 4.12 The Metapost menu 62
- 4.13 The Install PostScript font menu 64
- 4.14 The Install PostScript font family menu 67

Part 5 Editors and compilers 71

- 5.1 The editors 71
- 5.2 The compilers 73
 - 5.2.1 Preparation 74
 - 5.2.2 Command line 74
 - 5.2.3 Connecting an editor 76
 - 5.2.4 Format files 76

Part 6 Print and preview programs 81

- 6.1 \TeX dvi-drivers 81
 - 6.1.1 Fonts 82
 - 6.1.2 Em \TeX DVI-drivers command line 82
 - 6.1.3 Response files 85
 - 6.1.4 Environment variables 86
 - 6.1.5 The \LaTeX response files 86
 - 6.1.6 Interactive operation of DviScr and DviScrs 88
 - 6.1.7 Font substitution 89
- 6.2 Using PostScript and DviPS 89

- 6.2.1 Including PostScript graphics 90
- 6.2.2 PostScript fonts 91
- 6.2.3 DviPS command line 93
- 6.2.4 DviPS configuration 93
- 6.2.5 Ghostscript as printer and viewer 94
- 6.2.6 Ghostscript as previewer 94
- 6.3 DviDjC command line 95
- 6.3.1 Interactive operation of DviVGAc 96
- 6.4 DviWin as previewer 96
- 6.5 Available fonts 96
- 6.5.1 Commercial fonts 97
- 6.5.2 Using PostScript fonts 98
- 6.5.3 Automatic font generation 98
- 6.6 Supported output devices 99

Part 7 Utility programs 101

- 7.1 Bibliographies and indexes 101
- 7.1.1 BibT_EX 101
- 7.1.2 BibDB 103
- 7.2 MakeIndex 103
- 7.3 Spell-checking 105
- 7.3.1 Concepts 105
- 7.3.2 T_EX features 105
- 7.3.3 The user interface 106
- 7.4 Graphics 107
- 7.4.1 emT_EX and the \special command 108
- 7.4.2 BM2Font 110
- 7.4.3 HPGL files and HP2xx 112
- 7.4.4 PostScript and Ghostscript 112
- 7.4.5 The FIGURES and PSFIG style files 113
- 7.4.6 T_EXcad 115
- 7.4.7 L^AT_EXcad 115
- 7.4.8 Mayura Cad 115
- 7.4.9 GNUplot 115
- 7.4.10 QFig 115
- 7.4.11 Metafont 116
- 7.4.12 Metapost 116
- 7.4.13 Summary 116
- 7.5 L^AT_EX Help 117
- 7.6 L^AT_EX Mac 117
- 7.7 T_EXchk 117

Part 8 Overview of software used by L^AT_EX 119

- 8.1 Free software 119
- 8.2 Shareware 121

8.3 Editors 122

Part 9 \TeX support 125

9.1 \TeX user groups around the world 125

9.2 Mailing lists 129

9.3 4 \TeX Support 129

9.4 Bulletin Boards 130

9.5 File servers 130

9.6 Learning \TeX 132

Bibliography 133

Index 137

Part 1

What is T_EX and Metafont all about?

An abbreviated version of a paper presented by Kees van der Laan at NLUUG (the Dutch UNIX users group) meeting of 2 November, 1993. BLUe stands for ‘my innocent user and relative of Ben Lee User of the T_EXbook’. Nowadays we would say ‘Beginning L^AT_EX User’. Published in MAPS 93.2 (revised May 1995)

A survey of T_EX, its flavors, and its twin sister METAFONT, within the context of Electronic Publishing, is given.

1.1 Introduction

This work about computer-assisted typesetting by (L^A)T_EX and METAFONT in context, is aimed at a broad audience. Novice users à la BLU – Ben Lee User from the T_EXbook fame – who like to become informed what it is all about, advanced L^AT_EX users who hardly have heard of MANMAC, and mathematicians and publishers who will find the offerings of the AMS interesting.

Many notes, articles and books about T_EX have been published. Advanced ones exploring T_EX’s limits, and also contributions at the survey and introductory level. The latter deal with the macroscopic markup features as well as the microscopics of automatic kerning, (for example with A and V in AV) the automatic handling of ligatures (for example with f and i in fi), the automatic justification and hyphenation supported by hyphenation tables, and the formatting of math, tables and graphics. They also boast of the quality which can be obtained when formatting the typographic teasers: math, tables and graphics.

In the T_EXniques series we have the tutorials: ‘A gentle introduction to T_EX,’ by Michael Doob, and ‘First grade T_EX,’ by Arthur Samuel. For L^AT_EX there is: ‘An introduction to L^AT_EX,’ by Michael Urban, and – for the Dutch speaking community – ‘Publiceren met L^AT_EX,’ by de Bruin. Also noteworthy is Hoenig’s ‘T_EX for new users,’ and the introduction chapter in Salomon’s courseware ‘Insights and Hindsight’, published by Springer-Verlag with title ‘The Advanced T_EXbook.’ For METAFONT see Henderson’s ‘An introduction to METAFONT,’ Tobin’s ‘METAFONT for beginners,’ and Knuth’s introductory article on the issue in TUGboat. Superb is Hobby’s ‘A user’s manual for Metapost.’ A survey with respect to Electronic Publishing tools is ‘Document Formatting Systems: Survey, Concepts and Issues,’ by Furuta and co-authors.

For trying it out and working with it, the user groups distribute PD versions of (L^A)T_EX as well as integrated working environments for PCs, with all kinds of bells-and-whistles added. Ubiquitous is Mattes’ PD emT_EX, and the working environment 4allT_EX, NTG’s off-the-shelf, turnkey T_EX system distributed on CD-rom. The T_EX community distributes the T_EX Live CD-rom aimed at UNIX platforms.

This paper relates T_EX and METAFONT to EP, SGML and the like, as a helicopter view, and accounts for the many activities of its users.

Conventions and notations

I adhered to the historical development of T_EX et cetera, and did not order the tools with respect to perceived importance. The latter is a matter of taste and definitely time-dependent.

Because it is a ‘helicopter’ view I need to refer to other work. This has been done a little loose through the name of the (first) author and the title, or keywords from the title. The reader can easily spot from the supplied list of references which work is hinted at. Just start by the author name and look for the matching title. Hereby I assume that readers are familiar with some journal names, for example TUGboat, the journal of the T_EX Users Group and the MAPS, the journal of the NTG.

For common words in the T_EX arcana – like T_EX, L^AT_EX, AMS, et cetera – I adopted the TUGboat typesetting conventions by using their macros for formatting these names. File names are set in the typewriter font.

1.2 T_EX etc. tools

First of all T_EX etc. has been around since the late seventies, and the users have contributed to the components and to the porting to all conceivable platforms, with the result that it is not easy to really survey the whole complex.

Going back to the roots we can say that T_EX is a program for formatting documents, born as a twin with its sister METAFONT, for creating fonts. T_EX and METAFONT have been designed to facilitate the high-quality computer-assisted production of books. A more modern way of talking is that T_EX is a markup language with METAFONT the accompanying tool for designing the needed graphics, starting with the fonts. In the mid-nineties John Hobby released his METAPOST in the public domain. METAPOST complies with the METAFONT language and is aimed at creating pictures as files.

A nice survey of the most important components and files when working with T_EX is supplied by the diagram in Figure 1.1, which illustrates the two main fields: font design and typesetting, with the relations between the components and files, all in one, and abstracting from details.

That is

- the flow from copy to printed results
- where the editor and its associated tools come in
- the location of T_EX – its flavors, and add-ons – at the heart
- what is used from METAFONT and where
- the printer independence through various drivers
- at what level POSTSCRIPT can be included
- where Acrobat comes in.

The important files are indicated by their extensions and are depicted within ovals. What holds for creating the .tex file holds also for the .mf file.

Working environments

The needed tools are nowadays embedded in computer-assisted (scientific) working environments. At first sight this seems trivial, but it is really handy that the tools are integrated, also with non-formatting

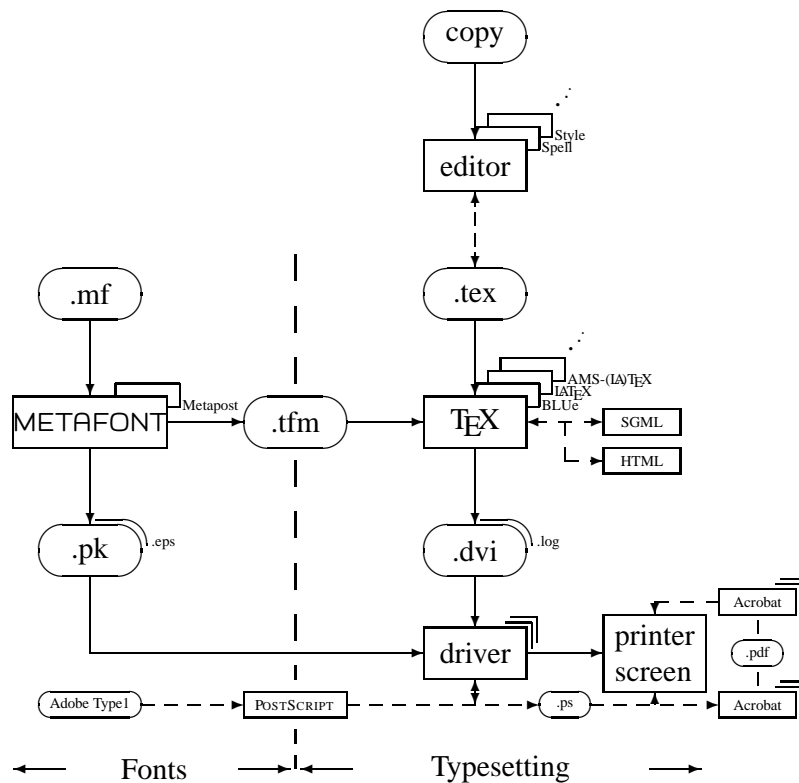


Figure 1.1: Overview of \TeX & METAFONT

applications per se, such as email, database applications, spreadsheets, and the use of programming languages such as C or good old FORTRAN. A model of thinking is that, for example, a thesis is prepared and all the simulations and calculations are done as a side-step of the main work: publishing! That is document preparation, formatting, typesetting, and dissemination. The graphics-oriented PCs like Macintosh and Atari paved the way. Nowadays the MS-DOS, OS/2, ... PCs with their (graphics) window facilities allow this way of working too.

Installation

The products are usually accompanied by their installation documentation. Famous, and top class, are the AMS installation guides for their formats and fonts. With the PD PC versions the idea is to supply turn-key scripts so that the installation goes automatically. When you have questions during installation or afterwards the '100 Frequently Asked Questions,' ukTUG Baskerville 4, 6, is handy.

Lifetime

The \TeX and METAFONT programs have been designed with flexibility, portability, and open-endedness in mind. Knuth envisioned that the two could be used a hundred years from now, just as we do today, with the same input and with the same or better results! In order to make this possible Knuth

- invented WEB, better known as literate programming
- documented the programs (open system)
- worked hard on making the systems error free

- delivered the twins in the public domain, and
- froze the kernels.

Because of these goodies the user community could port the systems to all platforms, and add layers on top to adjust for users' wishes and demands. All-in-all one can say that the twins are portable in place and time, are powerful, useful, and will serve a lifetime.

The working environments suffer from a much shorter lifetime. Read: need continuous maintenance and that is something, especially in a volunteer-based world. It is always a matter of the right balance: how fast do I need to do the day-to-day work and how often do I wish to upgrade the working environment.

1.3 Importance

From the computer science point of view $\text{T}_{\text{E}}\text{X}$ and METAFONT are big research achievements in how software engineering should be done, if not for the literate programming way of software design and creation. Top-class algorithms for line-breaking, hyphenation and page make-up have been incorporated. It is designed to be device-independent. That Knuth succeeded so well in his basic research can be concluded from the many publications which have been built upon his Computer and Typesetting works, and from the many honorary degrees he has received.

From the users' point of view $\text{T}_{\text{E}}\text{X}$ etc. is relevant because of the quality which can be obtained when used as a formatter, and because it is a high-quality, stable, open and freely available system. That Knuth succeeded here can be distilled from the many organized users of (\LaTeX) $\text{T}_{\text{E}}\text{X}$ world-wide, and from the tenfold who just use the system.

Its weakness is that $\text{T}_{\text{E}}\text{X}$ proper does *not* have easy user guides. This weakness has been compensated for by efforts like \LaTeX , $\text{AMST}_{\text{E}}\text{X}/\LaTeX$, and the styles from publishing houses and their user and installation guides. Of late manmac and developments since have been provided with a users guide called Publishing with $\text{T}_{\text{E}}\text{X}$.

Perhaps an unexpected side-effect of $\text{T}_{\text{E}}\text{X}$ is that it is so heavily used with alphabets different from Latin, and even with scripts which run from right to left (Hebrew) or scripts which run vertically (Japanese), not to mention specific hyphenation patterns. That $\text{T}_{\text{E}}\text{X}$ allows for these usages might give an idea of its power.

From the publishers' point of view $\text{T}_{\text{E}}\text{X}$ has the potential of being used for producing complex scientific documents cost-effectively. This is the current practice of the AMS, and the American Physical Society, APS for short. They supply authors with

- user and installation guides
- fonts
- style files
- templates, and
- support, in general.

The advantages

The advantages can be summarized as

- high-quality craftsman-like tool
- lingua franca for exchange of typographically complex documents
- stability ($\text{T}_{\text{E}}\text{X}$ kernel has been frozen)

- open system
- available for all platforms
- in the public domain
- portable, flexible, extensible, ...
- 7 500–10 000 organized users world-wide
- cost-effective production tool.

Disadvantages

Are there any? Of course there are. But it is questionable whether one should talk about disadvantages. Perhaps one should talk more in terms of incompleteness.

What is felt like an omission can be added, because it is an extensible system.

To fill up the gap for `manmac.tex` – Knuth’s macros for formatting his books – I provided BLUE’s format system, based on MANMAC and which also accounts for developments since, as well as a user’s guide called ‘Publishing with T_EX.’ Of course there is the T_EXbook – the bible for the T_EXies – but that does *not* hide the details – it is all there, for the beginner as well as for the advanced macro writer – which is confusing and simply too much for a novice. In summary

- (L^A)T_EX is not WYSIWYG-like (‘What You See Is What You Get’)
- unusual macro language
- complex: ≈ a thousand commands, parameters, ...

So its incompleteness is a challenge to all of us, to fill it up.

It is true, however, that professionals have found some niches which deserve further research and development. Surveys on these items are provided in the ϵ -T_EX paper by Mittelbach, and the New Typesetting System (NTS) efforts initiated by the German-speaking users group DANTE E.V.. Also noteworthy is the effort to improve L^AT_EX through the so-called L^AT_EX3 (better known as l^{xiii}) project.

One can also argue that delving into these details is sub-optimization, concentrating too much on the mapping onto paper. Bigger issues are related to the multi-media aspects, let us say to represent information in a flexible way such that it can be processed by various technologies, into forms suited for various users, their circumstances and their tastes, limited only by their senses. I like to call this *real* applied information technology: information to be accessed by the masses.

1.4 T_EX’s flavors

T_EX has gotten its children already, like MANMAC, L^AT_EX, and AMST_EX/L^AT_EX, to name but a few. As usual for children they live their own lives. For T_EX this means that they have the confusing side-effect of not being completely compatible. In spite of this incompatibility reality has it that authors and publishers make their choice – T_EX-based, or L^AT_EX-oriented – and therefore the incompatibilities don’t hinder most of us.

`manmac.tex`

`manmac.tex` is a set of macros written and used by Knuth to format his magnum opus ‘The Art of Computer Programming,’ his Computers and Typesetting series, and so on. This collection has been extended into BLUE’s format system, which also accounts for developments since the design of MANMAC, and comes with the user’s guide ‘Publishing with T_EX.’

L^AT_EX

L^AT_EX stresses the higher-level approach of descriptive markup and hides the formatting details as much as possible from an author. Because of the rigorous way this has been implemented, it is hard to customize the prefab styles.

Leslie Lamport's manual, L^AT_EX, A Document Preparation System, exhibits the functionalities

- prefab styles: article, book, letter, report, slides
- automatic (symbolic) numbering and cross-referencing
- multi-column formatting, with its embedded 1-column occasionally for tables and figures
- automatic generation of Table of Contents, List of Tables, List of Figures
- picture environment
- bibliography environment.

Important add-ons are the Bézier curve macros, Makeindex to prepare indexes, and BIB_TE_X to manage bibliographic references.

AMST_EX/L^AT_EX

AMST_EX/L^AT_EX are the tools of the pace-setting American Mathematical Society. This publisher adopted and supported the T_EX development from the beginning. (See below under T_EX and its publishers.)

LAMST_EX

LAMST_EX reimplemented in a flexible way the descriptive L^AT_EX approach, next to a general automatic numbering and symbolic referencing scheme, advanced table macros, and sophisticated commutative diagram macros. See my review of Spivak's oeuvre for more details about the Joy of T_EX and LAMST_EX – The Synthesis. Another uniform approach to diagrams – spread, grid, and hierarchical – is provided by Gurari in his Draw_TE_X.

In summary

- manmac.tex, Knuth's format, extended into blue.tex, with PWT user's guide
- L^AT_EX, descriptive markup, and user's guide
- AMST_EX/L^AT_EX styles and fonts, with support
- LAMST_EX
- TUGboat styles
- PD software and working environments

1.5 T_EX's drivers

The drivers come with your T_EX system. With the PD versions, users have to be aware of the PD available drivers for the various PCs and printers. Well-known is the PD Beebe driver family. EM_TE_X comes with ample drivers.

At the TUG '92 meeting the attendees were surprised by Raman's paper 'An audio view of (L^A)T_EX documents.' It has all to do with representing the contents of a publication for the blind.

With respect to POSTSCRIPT the DVIPS driver is important. At this level epsf – encapsulated POSTSCRIPT – figures can be included.

Formerly, I also used DVITODVI in order to print out selected pages. Now I use MANMAC's facility to do that which is essentially simpler for that purpose because it ships out only the required pages. Modern drivers such as DVIPS and EM \TeX are also able to print out selected pages.

1.6 \TeX and fonts

From the beginning Knuth provided \TeX with the computer modern family of fonts. These fonts can be generated, and varied through METAFONT, by adjusting some parameters. Since the introduction of the virtual font concept, in revision '89 better known as \TeX version 3, many industrial fonts can be used as well. Through this mechanism, font elements can be combined at the driver level. The need for handling in a flexible way the positioning of diacritical marks was the incentive for adding the virtual font concept. The object being to make it feasible to handle languages with their own special placements of diacritical marks without the need to regenerate complete new fonts. An other possibility would be to generate complete font tables for every language, which is a perfectly acceptable, but will entail many font tables and of larger size.

However, since \TeX is used for more and more applications the need for more fonts – different shapes, sizes and so on – has emerged. Using standard bitmap technology much computer memory is needed. To compensate for this the Adobe's Type 1 outline fonts came along with POSTSCRIPT.

Outline fonts

Outline fonts have the advantage of

- rasterization is done by the output device (e.g. POSTSCRIPT printer)
- resolution independence
- compactness
- scalability
- rotation.

Reality has it that scaling fonts linearly does not yield pleasing results. To compensate for this the hinting technology emerged. The downloading of the fonts to the printer is done more intelligently nowadays. Only the characters that are used of the font are downloaded (this is called partial font downloading).

Font markup

Font markup has gotten a new dimension too: the linear space of available fonts is seen as a more-dimensional space governed by the coordinates family, serie, shape, and size. The approach goes with the buzzword NFSS, New Font Selection Scheme. (See 'The \LaTeX companion.)

Which fonts can be used with \TeX ?

The following classes of *text* fonts can be used with \TeX

- CM, the native Computer Modern
- many thousands of fonts in industry standard Adobe Type 1
- several thousands in formats such as TrueType.

(Very) few fonts can be used with math, because of the specialties of the font characteristics \TeX assumes. However, the following fonts can be used with math

- CM math, the native Computer Modern
- Lucida math
- Lucida newmath
- MathTimes.

For more details see Horn's Scalable outline fonts paper, and for Japanese Fujiura in TTN 1, 2.

And what about color?

Coupled to POSTSCRIPT (L)T_EX can yield results in color. Together with Adobe's illustrator or Corel-Draw these color figures can be edited and included in (L)T_EX as an epsf figure.

Drivers for Deskjet color printers are also available.

1.7 Descriptive markup

Since the start of computer-assisted typography attention has been paid to abstract from details, to embrace structure, and to adhere to the principle of the *separation of concerns*.

Leading in this area is the SGML approach. It is argued that

authors should concentrate on the contents – and inherently on the structure – of their documents, leaving the details for formatting to the publisher.

Example: Call for papers, Furuta

```
\input cfp.tex%Call for Papers format and macros
%next copy proper
The aim of this paper...

Paper are solicited on ...
\lstitm Picture editing
\lstitm Text processing
\lstitm Algorithms and software...

Detailed abstracts should not ...

Duration of presentation...
\bye
```

The above example is a mixture of natural input, where blank lines have an intuitive but context-dependent meaning, and of handling trivia automatically behind the scenes. An example of a default is the heading.

For this format the heading is always the same, so there is no need for a user to provide it each time the format is used. It comes along with the format. So do the fonts used and the shortcuts like `\def\lstitm{\item{--} }`.

My approach looks simpler than Furuta's – in that paper all the low-level formatting details were present – because I applied the principle of the separation of concerns and abstracted from the low-level formatting details. The point I'd like to make is that it is possible to hide formatting details, to account for these

separately and at a lower level. I like to call this approach generic, because the markup is customized at a lower level to the suited tool.

1.8 T_EX and its author

Don Knuth started the design of T_EX in 1978. The first major revision dates back to 1982. The final version is dated 1989, and called T_EX version π . Essentially this is version 3, but because reality has it that even Knuth ‘makes mistakes’ he allows for adjusted versions denoted by the decimals of π : 3.1, 3.14, 3.141, et cetera.

It is all a side-step(!) of his magnum opus: *The Art of Computer Programming*, of which three volumes have appeared of the envisioned seven. Because of the rapid development in computer science volume four consists of three books already.

In designing and developing T_EX, Knuth adhered to several software engineering paradigms like: portability, flexibility, robustness, and not to forget correctness and documentation. In order to do this gracefully he coined the words *literate programming*, and provided en-passant tools for practical use! In fact T_EX is a real-life and significant example of literate programming.

In designing T_EX he adopted and developed the following

- boxes, glue and penalties as building blocks
- paragraph-wise searching for line-breaks
- page mapping through the OTR (Output Routine), optimizing for least penalties
- device-independent output, to be printed, typeset, or viewed, by independent driver programs
- virtual fonts.

T_EX was developed as a side-step. METAFONT can be seen as an off-off-spring.

1.9 T_EX and its users

It is unknown how many people use (L^A)T_EX, and for what purposes. We know, however, that it is used all over the world, to typeset

- scientific documents, exchange and publish such documents
- documents that require special fonts and layout conventions, such as Japanese, Arabic and Hebrew
- transparencies and slides
- material associated with a hobby (bridge, chess, crosswords, go, music, and add yours).

The user groups

We also know that many users have organized themselves into user groups¹ to start with the original T_EX Users Group (TUG), and more recently as the so-called LUGs – local users groups. The Dutch speaking people are organized since 1988 as the NTG, *Nederlandstalige T_EX Gebruikersgroep*, that is the Dutch language-oriented T_EX Users Group.

World-wide some 7 500–10 000 users are organized.

The benefits of being organized, apart from cooperation and sharing in general, are

1. See Part 7 for more details.

- meetings
- TUGboat, newsletter, casu quo bulletins, ‘specials’
- resource directory (information about the (L)T_EX working environments of members, their addresses and similar things)
- assistance
 - (electronic) archives
 - electronic mailing lists
 - BBS (Bulletin Board Services)
 - digests
 - FAQs (Frequently Asked Questions)
- courses
- PD sets (Public Domain)
- distributing point books (tutorials), software.

Moreover, the user groups stimulate and support research and development, such as the projects: T_EXHaX, BIBT_EX, and more recently L^AT_EX3, and NTS. From the social side we have the TUG and EuroT_EX bursary funds, to grant attendance for a TUG or EuroT_EX meeting for those who can’t afford it, next to TUG’s Knuth Scholarship award. The latter is a competition that rewards the winner with attending a TUG meeting for free.

Add-ons

Add-ons have been provided by the user communities. They have also supplied mutual support, and have provided logistic facilities. The latter is not restricted to (L)T_EX proper. It is about the general use of the electronic networks

- exchange trough e-mail
- electronic digests and list servers
- the file servers (archives) that store macro and style files.

Really, very nice goodies! The proper add-ons concern

- porting the package to every system, especially the affordable and widespread PCs
- CTANS – Comprehensive T_EX Archives – with macro and style files
- extra fonts, casu quo font couplings trough virtual font scripts
- WYSIWYG user interfaces (commercial)
- (L)T_EX-based PD/shareware working environments
- language-specific issues (hyphenation patterns, reserved words, . . .)
- drivers for new printers, in general new devices
- POSTSCRIPT etc. inclusion at the dvi level.

And the end is not yet in sight.

1.10 T_EX and the publishers

The AMS is leading in how (L)T_EX can be used cost-effectively as a high-quality tool in a production environment: publishers cooperating with authors.

This approach has been followed by the American Physical Society.

At the TUG ’91 meeting at Boston, it was estimated that commercial publishers handle some 5 to 10% of their (scientific) production trough (L)T_EX.

And in the CIS – Commonwealth of Independent States, the former Russia—MIR has adopted the AMS approach as well. Undoubtedly more are to follow.

The American Mathematical Society

The AMS do their complete production through $\text{T}_{\text{E}}\text{X}$: $\approx 100\,000$ pages/year, and provide authors with

- preprint styles
- macros, and fonts (Euler, Fraktur, ...)
- user guides
- support (keyboarding, markup, fine-tuning).

For more details consult the AMS sources.

The American Physical Society

The American Physical Society handle some 20% of their production through $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. They cooperate with The Optical Society of America and the American Institute of Physics. Their style is called $\text{REV}\text{T}_{\text{E}}\text{X}$.

MIR

MIR publishers Moscow – the driving force behind CyrTUG, the Cyrillic language-oriented $\text{T}_{\text{E}}\text{X}$ users group – translated among other things TUGboat and Spivak's *The Joy of $\text{T}_{\text{E}}\text{X}$* into Russian. I would not be surprised to hear that they do the production of their scientific documents with $\text{T}_{\text{E}}\text{X}$ too, completely. They have the knowledge and $\text{T}_{\text{E}}\text{X}$ ology. The $\text{T}_{\text{E}}\text{X}$ - and METAFONT-based technology does not require much hard currency for investment.

JTUG?

And what is going on in Japan? The JTUG has at least 500 members. They have translated among others the $\text{T}_{\text{E}}\text{X}$ book and the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ manual into Japanese. Some years ago I received a Japanese newspaper set by $\text{J}\text{T}_{\text{E}}\text{X}$!

1.11 $\text{T}_{\text{E}}\text{X}$ and other Electronic Publishing tools

Furuta gives a good account of the history and early tools in relation with computer-assisted typesetting. However, since that paper

- the laser printer technology has taken off
- many computer-based fonts have emerged
- thinking in structures has gotten more interest (SGML)
- the DTP (Desktop Publishing) credo has come into existence
- POSTSCRIPT outline fonts, and PDF – Portable Document Format – were invented
- hardware prices have continued to spiral down.

Almost everybody can afford a PC, a laser(jet) printer, and some software (especially *Wordwhatever* or the PD ($\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$)).

T_EX and intelligent editors

Keyboarding compuscripts in (L^A)T_EX can be assisted by (L^A)T_EX intelligent editors, next to the use of templates as ‘fill-in’ forms. This approach reduces errors like the level 1 endings, or non-matching braces. At this level we can also make use of spelling checkers and style assistants.

Wordwhatever and T_EX?

It is true that Word-you-name-it, has made the use of computers more popular. They replaced the typewriters, don’t forget that. And of course that was a step forward. These are the tools the masses are using because of the sufficient and improved quality which can be obtained. This must be seen in context of course: most of the publications are just in-house reports, memos and the like.

For high-quality typesetting a T_EX-like tool, high-resolution fonts and ipso facto printer, or viewer, are needed.

Because wordprocessors are so widespread and heavily used, it can be anticipated that users start from there and need T_EX’s formatting capabilities now and then. For that group there exist conversion software. Simpler, and better when it concerns complex structured copy, is to

output in ASCII from *Wordwhatever*
and insert (L^A)T_EX markup.

And, of course, the wordprocessor can always be used as an editor for T_EX, with taking advantage of the integrated spelling checker.

Troff or T_EX?

TROFF preceded T_EX. It comes with UNIX. Both have been in use for the last decade. To begin with Knuth built upon TROFF, SCRIBE and similar tools. On the other hand the TROFF add-ons have learned from T_EX. So there has been mutual influence.

With respect to the functionality the tools are comparable. Both aim at computer-assisted typography. But there is also a world of difference. Basically the difference is that TROFF is a program which can be extended by independent preprocessors, and T_EX is an extensible language itself, with plain T_EX – the kernel program – device independent, that is the mapping on the media has to be done by independent drivers. That the latter was not in TROFF can be discerned from the subsequent NROFF – with accompanying NEQN – and finally, DI-ROFF, device-independent ROFF. Furthermore, remember that T_EX is just one of the twins.

Rumour has it that interest in TROFF weakened because the early PCs did not come with UNIX, and that the kernel has remained undocumented (Its author Ossanna died in an accident.)

SGML and T_EX?

SGML stands for Standardized Generalized Markup Language. It is an effort to formalize markup, and is defined as a meta-language to define the markup language of each publication series into so-called Document Type Definitions, DTDs for short.

SGML is part of a huge standardization effort supported by the US military through the CALS initiative. Other components are: FOSI – Formatted Output Specification Instance – and DSSSL. It is not so much a question of T_EX *or* SGML, but more T_EX *and* SGML.

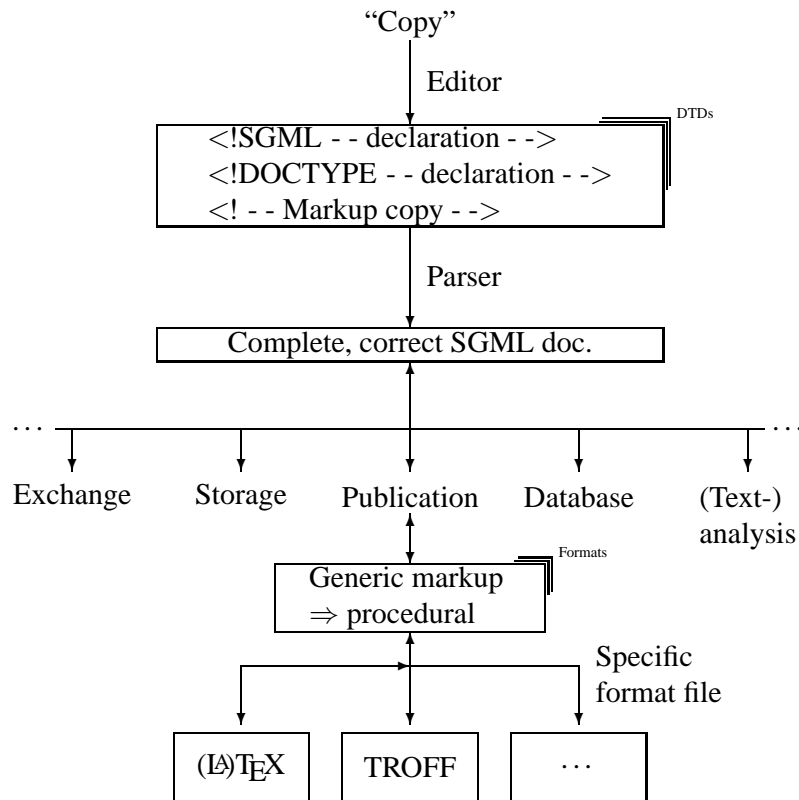


Figure 1.2: SGML

\TeX formats can learn a lot from the SGML approach and on the other hand SGML needs a formatter when it is used to print documents. This cooperative approach is known as

SGML the front-end, $(\text{\La})\text{\TeX}$ the back-end.

A diagram about the SGML- \TeX relation is supplied in the accompanying picture.

SGML and Hypermedia?

The following has taken from SGML FAQs and Personal Computer Word, March 1992

‘HyTime – Hypermedia/Time-based Structuring Language (ISO/IEC 10744). HyTime is a standard neutral markup language for representing hypertext, multimedia, hypermedia and time- and space-based documents in terms of their logical structure. Its purpose is to make hyperdocuments interoperable and maintainable over the long term. HyTime can be used to represent documents containing any combination of digital notations. HyTime is parsable as Standard Generalized Markup Language. HyTime was accepted as a full International Standard in spring 1992.

SGML’s hypermedia capabilities have been beefed up in the SGML standard extension HyTime. Although it started out in life as a specific set of standards for representing music, it was soon realized that these could be generalized for multimedia. HyTime provides

- SGML itself

- Extended Hyperdocument management facilities, including support for various types of hyperlink
- A Coordinate Addressing Facility which positions and synchronizes on-screen events. This allows authors to specify how hypermedia documents are to be rendered
- Better version-control of comments and activity-tracking policy support.

HyTime has been adapted as the basis for hyperlinking in the US Department of Defense's Interactive Electronic Technical Manual project. HyTime is an extension of SGML, providing a set of syntactic constructs: it doesn't specify a processing system.'

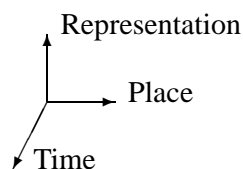
In the nineties www (World Wide Web) emerged, accompanied by the SGML-oriented language HTML (HyperText Markup Language) to mark up www pages. Following these developments \LaTeX to HTML converters were written (e.g. $\LaTeX2HTML$ and $\TeX4HT$). The big step forward of the www approach is that we can browse through a document with all what is linked to it, transparently, as if all is available at the connected server. The user is relieved from the connection protocols and moving to the right directories in order to access the linked document parts.

\TeX within the context of Electronic Publishing

When we think about Electronic Publishing we can't avoid being aware of the life-cycle of publications. This obeys the biological invariant: produce, consume and reuse.

Life-cycle: producing

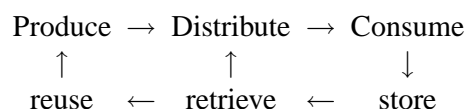
The production process has all to do with the dimensions



and with the characteristics

- representation of the contents, that is the typesetting proper aspects
- logistics, that is distribution and selling points – the place dimension
- reuse, that is the time aspect, when (parts of) document are reused.

The flow can be depicted trough

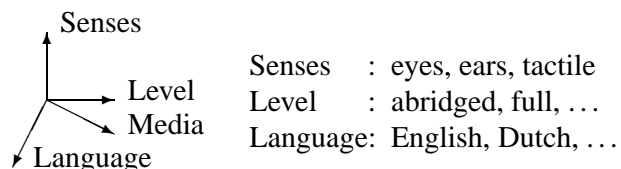


The big features are the unambiguous markup of copy trough $(\LaTeX)\TeX$ and the lifetime of the \TeX kernel. Therefore storing documents formatted by \TeX , leaves the reuse aspect open. Reality has it that documents formatted trough \TeX are easily redistributed trough the electronic networks, because it is all in ASCII, and \TeX is everywhere, so are its drivers.

My day-to-day reuse is transforming reports into articles and these into transparencies. In this work it is the other way round I'm recollecting elements I have set earlier.

Life-cycle: consuming

\TeX 's drivers have not paid attention to other representations as yet, although an exception is a driver for the blind. Difficulties in formatting languages different from English have been exercised in recent years. Undoubtedly research will be devoted to the aspects hinted at in the diagram given below with the dimensions



and with the characteristics

- choice of consumer language independent of the submitted language, that is automatic translation
- choice of representation, that is for example voice output from written submission.

Of course the above aspects will keep research busy for some time to come. This is the direction multi-media development will be going.

Adobe has concentrated on the consumer aspects, and launched in the mid-nineties its ACROBAT system. It abstracts from the document *preparation* system used. ACROBAT transforms POSTSCRIPT files into PDF files to facilitate exchange, because of the compact size of the file and the guaranteed availability of the fonts used through substitution of fonts by the multiple master technology. Moreover, the consumer can search over document boundaries, can link various documents (or parts of), and can choose from various representations for consuming. Bound to succeed not only because new functionalities have been added, but also because various drawbacks have been attacked and solved. Their ACROBAT reader has been released in the PD. The ACROBAT distiller (transforms POSTSCRIPT into PDF) and the ACROBAT catalogue (to provide for linking) and more packages to come, have their price. The latest development in \TeX makes it possible to generate PDF from \TeX , taking a short-cut around DVI and PS, by using PDF \TeX . Also new is that the freeware program GHOSTSCRIPT is now able to convert PS to PDF, as an alternative to ACROBAT distiller.

1.12 Trends

Adobe has been the trendsetter of the last decade with respect to new EP technologies.

I believe that the multi-media information technology will take off in the next century. Much is known under the buzzword hypertext. See the special issue of the Communications of the ACM for an introductory survey. As a \TeX ie it is fun to ponder about what niche there will be for \TeX . At the various TUG meetings people are concerned about the future of \TeX and share their doubts and optimisms. From that the following anthology

- \LaTeX is the future, forget about \TeX
- make $(\text{\La})\text{\TeX}$ available on low-cost machines
- embed \TeX etc. in working environments

- improve T_EX, in short keep it alive
- provide WYSIWYG user interfaces
- increase the number of (organized) (L)T_EX users
- get (L)T_EX accepted by publishers (formats, support, fonts, and the like)
- get (L)T_EX accepted by other communities: SGML, scientific societies
- provide user guides and templates
- education is paramount
- keep it simple and small is beautiful.

and so on.

Knuth secured a future for T_EX by embedding it as formatter in his CWEB literate programming system. The formatting is a subtask for designing, creating and maintaining quality software. By this he addressed the audience. Definitely not the casual layman.

Prophecy

The demand on information technology will be that

people can access cost-effectively, and easily, from their homes the information they need in a representation they wish.

I envision that the following technologies will influence each other in realizing the stated prophecy

- T_EX's role? Embedded in a Hypertext approach?
- increased self-publishing
- Electronic Production & Consumption
 - + photography
 - + CD
 - + TV/Radio, video
 - + PC
 - + phone, fax, email
 - + holography
 - + ...
- involvement of linguists and behaviorists

with the functionalities

- various inputs (o.a. voice, photography, ...)
- diverse outputs (language, level, media and representation,...).

Some years ago I day-dreamed about holographic-based true 3-D 'displays,' as a generalization of computer-assisted interactive TV. Science-fiction? Wait and see, or better hang on and make it happen!

Part 2

Δ TEX: a workbench for ms-dos PCs

This part is an adapted version of a paper presented at the European TEX conference in Gdańsk, September 1994.

Abstract

TEX and all its companions offer an enormous number of possibilities. This is both an advantage and a disadvantage. The advantage is that almost anything is possible; the disadvantage is that you need detailed knowledge of all related programs to fully exploit the possibilities.

The workbench Δ TEX is an attempt to integrate all major TEX related programs in a shell that shields you from the tedious and frustrating job of setting environment variables and program parameters. Δ TEX runs on MS-DOS but can also run in ‘DOS box’ on OS/2, LINUX or MS-WINDOWS. When running under WINDOWS 95/NT Δ TEX will use 32 bits Windows programs where possible.

Δ TEX includes the following tools (amongst others): TEX & METAFONT compilers, previewers, a spell-checker, BIBTEX, MAKEINDEX, TEXCAD, QFIG, graphics convertors such as HP2XX, BM2FONT, METAPOST and GHOSTSCRIPT, text convertors such as WP2L^ATEX and TROFF2TEX. Note that all programs used by Δ TEX are either free software or shareware.

Naturally, there is online help, and all functions are available through simple menu choices.

2.1 Introduction

Preparing documents with TEX or L^ATEX is not easy. However, those who persevere are rewarded with a beautifully typeset document. There are at least two reasons why writing with the ‘aid’ of TEX may cause some problems.

First of all there is the enormous number of possibilities that TEX offers. While this is one reason to use TEX in the first place, it has the disadvantage that many—at first puzzling and maybe hard to remember—commands have to be used. In principle this is a problem that you will have to live with. A thought to comfort you: many people have tried before you and most of them are by now enthusiastic TEX users.

The second reason why TEX may seem difficult to work with is of a more technical nature. Since TEX itself is only(!) a compiler, it needs several other programs and utilities to make a fully operational text preparation system. For example, at some time between conception and delivery of your document, you

might need an editor, a previewer or a printing program. All these programs require time to be correctly installed and may need setting of environment variables and/or some parameters to co-operate in the way you want them to. This is, of course, just the type of thing about which you do not want to bother. Therefore, here is the good news: you don't have to!

For those who do not like to spend their valuable time discovering all these technicalities, there is now \LaTeX . In its infant days, it did nothing more than constitute a simple menu for calling the \TeX compiler or an editor and the like. Although it is a relatively young program and is still growing, we feel that by now it is powerful enough to justify the title workbench. In the next sections you will find a description.

2.2 Principles of \LaTeX

The \LaTeX system consists of a host of separate programs, e.g. it uses $\text{em}\TeX$, the state-of-the-art \TeX implementation for MS-DOS and OS/2. It is named after its author, Eberhard Mattes. You must prepare your \TeX input file with an ASCII editor (we suggest you use TSE JUNIOR, TSE, or PFE); the actual \TeX compiler converts this input file into a .dvi file (from DeVice Independent); separate programs generate printed output from the .dvi file and allow you to preview the typeset page on your screen. In addition, there are a great many add-on utilities: a spell-checker; a database program for maintaining bibliographic references; an index generator; a utility to remove \TeX codes; a font generation program, extended graphics support etc.

Some of these programs, especially the various versions of the compiler and of the print and preview programs, require lots of parameters and/or environment variables. \LaTeX is designed to shield you from the dirty bits.

As mentioned above, the first objective in creating \LaTeX was to have some sort of integrating menu to call \TeX and the other main programs without the fuss of setting, let alone remembering, parameters.

One obvious way to do this is by using a batch file. However, plain `command.com` batch files tend to be slow, since only one command is read at a time. A much bigger draw-back is its lack of even the most basic commands for an interactive system. A very attractive alternative is provided by 4DOS (shareware by JP Software Inc.), nowadays a well-known replacement for `command.com`.

What once started as a small and simple batch file grew into a collection of batch files that currently comprises almost 15 000 lines of sometimes fairly sophisticated code. The main reason to program \LaTeX in the 4DOS batch language and not in some higher level programming language (e.g. Pascal or C) is that by using 4DOS we could create an open system, i.e. one in which anyone can modify the workbench to suit personal needs and taste without the need of special tools or extra compilers. Another reason to use the 4DOS batch language is the availability of environment variables and variable functions that enabled us to do things that would require very tedious programming in a higher level programming language. Since it is also fast (the complete batch file is read into memory at once), it was an easy decision to implement \LaTeX as a 4DOS batch file.

One might object that using 4DOS batch files deprives the old-fashioned `command.com` users from the benefits of \LaTeX . We happen to think that this would only be a mild punishment for not recognizing how good 4DOS really is. However, for those who have a good reason not to use 4DOS we also implemented \LaTeX to run under `command.com`. This is, of course, achieved by loading 4DOS as a secondary shell.

Though 4DOS is very powerful, some routines had to be written in other programming languages, e.g. to support mouse operation and to select files. Some free software and shareware programs are used, e.g. GWS, PAINTSHOP PRO, CSHOW, TSE JUNIOR and UNZIP.

2.2.1 Free software, shareware and commercial software

4 \TeX uses free software and shareware programs.¹

Using free software and shareware only, we are capable to distribute 4 \TeX without violating any law or agreement. As a 4 \TeX user you are supposed to pay for the shareware programs that you use. There is a complete list in the 4 \TeX documentation (see Chapter 8.). Remember that 4 \TeX could never have been built without these programs.

2.2.2 Setting up 4 \TeX

After installing 4 \TeX some customization may be required before 4 \TeX will run.

The most important files are `texuser.<os>`² and `system.set`. In these files all parameters that 4 \TeX needs are set. The file `system.set` contains values for the parameters that are needed for any user (e.g. what options does the compiler need, where are the fonts located), whereas `texuser.<os>` contains values for the parameters that are user-specific (e.g. where \TeX files are stored and what screen colors should be used).

As you might have guessed, 4 \TeX runs perfectly on a network. Because NOVELL NETWARE is the most popular type, 4 \TeX has an interface to use network printers. However, only a few lines of code in one of the batch files need to be changed to support this function on any other type of network.

Installing 4 \TeX from CD-rom is quite easy, i.e. just by running an installation script and answering a few questions the directory structure needed for 4 \TeX is created and the user specific settings are stored in the file `texuser.<os>`. The CD-rom contains an enormous amount of files that you probably do not need (e.g. the \TeX fonts for all the supported printers). It is possible to run the workbench from CD-rom or (partly) install it on your hard disk. This allows the user to choose the settings best suited for his/hers computer system, i.e. available hard disk space, completeness of the \TeX system, and performance of the workbench.

2.3 Features

In this section we will describe some features of the 4 \TeX workbench.

2.3.1 Add, delete, modify

4 \TeX aims to be an open system such that every user can add, delete or modify the 4 \TeX workbench to suit personal needs and taste. For instance, 4 \TeX uses Babel: a simple way to generate \TeX format files with multiple languages, and some control sequences to switch languages (and hyphenation patterns) within one \TeX document. Generating \TeX format files is done completely through menus that allow you to choose the basic format (e.g. plain, L \TeX , lollipop) and the languages you need. 4 \TeX calculates the memory requirements, generates the format and adds it to the list with its required parameters, ready to run.

1. We define free software as software that you don't have to pay for, shareware as software that you can try out for a fixed period of time and distribute freely. If you decide to keep using a shareware program you are supposed to pay for it. Commercial software is software that can only be obtained by paying for it.

2. The file extension `<os>` used here indicates the operating system that is used, i.e. `texuser.<os>` becomes `texuser.dos` for MS-DOS, `texuser.w31` for WINDOWS 3.1, `texuser.w95` for WINDOWS 95, and `texuser.nt` for WINDOWS NT

Of course any user or department has specific printers. \LaTeX currently supports more than 50 printer types including matrix printers, LaserJets, DeskJets and PostScript printers. Adding or deleting a printer type should not cause problems. \LaTeX lets you choose between local printers, network printers and printing to a file.

At this moment the spell-checker supports the languages Dutch, US English, UK English, French, German, Italian, South-African and Spanish but any other language can easily be added.

2.3.2 Help in \LaTeX

There are several types of automated assistance available. For each item in every \LaTeX menu there is a help screen.

The memory resident MS-DOS program `TEXHELP` is meant as a partial replacement for the \LaTeX 2.09 and \TeX manuals. It is a hypertext system that can be called from the editor. For example, if the cursor is at the word `\documentstyle`, pressing the `TEXHELP` key results in a help screen that refers you to the topic `Document Styles`.³ Select this topic and you get a help screen that gives information about all valid \LaTeX document styles and options. Moreover, there are cross references to all options and, among others, the commands `\flushbottom` and `\twocolumn`. It is also possible to jump directly to the index of `TEXHELP` or to review the last help screen. For most topics, the text is taken from the \LaTeX help system for MS-WINDOWS by M.F. Reid, which is based on G. Greenwade's help system for Vax-VMS computers. Under WINDOWS 95/NT one can use the Windows help file `latex2e.hlp` for help about \LaTeX 2 ϵ .

Several example files exist, varying from a standard \LaTeX file to more complex subjects such as tables with asymmetric columns or the creation of multiple indexes. Example files can be viewed from the editor and, if desired, inserted in the current document.

Finally, there is on-line \TeX documentation. This consists of documentation of e.g. style files. To save disk space, the documentation is kept in archives that are temporarily decompressed on selection.

2.3.3 Editing

We have chosen TSE JUNIOR (shareware by Semware) as the default editor in \LaTeX under MS-DOS. For this editor, we have developed many macros. For example, you can enter a \LaTeX environment defined by the commands `\begin{env} ... \end{env}` by picking it from a list; insert `\index{this}` behind the word `this` at the cursor position; or call the spell-checker (see below) to check the word at the cursor. Moreover, on loading a text file, the cursor will be placed at the exact location it was when you last edited the file (also available in the Block-View facility, see below). Of course \LaTeX allows you to use any editor you prefer. Under WINDOWS 95/NT one can use 32-bits windows editors (e.g. PFE, NOTEPAD etc.).

2.3.4 Block-View: quick partial compilation

One of the most often heard objections to \TeX is that it is not a so-called WYSIWYG system. On the other hand, \TeX users often reply that the acronym WYSIWYG is misleading: it should be WYSIAYG (What You See Is All You Get). We do not wish to solve this problem. Instead, \LaTeX offers a feature that we think both sides may like.

Imagine that you have just typed a very complicated formula. The natural way to check whether you did not make a mistake is to see what it looks like. Or maybe you are just curious. However, to see this

3. At the moment the \LaTeX 2 ϵ specific commands are not implemented.

single formula you would have to leave the editor, compile the entire document, start the previewer, and find the formula. Then you need to return to the editor, and find the exact position where you left it. If you made a mistake, you have to do this all over again.

In \LaTeX this whole procedure can be performed by the touch of a few keys. Even better, only the preamble of your document and the formula are compiled, which makes it as fast as possible. To view a certain part of a document, mark it as a block in the editor and press the ‘view-block’ key. Alternatively, if you want to view an entire file, just press the ‘view-file’ key. If you are making slides, it makes sense to view a complete slide at a time. Therefore, in this case \LaTeX automatically defines the before-mentioned block to be the current slide. If \TeX finds an error in the block, the user has the option to be returned to the editor at the line that contains the mistake.

2.3.5 Spell-checking

In \LaTeX we use `AMSPELL`, a public domain program by A. Merckens, to check and correct spelling in \TeX documents. `AMSPELL` is basically a spell-checker for plain ASCII files, with some special features for dealing with \TeX files.

The basic idea behind this program is to make spell-checking easier by

- providing the context of the possibly misspelled word;
- offering alternatives;
- offering facilities for editing the word;
- automatically replacing misspelled words in your document;
- learning new words.

`AMSPELL` does not require \TeX commands to be removed from your document (de \TeX ing). In fact, it will even interpret the standard accenting commands such as `\", \', \'` and will automatically use them while replacing misspelled words. When checking a \TeX file, `AMSPELL` will ignore all text between `$`'s and `$$`'s. Furthermore, `AMSPELL` will ignore parameters of the \LaTeX commands `\ref`, `\pageref`, `\cite`, `\nocite`, `\label`, and all text between `\begin{equation}`, `\begin{eqnarray}`, `\begin{eqnarray*}`, `\[` and their counterparts such as `\]`. You can change or expand the lists by means of environment variables. You can make your own file with correctly spelled words that are absent from the dictionaries.

2.3.6 Extended graphics support

\TeX was developed with the idea that it should be possible to have a \TeX implementation for every operating system (MS-DOS, VMS, UNIX etc.). Another important feature of \TeX is that documents can be freely exchanged between operating systems (because documents are written in standard ASCII). Graphics, however, are machine-dependent and the possibility to include graphics in \TeX or \LaTeX depends on the operating system and the DVI-driver you are using.

The solution often adopted for the inclusion of graphics is the incorporation of PostScript pictures in the document using the `\special` command. The `\special` command is ignored/passed on by the \TeX compiler but the PostScript DVI-driver will use the `\special` command to insert the PostScript picture at the right place and in the right size in your document. The disadvantage of this method is that you can only include PostScript pictures in your document and that you need a PostScript printer to produce output.

The `emTeX` DVI-drivers support a `\special` command to include black-and-white bitmapped pictures. Both this feature and the PostScript possibilities are used by \LaTeX to incorporate pictures in \TeX documents.

Graphic files come in many flavours. \LaTeX allows you to view, manipulate and include the following types of picture in your \TeX documents:

- bitmapped pictures: GIF, TIFF, PCX, BMP, IFF, LBM, IMG, CUT, JPEG, and PCL;
- vector pictures: HPGL and PostScript (textscaps).

This is done by using the following free software software: BM2FONT, HP2XX, PCLTOMSP, and GHOSTSCRIPT.

\LaTeX automatically chooses the appropriate conversion program, depending on the type of graphic file, or (to be more precise) on the file extension. Any conversion that \LaTeX performs will result in either \TeX fonts (accepted by any DVI-driver) or both a PCX and EPS file (for $\text{em}\TeX$ and PostScript resp.). Furthermore, a small \TeX file is produced that contains all the necessary commands to incorporate the picture in your document.

2.3.7 Printing

Currently \LaTeX supports more than 60 printer types including matrix printers, laser printers, inkjet printers and PostScript printers.

The standard Computer Modern fonts are available for most printer types. However, if any DVI-driver cannot find the required fonts, bitmaps will be generated ‘on the fly’. METAFONT is called if the METAFONT source is found, PS2PK is called if the PostScript source is found, all in one go. To accommodate PostScript users it is also possible to use GHOSTSCRIPT as your previewer. This way you can see if EPS pictures are included correctly.

2.3.8 Conversions to and from...

Sometimes you may want to convert files produced by other word processors to (\LaTeX) \TeX texts. Or you may want to use the extended ASCII set for accented letters instead of the less-readable \TeX equivalents.

\LaTeX supports a number of conversions from which we will mention but a few. WORDPERFECT / DISPLAYWRITE / TROFF / MS-WORD / PC-WRITE to (\LaTeX) \TeX , (\LaTeX) \TeX to ASCII ($\text{de}\TeX$), DVI to ASCII, $\langle\text{CR}\rangle\langle\text{LF}\rangle$ to $\langle\text{LF}\rangle$, $\langle\text{LF}\rangle$ to $\langle\text{CR}\rangle\langle\text{LF}\rangle$, TIB to BIB \TeX bibliography.

2.4 Availability

\LaTeX is available on CD-ROM. You can run \LaTeX from the CD-ROM after running the installation script. The script will generate all necessary files on your hard disk (less than 100 kB!). In less than one minute you are ready to run \LaTeX . This CD-ROM still is the world’s only turn key \TeX system for PCs!

2.5 Support

If you have trouble installing \LaTeX or need more information you can send E-mail to

`4TeX-support@eco.rug.nl`

However, don’t expect an answer within the hour. We will try to help you as soon and as best as we can, but \LaTeX is an ‘after-hours’ project.

\LaTeX users can join the \LaTeX mailing list. On this list, users can pose/answer questions regarding \LaTeX . New or desired developments and features are also announced and discussed on this list.

Subscribing to this list is very easy. Send the following message to `listserv@nic.surfnet.nl`
`subscribe 4tex Foo Bar`

where 'Foo Bar' is your real name. At the moment, about 225 people from 31 countries have joined the list.

Part 3

Installing \LaTeX

3.1 Introduction

This part is a guide to the \TeX installation as implemented in \LaTeX version 4.71. It does not describe the \TeX language, for that information, you can turn to various books listed in the bibliography, or look at the detailed documentation that can be found on the second CD-rom (4all \TeX Distribution sets + Documentation). This document should rather be considered a guide that gives you an overview of the \LaTeX system and directions how to use it. We don't aspire to completeness; full details can often be found in on-line documents.

Our chosen \TeX implementation is $\text{EM}\TeX$, which is public domain and state-of-the-art. It is named after its author, Eberhard Mattes.¹ \LaTeX 's power lies in the integration of extended graphics support, automatic format generation, automatic font generation, support for many different output devices, $\text{BIB}\TeX$, METAFONT , MAKEINDEX , AMSPELL and many other utilities.

A \TeX system consists of a host of separate programs: you must prepare your \TeX input file with an ASCII editor; the actual \TeX 'compiler' converts this input file into a `.dvi` file (from DeVice Independent); separate programs generate printed output from the `.dvi` file and allow you to preview the typeset page on your screen.

All of the programs included in \LaTeX , especially the various versions of the compiler and the print and preview programs, require *lots* of parameters and/or environment variables. The menu system \LaTeX is designed to shield you from managing these parameters and variables.

However, the standard installation may not fit your personal requirements, or you may be unhappy about the changes \LaTeX forces on their system setup. Therefore, we also include the basic information needed to set up one's own system, or to adapt \LaTeX to one's individual needs.

3.2 Typography conventions

In this document different typefaces are used to distinguish between names, commands, parameters etc. Here is a complete list with examples:

- Filenames: `myfile.tex`
- Paths: `c:\emtex\texinput`

1. The standard $\text{EM}\TeX$ package can be obtained through anonymous ftp from any CTAN (Comprehensive \TeX Archive Network) server (see Section 9.5 for details), directory `systems/msdos/emtex`.

- MS-DOS environment variables: COMSPEC
- Program names (not *file* names): WORDPERFECT
- MS-DOS commands:
 - ▷ dir *.*
 where the ▷ symbol represents the MS-DOS prompt.
- Keys on the keyboard: A
 Uppercase keys are typeset like this: ↑ Shift A Unless otherwise stated, upper case and lower case keys are equivalent. In the same way, we will for example use the notation Alt { to indicate that the user has to press the Alt key and the key that looks like { simultaneously. Keys may look different, depending on the nationality.
- Text as it is displayed on the screen: Unknown command
- Syntax descriptions: *SPDECOMP* <inputfile> [outputfile]
- In this manual we will assume that the drive $\mathcal{A}\text{T}\text{E}\text{X}$ is installed on is the δ : drive. E.g., if you run $\mathcal{A}\text{T}\text{E}\text{X}$ from CD-rom it will probably be drive D: or E:.

3.3 TEX and $\mathcal{A}\text{T}\text{E}\text{X}$

Preparing documents with TEX or $\mathcal{L}\text{A}\text{T}\text{E}\text{X}$ is not easy. However, those who keep trying can be rewarded with a beautifully typeset document. There are at least two reasons why writing with the ‘aid’ of TEX may cause some problems.

First of all there is the enormous number of possibilities that TEX offers. While this is one reason to use TEX in the first place, it has the disadvantage that many—at first strange looking and maybe hard to remember—commands have to be used. In principle this is a problem that you will have to live with. A thought to comfort you: many people have tried before you and most of them are by now enthusiastic TEX users. Also, there is an on-line help system.

The second reason why TEX may seem difficult to work with is of a more technical nature. Since TEX itself is only(!) a compiler, it needs several other programs and utilities to make a fully operational text preparation system. For example, at some time between conception and delivery of your document, you might need an editor, a previewer or a printing program. All these programs need setting of environment variables and/or some parameters to co-operate in the way you want them to. This is of course just the type of thing you do not want to bother about. Therefore, here is the good news: you don’t have to!

For those of you who do not like to spend their valuable time discovering all these technicalities, there is now $\mathcal{A}\text{T}\text{E}\text{X}$. While in its infant days, it did nothing more than constitute a simple menu for starting TEX or an editor and the like. With almost 7 years of experience and with many enthusiastic users all over the world we feel by now that $\mathcal{A}\text{T}\text{E}\text{X}$ has enough to justify the title ‘workbench’. In the following sections you will find a description.

3.4 A Historical Note, or ‘What’s in a Name?’

As mentioned above, the first objective in creating $\mathcal{A}\text{T}\text{E}\text{X}$ was to have some sort of integrating menu to call TEX and the other main programs without the fuss of setting parameters.

One obvious way to do this is by a batch file. However, plain `command.com` batch files tend to be slow, since only one command is read and executed at a time. A very attractive alternative is provided by 4DOS (shareware by JP Software Inc.), nowadays a well known replacement for `command.com`. Among

other things, this program offers an extensive batch file language with many predefined functions and variables. Since it is also fast (the complete batch file is read into memory at once), it was decided to implement \LaTeX as a 4DOS² batch file (denoted with an extension `.btm`, an abbreviation of Batch To Memory).

One might object that using 4DOS batch files deprives the old-fashioned `command.com` users from the benefits of \LaTeX . We happen to think that this would only be a mild punishment for not recognizing how good 4DOS really is. However, for those who have a good reason not to use 4DOS we have also implemented \LaTeX to run under `command.com`. This is of course done by loading 4DOS as a secondary shell.

Starting off as a small and simple batch file, \LaTeX became a collection of batch files consisting of about 15 000 lines of sometimes fairly sophisticated code.

The main reason to program \LaTeX in the 4DOS batch language and not in some higher level programming language (e.g. Pascal or C) is that by using 4DOS we could create an open system, i.e. one in which anyone can correct bugs or modify the workbench to suit personal needs and taste without the need of special tools or extra compilers (especially useful when one wants to assure quick and adequate support). Another reason to use the 4DOS batch language is the availability of environment variables and variable functions that enabled us to do things that would require very tedious programming in a higher level programming language. Since it is also fast (the complete batch file is read into memory at once), it was an easy decision to implement \LaTeX as a 4DOS batch file.

3.5 Principles of \LaTeX installation from CD-rom

Although there is an old diskette version of \LaTeX the principles of installation described here are solely about installation from CD-rom. Installation from CD-rom is easy because it is almost completely automated.

\LaTeX version 4.71 usually does not't require any alteration to the files `config.sys` and `autoexec.bat` (i.e., for MS-DOS users). However, we advise MS-DOS users you to add the line

```
mode lpt1 retry=r
```

to your `autoexec.bat`. This line will ensure that printing within the \LaTeX workbench will work correctly. In order to prevent strange output after ending \LaTeX we suggest adding `mode bw80` or `mode co80` to the `autoexec.bat` (depending on whether you have a monochrome or a color monitor). In rare cases you may have to add the line `device = ansi.sys` to your `config.sys`. \LaTeX will test for ANSI support and inform you if any change is needed.

3.5.1 Installation from CD-rom

In the root of the \LaTeX workbench CD-rom you can find several files that are worth reading:

2. For technical reasons \LaTeX needs 4DOS version 6.0 or higher to run.

<code>readme.txt</code>	some information about the fourth edition of the CD-rom
<code>readme.htm</code>	the same contents as in <code>readme.txt</code> but in HTML format with hyperlinks to other documents (to be used with a WWW browser)
<code>index.htm</code>	an HTML file that lists all files in the current directory and links to all other directories and files on the CD-rom (to be used with a WWW browser)
<code>licence.art</code>	some software on the CD-rom uses this file to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications
<code>licence.gnu</code>	the same as with <code>licence.art</code> many software is distributed under the GNU license statement
<code>dir.lst</code>	a list of all the directories on the CD-rom
<code>files.lst</code>	a complete list of all the files and directories on the CD-rom
<code>install.txt</code>	this file gives some information on how to install \LaTeX
<code>install.bat</code>	the \LaTeX installation program (see below for more details)

The \LaTeX installation program generates some files that are adapted specifically to the operating system your computer is running. These files will have a file extension that indicates the operating system that is used. For instance a file containing values for the parameters that are user-specific (e.g. where \TeX files are stored, what screen colors should be used) is created. This file will be indicated in this manual as `texuser.<os>` and the file extension `<os>` used here indicates the operating system, i.e. `texuser.<os>` becomes `texuser.dos` for MS-DOS or OS/2, `texuser.w31` for WINDOWS 3.1, `texuser.w95` for WINDOWS 95, and `texuser.nt` for WINDOWS NT.

In the root of the \LaTeX workbench CD-rom the file `install.bat` will perform an automatic installation of \LaTeX . This means that you do not need to create directories and set environment variables yourself but only have to go to the root of the CD-rom drive and type `install` (Windows 95 users can click the “Start” button, and then click “Run”. In the Open box, type `δ:\install` where `δ` is the CD-rom drive letter). The installation program will check the operating system you are using and ask if you want to change the automatic detection to something else (e.g. from WINDOWS 95 to MS-DOS).

Then the installation will set up the following directory structure:

```
x\texfiles
x\texfiles\4system
x\texfiles\bib
x\texfiles\fonts
x\texfiles\mf
x\texfiles\mp
x\texfiles\pictures
x\texfiles\psfonts
x\texfiles\spell
x\texfiles\texinput
x\texfiles\tfm
x\texfiles\vf
```

where `x` is a path that you have to supply, e.g., `c:\` or `d:\text`. When you specify a path that does not exist the installation program will prompt if the supplied path has to be created.

After creating the directory structure \LaTeX will check whether or not your video card is VESA compatible. This is used in combination with the $\text{EM}\TeX$ viewers to switch to a higher screen resolution (default is 640 by 480 points).

The installation program will then ask if you want:

0. A minimal \LaTeX installation, i.e. only the user specific files are stored on the harddisk and everything else is read from CD-rom (this will take about 100 kilobytes of disk space)
1. Not only a minimal installation but also copy the \LaTeX batch files to harddisk. Since these batch files are used in the \LaTeX system, reading them from harddisk instead of CD-rom will speed-up \LaTeX performance (this will take about 600 kilobytes of disk space)
2. Install \LaTeX on the hard disk and hence get maximum performance and/or independency of the CD-rom. The installation program will ask you which modules/parts have to be installed to harddisk. Disk space required is 50 to 300 megabytes, depending on your preferences. \LaTeX can be stripped down further afterwards if you like, or you can install other parts later.

The next step is choosing which editor you will use. We highly recommend the TSE editor (shareware) for MS-DOS and the PFE editor for WINDOWS 95/NT (freeware), but many other editors are supplied. In case none of these editors suite you, you can specify your own.

Because much documentation on the CD-rom is supplied in PDF and HTML format you are prompted to supply a HTML-viewer and a PDF-viewer. If you do not have one of them already installed you can choose one of the options in the selection menu. If you already have a viewer installed you can select this one by supplying the complete path and program name.³ If you do not know what to choose then do not install a viewer at this point. All the user settings are stored in the file `texuser.<os>` and can be changed/updated at any time later on.

To complete the installation the file `texuser.<os>` will be created in `x\texfiles\4system\` and adapted to your setup. Then the files necessary for automatic font generation under \LaTeX are copied (i.e. the files `fontmap`, `psfonts.map`, and `psfonts.inf` are copied to `x\texfiles\4system`). After copying some user files for the amSpell spell-checker (to `x\texfiles\spell`), the \LaTeX batch file `x\texfiles\4system\4tex<os>.bat` is created.⁴ You can start \LaTeX simply by typing/running `x\texfiles\4system\4tex<os>`. To complete the installation some tests are done and the results are displayed on screen and stored in `x\texfiles\4system\install.<os>`.

Note: The \LaTeX batch file `x\texfiles\4system\4tex<os>.bat` enters three environment variable settings and then starts (with 4DOS) the real \LaTeX batch file `\emtex\btm\4tex.btm`. If you decide to move the \LaTeX user files to other directories you will have to update these three environment variables. The first variable `4TEXPATH` indicates the file `texuser.<os>` file is stored. The second variable `4DOSPATH` indicated where the 4DOS command file can be found. The third environment variable `4TEXOS` indicates which operating system \LaTeX will run on (0 is MS-DOS or OS/2, 1 is WINDOWS 3.X, 2 is WINDOWS 95, and 3 is WINDOWS NT).

Note: We suggest 4DOS users to create an alias. For example

```
▷ alias 4tex=pushd^c:\texfiles\4system\4texDOS %&^popd
```

allows the 4DOS user to

- start \LaTeX from any subdirectory on any drive and
- return to this subdirectory after leaving \LaTeX ,

3. E.g. `c:\progra 1\intern 1\iexplore.exe` for the Internet explorer or `c:\progr 1\netscape\naviga 1\program\netscape.exe` for the Netscape Navigator, and `c:\Acrobat3\Reader\Acroord32.exe` for the Acrobat reader.

4. `x\texfiles\4system\4texdos.bat` for MS-DOS `x\texfiles\4system\4texw31.bat` for WINDOWS 3.1 `x\texfiles\4system\4texw95.bat` for WINDOWS 95 `x\texfiles\4system\4texnt.bat` for WINDOWS NT.

simply by giving the command

```
▷ 4tex
```

Note: WINDOWS 95/NT users can create a shortcut on the desktop by clicking with the right mouse button on the desktop and selecting “New”, “Shortcut”. Then browse to the \LaTeX batch file or simply type `x\texfiles\4system\4texw95.bat`. Type the name for the shortcut (default is `4TeXW95`) and select an icon. After creating the shortcut some fine-tuning is worth considering: click with the right mouse button on the \LaTeX icon and select “Properties”, then on the “Program” tab sheet; select “close on exit”. You can also change the icon to the \LaTeX icon by clicking on the “change icon” button and select “Browse” and select the icon `x\texfiles\4system\4tex4.ico`. In the “Screen” tab sheet you can unselect the “Toolbar” setting.

Note: WINDOWS 95/NT users should not quit \LaTeX by clicking on the `x` in the upper-right corner of the screen, but by using the menu-option “Quit”. If you not correctly close \LaTeX some temporary files will not be deleted and the next time you start \LaTeX it will complain that it thinks that you are already running \LaTeX . If this is not the case press the `[S]` to start \LaTeX anyway.

3.5.2 Installation on your hard disk

In case you chose to “Run \LaTeX entirely from hard disk” the installation program will copy all necessary files from the CD-rom to your hard disk, so you will not need the CD-rom to run \LaTeX .

Although hard disk space is quite cheap these days it doesn’t make sense to simply copy the entire content of the CD-rom to your hard disk. To enable you to setup a system in a more flexibel way we defined a set of modules. These (ASCII) `.mod`⁵ files are stored in directory `δ:\inst` and they can be read and processed by the program `insthd.bat`, which you can find in the same directory.

INSTHD expects two parameters. The first is the name (including path if not in the current directory) of a `.mod` file. The second parameter is the directory which is the root of your \LaTeX installation.

We advise you to run the program always from the directory `δ:\inst` to make sure it can find all necessary files. To install e.g., the \LaTeX 2 ϵ module you could enter the following command:

```
▷ INSTHD latex2e.mod c:\emtex
```

Note that the directory `c:\emtex` must already exist.

The file `latex2e.mod` looks like this:

```
LaTeX 2e formats and input files (complete) (13 MB)
require ltxbase.mod
latex\*.* /s
```

The first line is simply a comment that tells you what this module is supposed to install. All other lines either specify a set of files (possibly including files in subdirectory if followed by `/s`), or another `.mod` file which is required to make this module work. INSTHD processes a module line by line, but you can make any mix of “require” statements and file specifications. The number of lines in one module is unlimited. Note that INSTHD processes a “require” statement by recursively calling itself. Even multiple recursions (in case the required module requires yet other module, etc.) are possible.

The file specification(s) in a `.mod` file are relative to the directory `\emtex` on the CD-rom, so in the example above the third line specifies that all files stored on the CD-rom in directory `\emtex\latex`

5. Do not confuse these modules with digital music files (“mods”) which also have a file extension `.mod`.

and all subdirectories will be copied. Note that the size indication is a measure only for the files specified in that particular `.mod` file. The effect of installing required modules is not accounted for.

You can install modules at any time, so if you are not sure at a certain moment if you need a specific module, you can try without first.

We hope the set of `.mod` files we supplied is sufficient for your needs. However, we just might not have anticipated your exact needs. Therefore we encourage you to write your own modules and please send us copies so we can distribute them to others.

3.5.3 Fine-tuning \LaTeX to your taste

After installing \LaTeX , or when your preferences have changed you may want to modify the file `texuser.<os>` and perhaps `system.set`. Take your time reading these files and adjust them to your own personal needs. We advise you to save a copy of the original settings just in case something goes wrong.

The `texuser.<os>` and `system.set` files contain assignments of so called *environment variables*.⁶ Environment variables provide a means to adapt \LaTeX to your own taste. Both files are self-documenting. The file `system.set` can be found in the directory `\emtex\btm` if you did not install the \LaTeX batch files to harddisk or in `x\texfiles\4system\btm` if you installed the \LaTeX batch files to hard disk. The file `texuser.<os>` can be found in the directory `x\texfiles\4system`.

Using any ASCII editor, modify the files `texuser.<os>` and `system.set` to suit your personal preferences. Some variables *must* be given a (new) value, while others can be left empty or unchanged to take on default values. This is indicated for each variable separately in the files `texuser.<os>` and `system.set`. The environment settings are used within \LaTeX and wrong settings of these variables can result in unpredictable behavior of the \LaTeX workbench. If strange errors occur while running \LaTeX , check the two files containing the environment settings. For instance specifying a too long `PATH` variable (i.e. longer than 120 characters) can confuse some programs to the point of crashing.

Because `texuser.<os>` and `system.set` are self-documenting we will not explain here how each variable must be set. We will, however, give an example of how to change an environment variable to suit your personal needs.

The file `texuser.<os>` includes the lines

```
STYLE=lplain
: specifies TeX format file.
: for LaTeX      : use latex2e
: for LaTeX 2.09: use lplain
: " with NFSS   : use lplainn
: for plain TeX : use plain
: Format file xx is available if there is a xx.frm file in
: %EMTEXDIR\compiler.
: (optional, default: latex2e)
```

This means that after starting the \LaTeX workbench the variable `STYLE` refers to the \TeX format file that will be used when compiling a \TeX document. If omitted \LaTeX will use as default the \LaTeX format file (i.e. `latex2e`). If you want the plain \TeX format to be default you simply use an ASCII editor to change `STYLE=` into `STYLE=plain`. In this example `STYLE=lplain` means the old \LaTeX 2.09 format will be your default format.

6. Use and meaning of environment variables are explained in any MS-DOS manual or book.

Two environment files are used to separate *general* settings from *user specific* settings. This is especially of interest for network installations of \LaTeX where each user has his/her own preferences (i.e. `texuser.<os>`). The file `system.set`, however, contains the general settings that should be the same for any (network) user. This also means that the file `texuser.<os>` can be placed anywhere on any drive, whereas the file `system.set` always remains in the directory `\emtex\btm`. You can use the environment variable `4TEXPATH` to indicate where \LaTeX should look for the file `texuser.<os>`. If not specified \LaTeX will look in the current directory. If you specified e.g., `4TEXPATH=C:\TEXFILES\4SYSTEM`, \LaTeX will look for `texuser.<os>` in the directory `c:\texfiles\4system`.

\LaTeX will use a shareware version of 4DOS. If you have a newer version or even better a registered version of 4DOS the variable `4DOSPATH` will specify where to find 4DOS; e.g., `4TEXPATH=c:\4dos60` will use this 4DOS version within \LaTeX .

If strange errors occur, check the files `texuser.<os>` and `system.set`. Check e.g., in the file `texuser.<os>` if the following variables are set correctly: `EMTEXDIR=...` and `NETWORK=`. Take your time to check all the other variables (they are self-explaining), e.g., check them by looking if the paths and programs you are specifying do exist. After this you are ready to (re)start \LaTeX .

3.5.4 Network setup

For networked installations of \LaTeX the following files may need editing:

- `\emtex\btm\network.btm`
- `\emtex\prndest\capt*.pr_`
- `g:\texfiles\4system\texuser.<os>`

We will assume that the network you are using is a Novell network, and you want to install \LaTeX for network use. Installing \LaTeX on other networks is also possible (e.g., on Bayan or Decnet). Start the installation by adapting the file `network.btm` to your own situation. This file contains the network drive mappings and all other network specific commands used by \LaTeX , such as commands for selecting network printers.

Network printers can be made accessible by renaming the files `capt*.pr_` to `capt*.prd` (all files with extension `.prd` will show up in the output menu when choosing a print destination). Of course these files have to be adapted to your network setup. The versions supplied by \LaTeX only show examples of possible solutions as they were implemented on a specific Novell network.

Check in the file `texuser.<os>` if the following variables are set correctly: `EMTEXDIR=...` and `NETWORK=y`. All other variables of `texuser.<os>` should be correct. file `network.btm`, needs adjustment if you want to add support for network printers..

Then copy `texuser.<os>` to one of the directories of the network user (e.g., to `g:\texfiles\4system`) and adjust this file to the user's needs. Make a \LaTeX startup batch file, e.g., `tex.bat`, containing:

```
@echo off
set 4texpath=g:\texfiles\4system
REM the environment variable 4texpath specifies the path
REM where the file texuser.[os] can be found
set 4DOSPATH=w:\appls\emtex\utils\4dos
set 4TEXOS=2
REM drive w: is where TeX is installed, i.e.
REM directory \appls\emtex. and 4TEXOS stands here for Windows 95
mode co80
w:\appls\emtex\utils\4dos\4dos @w:\appls\emtex\btm\4tex.ini
```



```

/c w:\appls\emtex\btm\4tex.btm %1 %2 %3
REM Note: that should be ONE long line in reality!
set 4TEXPATH=
set 4DOSPATH=
set 4TEXOS=

```

The network user now starts the 4 \TeX workbench by simply typing

```
▷ tex
```

(with or without parameters). 4 \TeX will use all user specific information stored in `texuser.<os>` and run from the network drive.

It is *not* a good idea to copy `4tex.btm` to the directory `g:\texfiles\4system`. At irregular times updates of 4 \TeX may be installed on the network. Using your own copy of `4tex.btm` deprives you of automatically using the latest version and may cause errors.

3.5.5 Directory set up

The 4 \TeX workbench is set up in the directory `δ:\emtex` and has the following directory structure:

EMTEX	EMTEX root directory
BTM	4TeX workbench files
CONVERT	4TeX batch files for conversions
UTILS	4TeX batch files for utility menus
FORMATS	files for format file generation
COMPILER	emTeX compilers plus format information
LATEX	LaTeX 2e input files with many subdirs
...	all packages are stored in separate directories
LATEXALL	LaTeX 2e input files all in one directory
INPUTS	generic input files
...	where possible packages are created
LATEX209	LaTeX 2.09 style files
...	where possible packages are created
BTEXFMTS	big TeX format files
HTEXFMTS	huge TeX format files
DATA	emTeX DVI driver configuration files
TEXFONTS	fonts
VF	virtual fonts
LASERJET	bitmapped fonts for LaserJet
DESKJET	bitmapped fonts for DeskJet
VGA	bitmapped fonts for VGA screen
...	many printers have their own fonts
PS	DVI to PostScript
FONTS	PostScript fonts
TFM	TeX font Metric files
...	split into groups/families when possible
TFMALL	TeX font Metric files all in one directory
METAFONT	font generation
BMFBASES	big Metafont base files
MFINPUT	Metafont input files
...	split into groups when possible
MFJOB	Metafont job files
METAPOST	Metapost program
MPINPUTS	Metapost input files

MISCPROG	miscellaneous programs
OS2	OS2 help files
PRNDEST	printer/viewer types and print destinations
SPELL	amSpell spell-check program
DOC	emTeX and TeX documentation
...	grouped by topic
TEXSAMPL	sample TeX files
UTILS	TeX related utilities
GSTOOLS	Ghostscript the PostScript interpreter
4DOS	4DOS the command.com replacement
WIN	Windows 95/NT programs/utilities
...	many useful Windows/Dos programs
TEXFILES	directory for TeX text files
4SYSTEM	setup files
BIB	bibTeX bibliography files
FONTS	user generated bitmapped fonts
LASERJET	bitmapped fonts for LaserJet
DESKJET	bitmapped fonts for DeskJet
...	many other printers
MF	user specific Metafont input files
MP	user specific Metapost input files
PSFONTS	user directory for PS fonts
PICTURES	user directory for pictures
SPELL	user specific files for amSpell
TEXINPUT	user specific style files
TFM	user specific font metric files
VF	user specific virtual font files

3.6 Starting 4TeX

Assuming that the command `4tex` invokes 4TeX, the following syntax applies

Starts the 4TeX workbench.

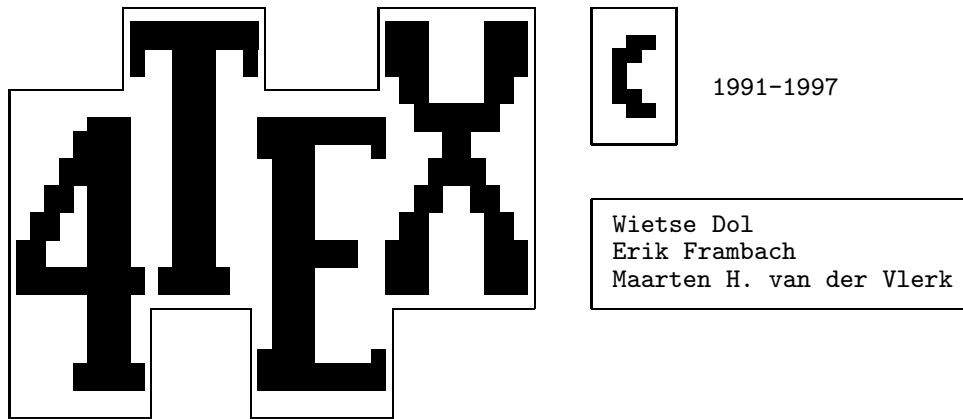
```
4TEX [[[path]main[.ext]] [include[.ext]]] | [/r | /?]
```

```
main      = name of TeX file (with or without extension)
include   = name of TeX include file ( " " )
/r        = restart (no quick startup, perform all checking)
/?        = this help screen
```

examples:

```
4TEX article                4TEX c:\texfiles\*
4TEX article.ltx            4TEX book.tex chap1
4TEX article.tex graph1.pic 4TEX book thesis.sty
4TEX g:\tex\book chap*      4TEX /r
```

Note: 4TEX without parameters will run the quick startup using the configuration from the last 4TeX session (if available).



4TeX uses shareware programs. If you use them please register!
Read the manual for details...

Figure 3.1: 4TeX opening screen

Enter

▷ 4tex /?

to view this explanation of the syntax on the screen.

When specifying the ‘Main $\text{T}_{\text{E}}\text{X}$ file’ you can add a path and even use wildcards. The file extension is optional, default is `.tex`. If no ‘Main $\text{T}_{\text{E}}\text{X}$ file’ is specified, the ‘Main 4TeX menu’ will prompt for one. It is also possible to specify an ‘Include $\text{T}_{\text{E}}\text{X}$ file’, with or without wildcards. The ‘Include $\text{T}_{\text{E}}\text{X}$ file’ has no path and can only be specified after you have given a ‘Main $\text{T}_{\text{E}}\text{X}$ file’. When using wildcards, 4TeX will look for all possible files that match the specification and will give a selection menu if there is more than one matching file.

The `/r` option is meant as a ‘reset’ option. If you start 4TeX without parameters it will attempt to rebuild the menu as it was the last time you left it. For that purpose 4TeX reads and writes a file called `4texpars.<os>`. Using this file 4TeX can start up much faster because many checks are skipped. However, if for some reason 4TeX crashed, you want to make a fresh start. This can be achieved by supplying the `/r` parameter (in fact by supplying any parameter).

Part 4

4_lTeX menus

Let us assume that you want to start working on the first chapter of a new book. At the command line you type

```
▷ 4tex book chap1
```

After a few seconds, in which 4_lTeX does some book-keeping, a screen similar to that in Figure 4.1 appears.

F1 is the general help key. The result of pressing this key depends on which options are installed. For each command in the main menu (and also in the other menus) there is also more specific help. The way to activate it is to press the **Alt** key in combination with the highlighted key in the name of the command. For example, to get help on the `change Main TeX file` command, press **Alt M**. After pressing the general help key **F1**, it is also possible to view all specific help screens that go with the current menu. Use **PgUp** and **PgDn** to browse, any other key to exit from the help system.

4.1 The main menu

Before a detailed description of all the commands in the main menu is given, we will first explain some of the terminology used in 4_lTeX.

The ‘main TeX file’ is the file containing the preamble (i.e., definitions, macros and the like) and maybe several `\include` or `\input` commands. In our example this is the file `book.tex`.

An ‘include TeX file’ is a file that is included in the main file, e.g., `chap1.tex`. Such a file typically contains part of the text that is to be typeset by TeX, and here the definitions and macros of the main file are actually used. If you are working on a large document it makes sense to cut up the complete text into logical parts like chapters or sections, each in a separate include file. Of course, if the document at hand is only a small one, the text can be part of the main TeX file and no include TeX file needs to be defined.

The ‘TeX files path’ is the subdirectory where TeX looks for the main and include files.

The remainder of this section will describe the key combinations that can be used in the main menu:

M `change Main TeX file`

clears the field `main TeX file` (and also the field `include TeX file`), and another main file can be chosen. There are essentially three different ways to do this:

- Type the name of a TeX file and press **Enter ↵**. If there already exists a file with the same name in another subdirectory, a selection menu will appear where you can indicate which file you want to work with. If a file in some other subdirectory is chosen, the `TeX files path` will be set accordingly.

```

main TeX file      : book      .tex
include TeX file   : chap1     .tex
TeX files path     : C:\TEXFILES\4TEXDOC
backup path        : a:\
TeX format         : LaTeX 2e + Babel

```

main menu:

change Main TeX file	choose TeX Format	check Spelling
change Include file	Compile TeX file	run Utilities
change TeX files Path	show Logfile	execute Dos command
change files eXtensions	manage Output	Backup TeX files
Edit file(s)	Quit	

press highlighted key or
 press [F1] or [Alt] highlighted key for help
 press [?] for info on 4TeX

make your choice...

(c) 4U 1991-1997

Figure 4.1: Main menu

- Press on the empty field. A selection menu will appear, showing all files in the current TeX files path with the currently chosen file extension. Also, all subdirectories and the ‘mother’ directory are shown. Use the cursor keys and to search the directory tree and select a file. To make it easier to remember what is in a file, the first line of the file or the 4DOS description is also shown (depending on the setting of FINFO in `texuser.<os>`).
- The third possibility is a mixture of the previous two. Typing a partial filename followed by a and pressing (plus can be abbreviated to) results in a selection menu showing only the files matching this mask.

Note: using the key at any moment during the procedure described above will restore the previous situation. This remark applies to all similar situations, except for change Include TeX file.

change Include TeX file

similar to change Main TeX file, except that on pressing the field include TeX file will be cleared.

change TeX files path

clears the field TeX files path and another subdirectory can be chosen. If you enter a directory name with wildcards (or an empty one) 4TeX will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Press to restore the directory name to the old setting.

change files eXtensions

By default the file extensions of both the main and the include file are `.tex`. You can change them in case you use different extensions like `.ltx`, or perhaps to simultaneously edit a \LaTeX file and a \LaTeX style file (`.sty`).

E Edit file(s)

invokes the editor chosen in `texuser.<os>`. Both the main file and the include file are loaded into the editor, the latter is shown and ready for editing.

Although it is possible to use any ASCII editor you like, we recommend MS-DOS-users to use TSE JUNIOR (shareware by Semware) or TSE (commercial software by Semware). At the moment these are the only editors that are fully supported by \LaTeX , in the sense that the facilities described in Section 5.1 are only implemented for TSE JUNIOR and TSE. WINDOWS 95/NT-users we recommend to use the PFE editor (freeware).

F choose TeX Format

presents a selection menu showing all the formats (such as plain TeX and \LaTeX) that are installed on your system. Select by using the cursor keys and press **Enter**. The difference between the *huge* versions and the normal format versions is that with *huge* the huge EMTeX compiler (larger memory settings) are used when compiling your document.

C Compile TeX file

compiles the main TeX file using the current TeX format. TeX writes all sorts of information on the screen, for instance page numbers and the currently processed include file. If an error occurs, TeX interrupts processing and asks the user to intervene. Press **F1** or **H** to get help. One possibility is to press **E** (followed by **Enter**). This has the following effects

- i) compilation is aborted
- ii) the main TeX file and the include file that contains the error are loaded into the editor. Note that this is not necessarily the include file that was selected as the include TeX file before compilation.
- iii) the cursor is positioned in the file at the beginning of the line containing the error.

The user can correct the error and start compilation once more.

Note: This option can also be used in block compiling (see Section 5.1).

L show Logfile

shows the logfile created by TeX during the last compilation of the main TeX file. This file contains all remarks and error messages shown on the screen during compilation and a bit more (statistics on memory usage etc.). Under MS-DOS the file is displayed using the internal 4DOS command list. Press **Esc** to quit. Under WINDOWS 95/NT the program QUICKEDIT is used instead.

O manage Output

loads the \LaTeX output menu that provides for viewing and printing the main TeX file. See below for a description of the commands in this menu.

S check Spelling

checks and correct the spelling of the include TeX file or the main TeX file if no include file is selected. This is done by the program AMSPELL, public domain software by A. Merckens. First a selection menu is presented, showing all languages supported. At this moment Dutch, US English, UK English, French, German, Italian, South-African, and Spanish are supported. Use the cursor keys and press **Enter** to select one.

If only the main file (no include file) is specified, \LaTeX will ask whether you would like to spell-check all files that the main file `\inputs` or `\includes`. If so, \LaTeX will invoke AMSPELL for each file it finds included in the main file. Input statements may be nested to almost any level.

See Section 7.3 for documentation of AMSPELL. If you only want to select the document language (e.g., for use of word spell-checking while editing a file) you can use the **Ctrl L** within the main menu and select a language.

U run Utilities

displays a selection menu containing the utilities implemented in \LaTeX . Use the cursor keys and **Enter** \leftarrow to select. Many of these utilities will display a submenu. The commands available in these submenus are discussed in Sections 4.3–4.10.

D execute Dos command

executes a single DOS command. The user is prompted for the command. After execution \LaTeX will wait for you to press either **D** to enter another DOS command, or any other key to return to the main menu.

To execute more than one DOS command, use **F9** to shell to DOS. At start up \LaTeX saves all DOS aliases (and also the environment and DOS history) to disk and removes them from memory. They are temporarily reloaded on shelling to DOS. To return to \LaTeX , enter

▷ exit

Shelling to DOS is possible in all \LaTeX menus and in the supported editors.

Within \LaTeX it is also possible to add, edit and run user specific utilities. This is done by pressing **F5**. This option is available in all \LaTeX menus and in the TSE editors.

B Backup TeX files

backups all relevant files (i.e., new and updated files) in the `texfiles` home directory and its subdirectories. The `texfiles` home directory is set in `texuser.<os>` by the environment variable `TEXFILES`. Backups are made to the directory set in the environment variable `BACKUP_PATH` and displayed in the field `backup path : a:\`, which is also set in `texuser.<os>`. Unless stated otherwise in `texuser.<os>`, a backup is made automatically on quitting \LaTeX . Backups can be made using the MS-DOS program `XCOPY` (default) or archivers like `ARJ` or `PKZIP`.

Q Quit

quits the \LaTeX workbench. On exit a backup is made (see above), several scratch files created by \LaTeX are deleted. Also, an options file (e.g., `book.opt`) is written. Among other things, this file contains information about the language used in the document and which \TeX format was used.¹ This information is read if the file `book.tex` is again loaded as `Main TeX file`. In this way, the user does not have to supply this information over and over again each time a file is loaded. Moreover, it safeguards the user from time-wasting experiences such as accidental compilation of a `SLIT \TeX` input file by `L \TeX` . The file `4texpars.<os>` is written at this time (see Section 3.6).

To prevent that you pressed **Q** or **Esc** unintentionally, you will be asked for confirmation before you really quit. You can set the `ASKQUIT` variable in `texuser.<os>` to skip the confirmation.

If for some reason \LaTeX is disrupted in some other way (e.g., due to wrong settings in the files `texuser.<os>` and `system.set`), the scratch files will remain on disk. These files are: `_4tex.`, `keepenv.set`, `histfile`, `aliasfil` and `4texpars.<os>` To remove them, you only need to restart \LaTeX with any parameter (e.g., `/r`) and quit again. Naturally you must correct the file `system.set` and/or `texuser.<os>` first.

4.1.1 Shortcuts and extras

\LaTeX supports the following extra key combinations in the main menu:

1. If you're wondering how \LaTeX knows all these things, here's a simple explanation. Default values are set in the file `texuser.<os>`. A value can change during a \LaTeX session for example by using the spell-checker, when you will be asked to choose a language.

Ctrl V

View the output of the current main \TeX file. This is a shortcut to going to the output menu, pressing **V** (view output) and returning to the main menu.

Ctrl R

Compile the main \TeX file and view it. A shortcut for 'run'.

Ctrl B

By using this key combination you can toggle \TeX 's *batch* mode. By default batch mode is off, meaning that \TeX will halt at any error that it encounters while compiling a text. In batch mode it will continue as best as it can, not showing any error messages on screen. In case you use TSE as your editor you can jump from one error to the next using the key combination **Ctrl N E** (next error) while editing (see also Section 5.1).

Ctrl D

The $\text{EM}\TeX$ dvi viewer comes in two versions: a 'normal' one (DVISCR) and a 'small' one (DVISCRS). The latter needs less memory but is a bit slower. In case you want to view a dvi file that calls many fonts the 'normal' viewer may run out of memory and abort. By toggling to the 'small' version you may still be able to view the output.

Ctrl E

The $\text{EM}\TeX$ compilers support several parameters for runtime memory settings. $\mathcal{A}\TeX$ sets them to appropriate values whenever you choose a format file. However, in some circumstances you may need to adjust them. Parameter stack, font memory, main memory, semantic nest size, pool size, save size and pattern memory can be dynamically changed. $\mathcal{A}\TeX$ will display defaults and ranges of these parameters, and enables you to change their current values.

Ctrl L

Select a language for spell-checking. A selection menu is presented, showing all languages supported. At this moment Dutch, US English, UK English, French, German, Italian, South-African, and Spanish are supported. Use the cursor keys and press **Enter ↵** to select one.

Ctrl G

Toggle between the MS-DOS version and the WINDOWS 95/NT version of Ghostscript. This option is introduced since the MS-DOS version is slightly quicker than the WINDOWS 95/NT version.

Ctrl T

Reindex the directory trees. $\mathcal{A}\TeX$ uses the 4DOS directory navigation. For this purpose a index files has to be generated and from to time this file has to be reindexed, so 4DOS can find al the newly created directories. The index is saved in the file `jpstree.idx`.

Ctrl F

Toggle devirtualizing fonts. If you get memory errors when running the $\text{EM}\TeX$ viewers, or when you view the `.dvi` file with DVIWIN you need to devirtualize the `.dvi` file. The `.dvi` file is devirtualized by the program DVICOPY. This takes some extra time, hence the toggle. See also Section 4.2.

PgUp or **PgDn**

Using the **PgUp** or **PgDn** key a history list of the last 10 selections of the main/include file are presented. You can use the cursor keys to select the main/include file you want, or press the **Esc** to abort history selection.

```

printer type      : Xerox Docutech PostScript (600dpi)
print destination : LPT1
viewer type      : HP Laserjet II, III, 4L (300dpi)
paper size       : A4 (210mm x 297mm)
dviscr type      : normal
devirtualize     : yes

output menu:
-----
Print book.dvi           Move book.dvi to a:\
View book.dvi           move Binary file book.ps to a:\
change printer Type     change Options
change print Destination Advanced options
change viewer tType     edit Fontmap files
show Logfile of dvi driver toggle dviscr Size
Return to main menu     toggle divirtualize

make your choice...

```

Figure 4.2: Output menu

4.2 The output menu

Upon pressing in the main menu, the Output menu appears (see Figure 4.2). As the name suggests, in this menu several options are presented which are all related to output to the screen or to a printing device. \LaTeX allows you to view or print on many output devices of different resolutions.

We will assume that `book` is the current main TeX file. Also, we assume that this file has been compiled successfully so that the file `book.dvi` exists.

The first line on the screen shows the current choice of printer type. If you did not change it during this \LaTeX session, it is the default printer type that can be set in the file `texuser.<os>`. It can be changed by pressing , see below.

The second line shows the print destination. If not changed during this session, it indicates the default print destination set in the file `texuser.<os>`. It can be changed by pressing , see below.

The third line shows the current choice of viewer type. If you did not change it during this \LaTeX session, it is the default viewer type that can be set in the file `texuser.<os>`. It can be changed by pressing , see below. The viewer type can be very different from the printer type, both in technology (e.g., PostScript or PCL) and resolution (e.g., 600dpi or 204dpi).

The fourth line shows the currently chosen paper size. It may also display that you chose to print multiple pages per sheet (2-up, 4-up, etc.)

The fifth line indicates which $\text{EM}\text{\TeX}$ viewer you use. The $\text{EM}\text{\TeX}$ dvi viewer comes in two versions: a ‘normal’ one (DVI`SCR`) and a ‘small’ one (DVI`SCR``S`). The latter needs less memory but is a bit slower. In case you want to view a dvi file that calls many fonts the ‘normal’ viewer may run out of memory and abort.

The sixth line indicates whether or not the .dvi file has to be devirtualized. If you get memory errors when running the EMT_EX viewers, or when you view the .dvi file with DVIWIN you need to devirtualize the .dvi file. The .dvi file is devirtualized by the program DVICOPY. This takes some extra time.

When using a PostScript printer \mathcal{A} T_EX uses partial font downloading. Partial font downloading is only possible for PostScript Type1 fonts. It helps to keep your PostScript output as small as possible by downloading only those characters of a font that are needed in a particular print. At low resolutions (say, up to 500 dpi) bitmapped fonts (.pk files) may be more efficient, at high resolution Type1 fonts are more efficient.

By default the Type1 versions of Computer Modern fonts are used for printing on PostScript printers, so psfonts.map will list, e.g.:

```
cmr10 CMR10 <cmr10.pfb
```

If you want to use bitmap fonts you must either delete that line entirely or comment it out by prepending an asterisk (*):

```
* cmr10 cmr10 <cmr10.pfb
```

You can edit the font map files psfonts.map, psfonts.inf and fontmap by pressing (see below).

The following keys are active in the output menu:

Print book.dvi

prints the file book.dvi to the current print destination, using the current print options and assuming that the actual printing will be done on a printer of the current printer type.

View book.dvi

views the file book.dvi on the screen, using the current view options. See Section 6.1.6 for further details on the viewing program.

change printer Type

presents a selection menu showing all printer types that are installed on your system. Select by using the cursor keys and press . For a list of currently available printer types see Section 6.6. Set the printer type to match the printer that will be used to print file book.dvi. See also part 6.

change print Destination

presents a selection menu showing all the print destinations that are installed on your system. Select by using the cursor keys and press . Possible choices are

- output to printer port LPT#
where # is 1, 2 or 3. Select the printer port of your computer to which the printer is connected. A print command will result in the file book.dvi being printed on your own printer.
- reroute output to serial port COM#
where # is 1, 2, 3 or 4. Select the serial port of your computer to which the printer is connected. A print command will result in the file book.dvi being printed on your own printer.
- output to own binary file
If selected, a print command results in writing the file book.x, where the extension .x depends on the current printer type. For example, if the current printer type is HP DeskJet, the binary file will be named book.dj. How to print this file is explained above (). *Note:* Binary files are difficult to read by humans. A PostScript binary file, however, consists of standard ASCII and therefore can be read in any editor.

- `reroute output to file TEXPRINT.PRN on drive G:`
Similar to above, except that the binary file will always be named `texprint.prn`. Also, if the file already exists the newly created output is appended to the file. Thus it is possible to combine several documents in one printable file. This option only applies if you run \TeX on a Novell network.
- `reroute output to network printer #`
where `#` is the name of a Novell network printer. If selected, a print command will result in the file `book.dvi` being printed on the network printer `#`. The correct printer type is automatically selected. After the file is printed, the user is notified by a message on the screen.

`change viewer tType`

presents a selection menu showing all viewer types that are installed on your system. Select by using the cursor keys and press . For a list of currently available viewer types see Section 6.6.

`show Logfile of dvi driver`

shows the logfile(s) created by `DVIHPLJ` and/or `DVISCR`, `DVIDOT`, `DVIPS` and/or `GHOSTSCRIPT`. These files contain all remarks and error messages shown on the screen and a bit more (statistics on memory usage etc.). The logfile is displayed using the internal `4DOS` command `list`. Press to quit. Under `WINDOWS 95/NT` the program `QUICKEDIT` is used instead.

`Move book.dvi to a:\`

moves the file `book.dvi` to a floppy in drive A: (or B:, set in `texuser.<os>`). This is useful if no (high-quality) printer is connected to your computer. The file can now be taken to another `MSDOS` computer on which \TeX is installed and printed there.

Note: this works only if all fonts that are used in the document `book.dvi` are present on the host computer. This may for example not be the case if you incorporated graphics generated by `BM2FONT` (see Section 7.4.2).

`move Binary file book.lj to a:\`

assuming that `book.lj` exists, moves it to a floppy in drive A: (or B:, the environment variable `MOVE_PATH` set in `texuser.<os>`). This is useful if no (high-quality) printer is connected to your computer. The file can now be taken to *any* other `MS-DOS` computer and printed there, simply by entering the command

▷ `copy /B book.lj PRN:`

(Do not forget the `/B` parameter!)

`change Options`

changes the print/view options. The default options are set in `texuser.<os>`. On pressing a fill-in screen appears, containing six frequently changed options. For each option the current setting is displayed in parentheses (see Figure 4.3). By pressing while the cursor is on the line of an option, this setting is accepted. To change it, type a new value and press . At any moment printing or viewing can be started using the current options by pressing or respectively. Press to abort changing and recover the old options.

`Advanced options`

changes the print/view options. On pressing a screen with the Change Options menu appears, showing many not so frequently used or changed options (see figures 4.4 and 4.5). By pressing in this menu, the user can set all parameters described in the documentation of `DVIHPLJ`, `DVIPS`, `DVIDOT`, `DVISCR`, and `DVISCRS`. To do so, type the relevant parameter values as described in the manuals and press . To restore the old parameter values press . If the printer type differs from the viewer type, \TeX will ask you whether you want to change advanced options for the printer or the viewer. By pressing or the file `book.dvi` will be printed or viewed respectively. Press to return to the output menu.

options:

```

begin page           (1)  1
end page             (last) last
begin with last page (N)  N
two side printing    (N)  N
number of copies     (1)  1
paper size           (A4)
  c: change
  m: multiple pages on one page

```

```

[Ctrl] P   start printing any time
[Ctrl] V   start viewing any time
[Esc]      abort options

```

Figure 4.3: Output options menu

[F] edit Fontmap files

loads the files `psfonts.map`, `psfonts.inf` and `fontmap` into the editor, so you can adjust them to your needs.

[R] Return to main menu

returns to the main 4TeX menu. Equivalent keys are **[Q]** and **[Esc]**.

Note: In case you use the CD-rom and you have a printer/viewer that is not present in the selection menu of all printer/viewer types, it is possible to add a new type (or change an existing one). This is done by setting the environment variable `OWNPRT` in the file `texuser.<os>`. The variable `OWNPRT` will specify where 4TeX can find additional printer and viewer types. Examples of files that define a new printer/viewer type (with the extension `.prt` and `.vie`) can be found in the directory `δ:\emtex\prndest`. When the variable `OWNPRT` is not empty and you want to change the printer/viewer type 4TeX will ask which directory to search for types (i.e. the CD-rom or the `OWNPRT` directory).

4.3 The BibTeX menu

This menu appears after choosing `Bibtex` in the `run Utilities` selection menu of the main 4TeX menu. This menu presents the options that have to do with maintaining the `.bib` database files and running BIBTeX (see Figure 4.6). See Section 7.1.1 for more information on BIBTeX.

In `texuser.<os>` the user can indicate the default `.bib` file by specifying the environment variable `BIBNAME`. We assume here that this is `bibfile.bib`.

[F] change bib File

selects one or more `.bib` files. Press **[Enter ↵]** on the empty field to get a list of all available `.bib` files

```

-b # generate # copies of every page    -m manual feed for printer
-p # first page to print is #           -q run in quiet mode
-l # last page to print is #            -t # papertype: # is a3,a4,landscape
-n # print at most # pages              -x # magnification ration is #/1000
-r stack pages in reverse order         -a conserve memory, not time
-A print only (TeX) odd pages           -E create EPSF file
-B print only (TeX) even pages          -V download PK fonts, not PS fonts
-j0 disable partial font downloading    -z Hyperdvi to HyperPostScript

```

```
print options: -p1
```

```
print options menu:
```

```

change Options
Print book.dvi
View book.dvi
Return to output menu

```

```
make your choice...
```

```
(c) 4U 1991-1997
```

Figure 4.4: Advanced output options for DVIPS

in the current directory, indicated by `bib file path`. Alternatively, type the name of the `.bib` file you want to work with or type `@` to select *all* `.bib` files in the `bib files path` (see below).

`D` edit bib Database

edits the selected `bib file` with the aid of BIBDB, a public domain database program by E. Doron. See Section 7.1.2 for a detailed description of BIBDB.

`E` Edit bibfile.bib

loads `bibfile.bib` into the editor. The file can now be edited as a plain ASCII file. See L. Lamport (1986) for a detailed description of how to edit an entry in a `.bib` file.

`B` run Bibtex

runs the program `BIBTEX` to automatically make a list of references occurring in `book.tex` and the include files (such as `chap1.tex`). See Section 7.1.1 for more details.

`L` show Logfile

shows the logfile created by `BIBTEX`. This file contains all remarks and error messages produced during compilation, and shown on the screen. It is displayed using the internal 4DOS command list. Press `Esc` to quit.

`I` generate TeX list of references

generates the file `reflist.tex` that contains *all* references in one or more `.bib` database files. `reflist.tex` is automatically generated and compiled, then `BIBTEX` is run and finally the file is compiled for a second time. It can now be viewed and/or printed in the usual way. This option makes it possible to have a printout of all your references. This can be handy when typing text and looking for the Bibtex keys.

`P` change bib files Path

clears the field `bib files path` and another subdirectory can be chosen. The default for this path can be set by the `BIBINPUT` parameter in `texuser.<os>`. If you enter a directory name with wildcards (or

```

/?  show options           /al# left margin e.g. /al:1in
/?? show all options      /l#  move left margin
/m#  magnification e.g. /m:1200
/at# top margin           /n#  # pages to be printed
/tr# transformation      /t#  move top margin
/w#  page width           /h#  page height e.g. /h:12in
/k#  # pages to be skipped before printing
/v5  show all statistics

```

```
options: /b1
```

```
print options menu:
```

```

change Options
Print book.dvi
View  book.dvi
Return to output menu

```

```
make your choice...
```

Figure 4.5: Advanced output options for DVIHPLJ, DVIDOT and DVISCR

an empty one) $\text{\textbackslash}4\text{T}\text{E}\text{X}$ will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Use the $\text{\textbackslash}\text{Esc}$ to restore the directory name to the old setting.

$\text{\textbackslash}\text{T}$ Toggle main TeX file

toggles between the ‘main TEX file’ and the ‘include TEX file’ from the main menu. In case you use a $\text{L}\text{A}\text{T}\text{E}\text{X}$ style such as `chapterbib` you will want to run $\text{BIB}\text{T}\text{E}\text{X}$ on each chapter of your document.

$\text{\textbackslash}\text{R}$ Return to main menu

returns to the main $\text{\textbackslash}4\text{T}\text{E}\text{X}$ menu. Equivalent keys are $\text{\textbackslash}\text{Q}$ and $\text{\textbackslash}\text{Esc}$.

4.4 The graphics menu

This menu appears after choosing `Graphics` in the `run Utilities` selection menu of the main $\text{\textbackslash}4\text{T}\text{E}\text{X}$ menu (see Figure 4.7). See Section 7.4 for more information on graphics in TEX .

$\text{\textbackslash}\text{F}$ choose picture File

selects a picture file containing a picture that has to be incorporated in your TEX document. $\text{\textbackslash}4\text{T}\text{E}\text{X}$ recognizes files with the following extensions as picture files:

```

Bitmap files: .gif, .tif, .pcx, .bmp, .iff, .lbm, .img, .cut, .jpg
HPGL plotter files: .hpg, .plt, .hpp
POSTSCRIPT files: .eps, .ps
HP PCL bitmap files: .pcl

```

Depending on the type of the file that is selected, an appropriate viewer and conversion program is automatically chosen by $\text{\textbackslash}4\text{T}\text{E}\text{X}$.

```

main TeX file   : book.tex
bib file        : bibfile.bib
bib files path  : c:\cdtex\texfiles\bib
bib style path  : c:\emtex\latex209\bibinput!;C:\cdtex\texfiles\TEXINPUT!;
                  d:\emtex\LATEXALL;d:\emtex\INPUTS!;d:\emtex\LATEX209!

bibtex menu:
-----
change bib File
edit bib Database
Edit bib file
run Bibtex
show Logfile
generate TeX list of references
check Syntax of bib file
change bib files Path
Toggle main TeX file
Return to main menu

make your choice...

```

Figure 4.6: BibTeX menu

Note: If you have a (Enhanced) Windows Metafile (.wmf, .emf) you can use a program like PaintShop Pro or the Irfan viewer to convert the Windows Metafile to one of the graphic formats as described above.

View picture

views the selected picture using the current options (see below). Under MS-DOS press to exit the viewer.

Convert picture file

converts the selected picture file to a format which can be imported into your TeX document, using the current options. Depending on the format of the selected picture file 4TeX selects a conversion program. How to include the picture in your document is explained on the screen (see also Section 7.4).

change picture files Path

clears the field picture files path and another subdirectory can be chosen. If you enter a directory name with wildcards (or an empty one) 4TeX will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Press to restore the directory name to the old setting.

change Options

changes the options for the viewer and conversion program. Defaults can be set in texuser.<os>. On pressing this key a screen appears showing all options for the appropriate viewer and conversion program (see figures 4.8-4.10). The user can set all parameters described in the documentation of the viewer and conversion program. To do so, type the relevant parameter values as described in the manuals and press . To restore the old parameter values press .

change printer Type

presents a selection menu showing all printer types that are installed on your system. Select by using the


```

picture file      : golfer.ps
picture files path : c:\graphics\plaatjes
TeX files path    : c:\texfiles\4texdoc
options           : -m100 -pmono
converting        : (Encapsulated) Postscript picture to PCX picture
printer type      : HP Laserjet II, III, 4L (300dpi)
viewer type       : HP Laserjet II, III, 4L (300dpi)

```

graphics menu:

```

-----
choose picture File
View picture
Convert picture file
change picture files Path
change Options
change printer Type
change viewer tYpe
Return to main menu
(re)calculate BoundingBox

```

make your choice...

(c) 4U 1991-1997

Figure 4.7: Graphics menu

cursor keys and press . For a list of currently available printer types see Section 6.6. Set the printer type to match the printer that will be used to generate the high-quality pictures (e.g., PostScript pictures).

change viewer tYpe

presents a selection menu showing all viewer types that are installed on your system. Select by using the cursor keys and press . For a list of currently available viewer types see Section 6.6. Set the viewer type to match the viewer that will be used to generate the low-quality pictures (e.g., PCX bitmap pictures).

Note that the graphics utility allows you to generate low resolution pictures that can be used for viewing and high resolution pictures that can be used for printing. For instance if you have a 600dpi HP LaserJet IV PostScript printer you probably want to print at 600dpi but want to view at 300dpi (because 300dpi fonts need less memory and load more quickly). This can be done by selecting the correct printer and viewer type. The selection of the viewer/printer type will also change the printer/viewer type for the output menu!

Return to main menu

returns to the main 4TeX menu. Equivalent keys are and .

If the current picture file is a POSTSCRIPT file then there is the extra option:

(re)calculate BoundingBox

A BoundingBox specifies the size and origin of an .eps or .ps picture. Some programs do not calculate the BoundingBox correctly or even do not calculate it at all. Use this option to make sure that the BoundingBox is correct. See Sections 7.4.4 and 7.4.5 for more information on POSTSCRIPT pictures and BoundingBoxes.

Options for converting Bitmap pictures to TeX fonts (BM2FONT)

-h<horizontal resolution>	dots/inch, default 300 dpi
-v<vertical resolution>	dots/inch, default 300 dpi
-a<show pictures on screen>	y/n, default n
-e<stretch EGA pictures>	y/n, default y
-i<inversion of pixels>	y/n, default n
-g<greypixels in bitmap>	y/n, default n
-p<write pixel files>	y/n, default n
-w<let white be light grey>	y/n, default y
-d<distribute errors>	y/n, default y
-s<separation of grey dots>	y/n, default n
-r<repeat each grey pixel>	y/n, default n
-u<pixels for grey rectangle>	less 8
-c<vert. pixels for rectangle>	less 8
-x<bits per sample>	0 < x < 9
-b<reduce halftone colors>	f.e. by 1, less u*c*4,default 0
-t<gradation value>	in %, default 70
-z<area of gradation>	in %, default 70
-m<width of picture on paper>	in mm
-n<height of picture on paper>	in mm

options: -wn -b3 -t0 -z0 -m100

(c) 4U 1991-1997

Figure 4.8: Options for bitmap graphics in the graphics menu

If the current picture file is a Bitmap file then there is the extra option:

G run Graphics program *xxx*

where *xxx* is GRAPHIC WORKSHOP (in case you are running MS-DOS) or PAINTSHOP PRO (in case you are running WINDOWS 95/NT). Both (shareware) programs are fine tools for working with bitmapped graphic files. They will handle most of the popular bitmap formats. GRAPHIC WORKSHOP supports the following operations on graphic files: View, Convert between any two formats, Print, Dither color pictures to black and white, Reverse, Rotate, Flip, Scale, Reduce the number of colors, Sharpen, Soften, Adjust the brightness, contrast and color balance of the bitmaps. PAINTSHOP PRO is even more powerful.

4.5 The MakeIndex menu

This menu appears after choosing MakeIndex from the Utilities menu of the main \LaTeX menu (see Figure 4.11). See Section 7.2 for more information on how to make an index using the program MAKEINDEX and how to generate multiple indices.

We assume again that the main TeX file is `book.tex`. Moreover, we assume that all `\index` commands are placed in the file `book.tex` or files included by `book.tex`.

E Edit index file

loads the file `book.ind` that was created by a MAKEINDEX (see below) into the editor. Corrections can now be made manually. Remember that these alterations are discarded each time MAKEINDEX is run.

M run Makeindex

runs MAKEINDEX to create the index file `book.ind`. If this file is included in the document `book.tex` the index will be created on re-compilation of `book.tex`.

Options for converting PostScript to PCX pictures (Ghostscript)

-h<horizontal resolution>	dots/inch, default 300 dpi
-v<vertical resolution>	dots/inch, default 300 dpi
-m<the horizontal size in mm>	default BoundingBox size
-n<the vertical size in mm>	you only need to specify -m or -n
-r<rotation angle>	rotation in degrees
-p<pcx type>	mono a mono pcxfile (default)
	16 a 16 color pcxfile
	256 a 256 color pcxfile

Current picture size:

-m=100 (max. 205 mm)
-n=131 (max. 294 mm)

options: -m100 -pmono

(c) 4U 1991-1997

Figure 4.9: Options for PostScript graphics in the graphics menu

L show Logfile

shows the logfile created by MAKEINDEX. This file contains all remarks and error messages shown on the screen. Under MS-DOS it is displayed using the internal 4DOS command list. Press **Esc** to quit. Under WINDOWS 95/NT it is displayed using QUICKEDIT.

G Generate word list

generates the ASCII file book.wrd that contains an alphabetically sorted list of all words in the file book.tex. All `\input` and `\include` statements will be traced. This file can be of help in deciding what should be in the index and how it should be organized. It is also used for automated indexing with the TSE editor (see Chapter 5.1).

W edit Word list

loads the file book.wrd that was generated by `Generate word list` into the editor to view, edit or print.

X change file eXtensions

By default MAKEINDEX will compile book.ind from the input file book.idx. However, if you defined multiple indexes (e.g., a 'Subject index' and an 'Author index') by means of the \LaTeX style file index.sty, you will want to compile e.g., book.sdx and book.adx. For that reason you can choose from file extensions .idx or .*dx. In the latter case all files matching book.*dx will be compiled to book.*nd (see Section 7.2).

O change makeindex Options

changes options for the MAKEINDEX program (see Figure 4.12). MAKEINDEX support several styles of index generating, e.g., the ordering and the way page ranges are handled can be changed.

R Return to main menu

returns to the main \LaTeX menu. Equivalent keys are **Q** and **Esc**.

Options for converting HPGL to PCX and EPS pictures (HP2xx)

```
-t<conversion type>      cad,em,epic,eps,img,mf,pbm,pcl,pcx
-h<horizontal resolution> dots/inch, default 300 dpi
-v<vertical resolution>  dots/inch, default 300 dpi
-m<the horizontal size in mm> default 100 mm
-n<the vertical size in mm> you only need to specify -m or -n
-r<rotation angle>      rotation in degrees
-p<pen size>             in 1/10 mm (mf,ps) or dots (rest)
                        default 11111111
-c<pen color>           see manual for details
```

Current picture size:

```
-m=100      (max. 200 mm)
-n=82       (max. 200 mm)
```

options: -m100

(c) 4U 1991-1997

Figure 4.10: Options for HPGL graphics in the graphics menu

4.6 The conversion menu

Sometimes you may want to convert files produced by other word processors to (La)TeX texts. Or you may want to use the extended ASCII set for accented letters instead of the less readable equivalent TeX commands. This may also be necessary when you want to use the text in other word processors. A more drastic approach is to ‘deTeX’, i.e. strip all TeX commands from a file.

4TeX currently supports the following conversions:

- WORDPERFECT 5.1 to L^ATeX
- WORDPERFECT 4.2 to TeX
- WORDPERFECT 4.2 to L^ATeX
- MS-WORD to L^ATeX
- UNIX text file (LF) to MS-DOS text file (CR-LF)
- TROFF to L^ATeX
- TROFF to TeX
- TIB bibliography to BIBTeX
- TeX ASCII to TeX 8 bits
- TeX 8 bits to TeX ASCII
- RTF (Rich Text Format) to L^ATeX
- PC-WRITE to L^ATeX
- MIDI to MusicTeX
- MACINTOSH text file (CR) to MS-DOS text file (CR-LF)
- program listing to TeX
- (L^A)TeX to HTML
- HTML to L^ATeX
- HTML to human readable ASCII

```

index file      : book.idx
word list      : book.wrd
makeindex options :

```

```

makeindex menu:

```

```

-----
Edit index file
run Makeindex
change index file eXtension
change makeindex Options
show Logfile
Generate word list
edit Word list
Return to main menu

```

```

make your choice...

```

```

----- (c) 4U 1991-1997 -----

```

Figure 4.11: MakeIndex menu

- DISPLAYWRITE to ASCII
- DVI file to ASCII
- MS-DOS text file (CR-LF) to UNIX text file (LF)
- MS-DOS text file (CR-LF) to MACINTOSH text file (CR)
- (La)T_EX to plain ASCII using unT_EX
- (La)T_EX to plain ASCII using DeT_EX
- CHIWRITER to T_EX
- CHIWRITER 4.x to L^AT_EX
- Hard returns (CR) to soft returns (WP input)
- ASCII screen dump to L^AT_EX picture
- ABC to MusicT_EX
- SJIS text into preprocessed (L^A)T_EX
- PostScript to canonical EPS (curves by PS_CONV)
- SJIS text containing CEF macros into preprocessed (L^A)T_EX
- Text containing CEF macros into preprocessed (L^A)T_EX
- Big-5 text containing CEF macros into preprocessed (L^A)T_EX
- Big-5 text to preprocessed (L^A)T_EX

The conversion menu can be chosen from the utility menu (press U in the main menu, see Figure 4.13).

The following functions are available in the conversion menu:

change Input file

Enter the name of the file that you want to convert. You may enter a pattern using wildcards to select a file.

4TeX help

change makeindex Options

```
-c      Compress intermediate blanks (ignoring leading and trailing blanks
        and tabs). Default: retain blanks.
-g      Employ German word ordering in the index. See documentation!
-l      Letter ordering. Default: word ordering.
-p num  Set the starting page number of the output index file to be 'num'
        (useful when the index file is to be formatted separately). The
        argument num may be numerical or one of the following:
        any    The starting page is the last source page number plus 1.
        odd   Start at first odd page following last source page number.
        even  Start at first even page following last source page number.
-r      Disable implicit page ranges (e.g. '1, 2, 3, 4'). Default: '1-4'.
-s sty  Use 'sty' as style file. See documentation!
```

press any key to continue...

(c) 4U 1991-1997

Figure 4.12: Help screen for options for MAKEINDEX

change Output file

Enter the name of the file in which the output of the conversion will be stored. Obviously this cannot be the same as the input file. Wild cards are not allowed.

change files Path

changes the path to the file(s) you want to convert. If you enter a directory name with wildcards (or an empty one) 4TeX will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Press to restore the directory name to the old setting.

View input file

views the file that you will be converting.

Note: Binary files (such as WORDPERFECT files) can be hard or impossible to read for human beings.

vieW output file

views the file produced by the conversion program.

Note: Binary files (such as WORDPERFECT files) can be hard or impossible to read for human beings.

choose coNversion

chooses a conversion from the list of available conversions.

Convert input file

starts the conversion process. Note that 4TeX does *not* check if the input and/or output file are valid for the chosen conversion. It does warn you when you are about to overwrite an existing file.

Return to main menu

leaves the conversion menu and return to the main menu. Equivalent keys are and

```

input file : text.wp5
output file : text.tex
files path : C:\TEFILES\4TEXDOC
conversion : WordPerfect 5.1 to LaTeX

```

```

conversion menu:

```

```

change Input file
change Output file
change files Path
View input file
vieW output file
choose coNversion
Convert input file
Return to main menu

```

```

make your choice...

```

```

(c) 4U 1991-1997

```

Figure 4.13: Conversion menu

Note: the quality of the conversion programs varies greatly. To get the best results you may have to experiment, perhaps using two or three different conversions in a row.

4.7 The Fontlib menu

This menu appears after choosing `Fontlib` from the `Utilities` menu of the main `4TEX` menu (see Figure 4.14). Normally DVI-drivers use PK fonts to print/view your document. With `EMTEX` it is possible to store several PK bitmap fonts into one font library (`fli` file). A font library takes less space than the bitmap fonts and does not flood your directories with many fonts. This simplifies your directories setup (see also Section 3.5.5).

[N] change font library Name

specifies a new font library name. This file will be used to store all selected `TEX` fonts (`.pk`).

[P] change font library Path

clears the field `font library path` and another subdirectory can be chosen. Select a directory where the font library file can be found or the new font library file will be written. The fonts are found in the font path and all its subdirectories. The font path directory is determined by the printer selected in the output menu. If you enter a directory name with wildcards (or an empty one) `4TEX` will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Press **[Esc]** to restore the directory name to the old setting.

[A] add All fonts in font path and subdirectories

All `.pk` fonts in the font path directory and its subdirectories are stored in the font library. The stored fonts are deleted from the directories and any empty directories are removed.

```
font library name : ownfli.fli
font library path : C:\cdtex\texfiles\FONTS\laserjet
font path         : C:\cdtex\texfiles\FONTS\laserjet
```

```
font library menu:
```

```
change font library Name
change font library Path
add All fonts in font path and subdirectories
add Selected fonts from font path and subdirectories
Unpack library into subdirectories of font path
List fonts in font library
Test integrity of font library
Return to menu
```

```
make your choice...
```

```
(c) 4U 1991-1997
```

Figure 4.14: Fontlib menu

S add Selected fonts from font path and subdirectories
selects the fonts you want to store in the font library. The **+** key will select/mark a font to store in the library, **-** will unselect/unmark a font to store. Press the **Esc** key if you do not want to select any font. You can select a single file by moving the scroll bar to the filename and pressing the **Enter** key without marking any other files. The stored (.pk) fonts are deleted from the directories and empty directories are removed.

U Unpack library into subdirectories of font path
unpacks the selected font library. Subdirectories named after the (rounded) resolution (e.g. 518dpi) of fonts are automatically made in the font library path specified by the variable MYFONTS in the file texuser.<os>.

L List fonts in font library
makes a listing of all the fonts in the selected font library (see also Section 6.5).

T Test integrity of font library
tests the integrity of the font files in the selected font library, i.e. if the files stored in the font library are valid .pk files, that can be used for viewing/printing your documents.

R Return to menu
leaves the fontlib menu and return to the main menu. Equivalent keys are **Esc** and **Q**.

4.8 The format menu

This menu appears after choosing **Format** from the **Utilities** menu of the main 4TeX menu (see Figure 4.15). Format files are files with preloaded declarations, assignments, and macro definitions that can


```

format name      : all available format files
format description :
format input path : d:\emtex\formats
format output path : d:\emtex\btexfmts
languages       : US English
codepage        : none
emTeX options   : /mt:7300

```

format menu:

specify format Name	Toggle between Big and Huge format
edit format Description	change emTeX options
change format input Path	select codepage
select Languages for BABEL	generate Format
Return to main menu	show log file

make your choice...

(c) 4U 1991-1997

Figure 4.15: T_EX format menu

be used within your document. For a detailed discussion of format files and how they can be generated see Section 5.2.4

N specify format Name

specifies the name of the format you want to generate. Examples of available format files are:

plain the plain T_EX format file

latex2e the L^AT_EX 2_ε format file

lplain the L^AT_EX 2.09 format file

blue the BLUe format file

contxnl the ConT_EXt format file

Many more formats may be available, depending on the installation.

After typing **N** you can type the format you want to generate. You can use wildcards to select a certain format. By typing **@** you will select all available formats.

D edit format Description

A format file can be assigned a description. If left blank the format name will be used as the description. When using a *big* T_EX version the description will automatically start with the word ‘big’. The descriptions are used for the selection of the formats in the main menu.

P change format input Path

clears the field `format input path` and another subdirectory can be chosen. If you enter a directory name with wildcards (or an empty one) 4T_EX will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Press

Esc to restore the directory name to the old setting.

select Languages for BABEL

With T_EX 3.0 it has become possible to load hyphenation patterns for more than one language. To take advantage of this new feature some extra book keeping is needed. BABEL is a set of T_EX macros that provides a simple way to generate T_EX format files (see below) with multiple languages, and macros to switch languages within one T_EX document. A list of all available languages and hyphenation patterns is displayed and you can select the languages you want to use in the new format.

change emT_EX options

4T_EX automatically sets the options for the T_EX compilers. **Warning:** it is not recommended to change them, unless you have studied the documentation very carefully.

Toggle between Big and Huge format

You can either generate *big* or *huge* versions of formats. The huge version is used with HugeT_EX and has more main memory, higher memory settings, but may run slowly.

select codepage

selects the codepage translation table to be used when generating a new format file. See documentation of T_EX compilers for details about the usage of codepage tables. Selecting ‘none’ implies true 8 bits input that can be used with the DC fonts or special input encodings.

generate Format

available formats” you are prompted for every format whether you want to generate it or not.

show logfile

INIT_EX writes informative messages to a log file. Use this option to view the log file.

Return to main menu

leaves the conversion menu and return to the main menu. An equivalent key is Esc.

Note: In case you use the CD-rom you can add additional format files on a hard disk. This is in particular handy if you have an update or new version of a format that is already on the CD-rom. The (new) format files are stored in the directory specified by the environment variable OWNFMT that can be found in the file `texuser.<os>`. When a new format is generated the format file is stored in `%OWNFMT\btxfmfs` or `%OWNFMT\htexfmfs`. If the variable OWNFMT is not empty 4T_EX will ask from which directory you want to select a format in the main menu (i.e. from CD-rom or from the directory specified by OWNFMT).

4.9 The Metafont format menu

This menu appears after choosing MFformat from the Utilities menu of the main 4T_EX menu (see Figure 4.16).

Edit printer definitions

The file `δ:\emtex\metafont\mfinput\local.mf` contains the printer definitions used by METAFONT to generate the printer fonts. The file `δ:\emtex\metafont\mfjob\modes.mfj` contains several directory settings, path and environment definitions needed to run METAFONT for several predefined printers (see the 4T_EX printer list in Section 6.6). When you have changed a printer definition or added a new printer, you need to generate new METAFONT format files (e.g., for use with automatic font generation).

generate MetaFont Format

runs INI-METAFONT to generate the specified format(s).

show Log file

If METAFONT detects errors, they are written in a logfile. Use this option to view the logfile.

```

printer definitions : d:\emtex\METAFONT\MFINPUT\local.mf
metafont path      : d:\emtex\METAFONT
emTeX options      :

```

```
metafont format menu:
```

```

-----
Edit printer definitions
generate MetaFont Format
show Log file
change emTeX Options
Return to main menu

```

```
make your choice...
```

```
----- (c) 4U 1991-1997 -----
```

Figure 4.16: METAFONT format menu

change emTeX Options

4TeX automatically sets the options for the METAFONT compilers (see Figure 4.17). **Warning:** it is not recommended to change them, unless you have studied the documentation very carefully.

Return to main menu

leaves the METAFONT menu and return to the main menu. An equivalent key is Esc.

4.10 The MFjob menu

This menu appears after choosing MFjob from the Utilities menu of the main 4TeX menu (see Figure 4.18).

change metafont job File

selects the METAFONT job file (.mfj) to be edited/created. In a METAFONT job file you can specify which fonts should be generated.

change metafont job Path

clears the field job path and another subdirectory can be chosen. If you enter a directory name with wildcards (or an empty one) 4TeX will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Press Esc to restore the directory name to the old setting.

change metafont source pAth

selects a directory containing METAFONT sources (i.e. .mf files containing the font description). There are two directories to choose from, i.e. the directory δ : \emtex\metafont\mfinput (specified by the

/8 Characters with codes 128-254 are accepted in input files.
/a* This option is used to set the name of the file used for calling an editor. Between /a and the file name you can insert `:',`=',' :=', or nothing. Spaces are not allowed. You have to use `\' as path delimiter unless your operating system is setup to use `-' for options. See below for more information.
/b Batch mode.
/e Error stop mode.
/g* Select graphics adapter (\gc, \ge or \gv for cga ega and vga)
/mp# Set string memory size: 20000 to 65500, default 35000.
/n Nonstop mode.
/o Allow 8 bit characters (128-254) for output. Only observed by INIMF or when using a base file created with /o in effect. Output of `ä' without /o: ^84, with /o: ä.
/pv* Set drive name for default directories. Between /pv and the drive name (a-z, A-Z) you can insert `:',`=',' :=', or nothing.
/s Scroll mode.

options:

==== (c) 4U 1991-1997 =====

Figure 4.17: Options for the METAFONT format menu

environment variable MFINPUT in the file system.set) and the directory c:\texfiles\mf (specified by the environment variable MYMF in the file texuser.<os>).

S Select source files to make metafont job file

After selecting one ore more .mf files, a METAFONT job file will be created or extended.

E Edit metafont job file

The METAFONT job file contains all statements needed to generate fonts. You can edit this file.

O specify metafont job Options

changes options for the MetaFont program. An overview of all options is displayed on screen (see Figure 4.19). **Warning:** it is not recommended to change them, unless you have studied the documentation very carefully.

M run Metafont job

METAFONT will generate fonts and store them in the right directories.

L show Logfile

shows the remarks and error messages that METAFONT produced. They are stored in the log file.

R Return to main menu

leaves the MFjob menu and return to the main menu. An equivalent key is Esc.

4.11 The Metafont menu

This menu appears after choosing Metafont from the Utilities menu of the main 4TEX menu (see Figure 4.20).

```

job file      : book.mfj
job path     : C:\cdtex\texfiles\MF
sources path : C:\cdtex\texfiles\MF
output path  : C:\cdtex\texfiles\FONTS
mfjob option :

```

MFjob menu:

```

change metafont job File
change metafont job Path
change metafont sources pAth
Select source files to make metafont job file
Edit metafont job file
specify metafont job Options
run Metafont job
show Logfile
Return to main menu

make your choice...

```

(c) 4U 1991-1997

Figure 4.18: MFjob menu

change input File

selects the METAFONT file (.mf) to be edited/created.

change input Path

clears the field input path and another subdirectory can be chosen. If you enter a directory name with wildcards (or an empty one) 4TeX will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you do not have to type the whole directory name (correctly) but only a few letters are often enough. Press to restore the directory name to the old setting.

change Options

changes options for the METAFONT program.

Edit input file

The METAFONT file contains all statements needed to generate screen output and/or output to .gf file. You can edit this file. The .gf file contains the bitmapped METAFONT output. It can be converted to a TeX font file by the program GFTOPK.

run Metafont

starts the METAFONT program, using the options supplied and the chosen METAFONT file as input. All messages that METAFONT generates in this run are also written to the log file.

show Logfile

shows the remarks and error messages that METAFONT produced. They are stored in the log file.

eXtract EPS from logfile

In case you use the MFtoEPS package written by B. Jackowski, P. Pianowski and M. Ryćko, you may want to extract the Encapsulated PostScript code that was written in the log file. Pressing will extract all .eps files from the log file.

MFJOB parameters

```

/1          use MetaFont for 80186/80286 processors
/3          use MetaFont for 80386 and higher processors
/b          use big MetaFont
/c<filename> create batch file, do not create fonts
/a          create all fonts
            (default: create only those which do not exist)
/d          do not use expanded memory
/i          ignore errors produced by MetaFont
/n          do not make fonts, just list commands
/t<directory> set temporary directory
/m          MFJOB runs in master mode (for network)
/s          MFJOB runs in slave mode (for network)
/w          use Wayne Sullivan's Metafont

```

options:

Figure 4.19: Options for the MFjob menu

View EPS file

In case you use the MFTOEPS package written by B. Jackowski P. Pianowski and M. Ryćko, and you have extracted the .eps files you may want to view them on screen to check the results.

Return to main menu

leaves the METAFONT menu and return to the main menu. An equivalent key is .

4.12 The Metapost menu

The METAPOST system implements a picture-drawing language very much like Knuth's METAFONT except that it outputs PostScript commands instead of bitmaps. METAPOST is a powerful language for producing figures for documents to be printed on PostScript printers. It provides easy access to all the features of PostScript and it includes facilities for integrating text and graphics.

To use METAPOST, you prepare an input file containing METAPOST code and then run METAPOST. For an input file called `foo.mp` METAPOST will produce output files with names like `foo.1` and `foo.2`. These files contain the output in EPS (Encapsulated PostScript) format. Any messages sent to the screen are also saved in a log file.

change input File

selects the METAPOST file (.mp) to be edited/created.

change input Path

clears the field MP input path and another subdirectory can be chosen. If you enter a directory name with wildcards (or an empty one) 4TeX will display a list of all directories that satisfy the wildcard specification. By using the cursor keys you can select the directory you want. Using this approach you

```
MF input file : book.mfj
MF input path : C:\cdtex\texfiles\MF
MF options    :
```

```
metafont menu:
```

```
change input File
change input Path
change Options
change Cpu
Edit input file
run Metafont
show Logfile
eXtract EPS from logfile
View EPS file
Return to main menu
```

```
make your choice...
```

```
(c) 4U 1991-1997
```

Figure 4.20: The METAFONT menu

do not have to type the whole directory name (correctly) but only a few letters are often enough. Press to restore the directory name to the old setting.

Edit input file

The METAPOST file contains all statements needed to generate PostScript output. You can edit this file.

run Metapost

starts the METAPOST program, using the chosen METAPOST file as input. All messages that METAPOST generates in this run are also written to the log file. chosen METAFONT file as input. All messages that METAFONT generates in this run are written to the log file.

show Logfile

shows the remarks and error messages that METAPOST produced. They are stored in the log file.

View Metapost output file(s)

will run a $\text{T}_{\text{E}}\text{X}$ job² that includes all METAPOST output files, convert the resulting DVI file to PS, and display it using GHOSTSCRIPT.

Convert Metapost output file(s) to full EPS

converts the METAPOST output to ready-made EPS files. This extra step is necessary if the METAPOST input file called for $\text{T}_{\text{E}}\text{X}$ to typeset pieces of text.

view EPS file

METAPOST produces PostScript output files. You can view these on screen by using GHOSTSCRIPT. Select the output file from the menu and then press to view it. Note that choosing a non-PostScript file may generate strange errors. Any PostScript files usually starts with %!PS.

2. Actually, the $\text{T}_{\text{E}}\text{X}$ file `mproof.tex` is compiled. It can be found in `\delta:\emtex\inputs\misc`.

```
MP input file : examples.mp
MP input path : d:\emtex\metapost\mpinputs
```

```
metapost menu:
```

```
change input File
change input Path
Edit input file
run Metapost
show Logfile
View Metafont output file(s)
Convert Metapost output to full EPS
vieW EPS file
Return to main menu
```

```
make your choice...
```

```
(c) 4U 1991-1997
```

Figure 4.21: The Metapost menu

R Return to main menu

leaves the METAPOST menu and return to the main menu. An equivalent key is Esc.

4.13 The Install PostScript font menu

This menu appears after choosing `InstPsFont` in the `run Utilities` selection menu of the main 4TeX menu. This utility makes it easy to use whatever PostScript font you need in your (L)TeX documents. If you have a complete font family of PostScript fonts you can use the utility `InstPsFam` to develop your own style file and use this as your default font family (instead of the Computer Modern family).

Before you can use a PostScript font within TeX documents you need several utilities to convert the PostScript font into all kinds of other files (`.vf`, `.tfm` and `.pk`). A PostScript font consists of at least two files: a `.afm` file and a `.pfb` or a `.pfa` file. The `.afm` (Adobe font metric) can be regarded as the PostScript equivalent of the `.tfm` file within TeX. I.e. it contains the information about the sizes of the characters. The `.pfb` or `.pfa` file can be seen as the PostScript versions of a `.mf` file METAFONT uses. The difference between a `.pfb` and a `.pfa` file is that the first one is an 8 bits file and the second one is an ASCII file.

If you do not have an `.afm` but only a `.pfm` (i.e. this file is used within MS-Windows) you can convert the `.pfm` to a `.afm` by running `pfm2afm.exe`. This utility can be found in the directory `δ:\emtex\utils`. Note, however that not all the information contained in the `.afm` is also available in the `.pfm`, so there may be some loss in quality.

We will not try to explain all the technical details but we will only describe what has to be done to use a PS font within TeX. The first thing that should be done is to convert the `.afm` file into a `.tfm` file. This


```

AFM file      :      .afm
transformation : Raw
encoding type  : NFFS v.1 encoding

```

```

makefont menu:

```

```

-----
choose Font to install
Install font
change Encoding type
choose Transformation
edit font Setting files
Print font table
View font table
show Log file font table compilation
Return to main menu

```

```

make your choice...

```

```

----- (c) 4U 1991-1997 -----

```

Figure 4.22: The InstPsFont menu

is done by the program AFM2TFM. As a result of differences between font encodings³ we also need a virtual font. The virtual font is generated by the programs `vptovpl.exe` and `vptovf.exe`. When we want to view or print the font a bitmap needs to be generated (i.e. a `.pk` file). This `.pk` can be generated by using the `ps2pk.exe` program.

If a DVI-driver cannot find a specific font 4TeX will check if PostScript or METAFONT sources are available. If it is a PostScript font it will call `ps2pk.exe` to generate a bitmap. If it is a METAFONT font it will call METAFONT. The user does not't need to bother, after the detection of a missing font, 4TeX will check a file called `c:\texfiles\4system\psfonts.inf`⁴ if it is a PS font. If you want bitmaps of PS fonts the PS fonts must to be listed this file. If you want to use the PostScript font with `dvips` to print on a PS printer, then you need to add this font to the `c:\texfiles\4system\psfonts.map` file. If you want to use the PS font with Ghostscript the file `c:\texfiles\4system\fontmap` must be updated. Of course all these files are updated automatically by the InstPsFont utility.

F choose Font to install

You can type the font name or use the wildcard options as anywhere else in the 4TeX workbench. You can select from all the `.afm` files that are stored in the directory specified by the environment variable `MYPSFONTS`. This variable is stored in the file `c:\texfiles\4system\texuser.<os>`.

I Install font

After selecting the font, the encoding and the type of transformation you now can convert the PS font for use within T_EX, simply by pressing this key. Because not all the PS fonts follow Karl Berry's font naming rules, 4TeX takes the first 5 letters of the PS font and the last character of the font name to make the (L)T_EX font. For example the PS font `testfont.ps` will result in the T_EX font `testft`. After

3. Font encoding can be seen as a way of storing the letters of a font into a fonttable that can be used within T_EX.

4. Assuming that the 4allT_EX CD-rom installation is done on the `c:` drive

generating the files needed within T_EX and updating the font-setting files, 4T_EX will tell you how to use the font within your document. Converting the font testfont.afm will result in the following message:

```
The PostScript font testft is now available for use
within TeX. Within LaTeX this is done by
defining the font, e.g.
```

```
    \font\ownfontname=testft
or
    \font\ownfontname=testft at 12pt
or
    \font\ownfontname=testft scaled \magstep2
```

```
The font is activated e.g. by the command
{\ownfontname This is a test}.
```

```
When using NFSS v1 just look at the file
d:\emtex\latex209\ps\avantgar.sty
and with NFSS v2 look at the file
d:\emtex\latex2e\psnfss\times.sty
how the new font can be used within NFSS.
```

After converting the font you can view or print the font on a test page.

change Encoding type

You can select one of the following three font encodings:

- No font encoding, i.e. use the Adobe encoding
- old NFSS v.1 encoding, i.e. use the T_EX encoding
- new NFSS v.2 (Cork) encoding

The default encoding is the old NFSS v.1 encoding.

choose Transformation

With this key you to select one of the following transformations:

- Raw (no transformation)
- Caps-small caps
- Oblique
- Narrow.

The default transformation is Raw (no transformation). The type of transformations as well as the use of fonts are clearly explained in the “L^AT_EX companion”.

edit font Setting files

You can edit the font-setting files: psfonts.map, psfonts.inf, and fontmap.

Print font table

After installing the font you can print a test page of the newly installed font.

View font table

After installing the font you can view a test page of the newly installed font.

show Log file font table compilation

In case there are some errors during viewing/printing the test page you can edit the Log file to look for errors.

```

===== 4TeX v4.71 =====
AFM files:      (Normal)      (Bold)      (Italic)      (boldItalic)
                .afm          .afm          .afm          .afm
PostScript family:      .fti (      )
stylefile      :      .sty
transformations : Adobe st. encoding [Roman]

font family installation menu:
-----
choose foNt to install (N B I D)
choose Stylefile name
choose Transformations and encodings
quick-n-dirty preView
install font Using afm2tfm
Append map file to PSFONTS.MAP
test stYlefile with font table
Edit font mapping files
install font using fontinst
Print font table
show Log file font table compilation
Move all files to final destination
make your choice...
===== (c) 4U 1991-1997 =====

```

Figure 4.23: InstPsFam menu

R Return to main menu

Returns you to 4TEX's main menu. Equivalent keys are **Q** and **Esc**.

4.14 The Install PostScript font family menu

Installing PostScript fonts and defining font families in L^AT_EX style is rather complex. It requires deep knowledge of font encodings, conversions of PostScript font metrics (.afm files) to T_EX font metrics (.tfm files), and knowledge of L^AT_EX's font selection scheme.

This menu appears after choosing InstPsFam in the run Utilities selection menu of the main menu. This utility makes it easy to install a complete PostScript font family for use with (L^A)T_EX. Also the necessary fonttable files are updated and an appropriate style file (for use with L^AT_EX 2_ε) is created. To install a font family one is advised to use the sequence of the keys of the menu as displayed in figure 4.23.

Note: This menu is still in betatest. It is likely to contain errors and may change in new releases.

N, **B**, **I**, **D** choose foNt to install (N B I D)

The 4TEX system will use the Karl Berry font naming convention to choose a font family. I.e. by pressing the **N** one can select one of the PostScript fonts in the directory specified by the environment variable MYPSTONTS. This variable is set in the file c:\texfiles\4system\texuser.<os>. 4TEX will then assume that the Karl Berry naming is in use and that it is the Normal (or roman) font. It also will (if those fonts exist) choose the appropriate fonts for the Bold, Italic, and BoldItalic font. E.g. choosing the font bbr.afm will indicate that we want to install the Bembo font family for use within (L^A)T_EX. Using the Berry naming scheme we will have the Bold font bbb.afm, the Italic font bbro.afm, and the BoldItalic

```

===== 4TeX v4.71 =====
AFM files:      (Normal)      (Bold)      (Italic)      (boldItalic)
                bbr.afm      bbr.afm      bbr.afm      bbr.afm
Postscript family : bb.fti Bembo

Normal font families      Special font families
-----
R: Roman(serif) font      S: symbol font (postscript symbols)
F: Sansserif font        M: math font (TeX cmmi)
T: Typewriter(fix-space) Y: math symbol font (TeX cmsy)
                          V: math extension font (TeX cmex)
                          U: unknown encoding

Choose family type

===== (c) 4U 1991-1997 =====

```

Figure 4.24: choose Transformation and encodings menu

font `bbbo.afm`. By using the other keys one can choose a font that is different from the Karl Berry convention. We, however, strongly discourage this because several things in the `FONTINST` program will not work correctly, resulting in an incomplete fontfamily for use with \LaTeX .

[S] choose Stylefile name

the `InstPsFam` utility will check the font family name (in our example \LaTeX will see that it is the Bembo family). For proper use within $\LaTeX 2_{\epsilon}$ we need a style file that will define the Bembo family as the default font family. By pressing the **[S]** one can change the name of the appropriate style file that will be created.

[T] choose Transformations and encodings

After pressing this key the menu as show in figure 4.24 will pop up and you can select the transformation/encoding that is needed. After choosing the appropriate transformation \LaTeX will ask whether you want to generate an extended, narrowed, slanted, and a small caps font of the family as well.

[V] quick-n-dirty preView

With this key you can generate a rough version of the files necessary for use with \LaTeX . The result will be a quick-and-dirty preview of the font. Note that this is not the final version of the font but it will give you a first impression of the font.

[U] install font Using `afm2tfm`

As with the `InstPsFont` utility the necessary files for use with \LaTeX are generated. Note that the quality of the fonts can be improved by running `FONTINST`.

[A] Append map file to `PSFONTS.MAP`

With this key the fonttable file `psfont.map` will be updated. If you forget to update this file you can not use the font family within \LaTeX simply because the created files necessary for the fonts will be deleted after quitting the `InstPsFam` utility.

Y test stylefile with font table

The font family will be used together with the generated style file for L^AT_EX 2_ε to view a two page sample document.

E Edit font mapping files

You can edit the font-setting files. E.g. with the Bembo family you can edit `bb.map`, `bb.fti`, `psfonts.map`, `psfonts.inf`, and `fontmap`.

O install font using fontinst

The program FONTINST is the best way to install PostScript fonts for use with T_EX. By pressing this key the FONTINST is called and the font family is automatically installed. This procedure may take much time to complete.

P Print font table

The font family will be used together with the generated style file for L^AT_EX 2_ε to print a two page sample document.

L show Log file font table compilation

In case there are some errors during compiling the test pages you can edit the log file to look for errors.

M Move all files to final destination

With this key you will install the font family from their temporary location to their final location. L^AT_EX will show how the newly generated font family can be used, e.g. after installing the Bembo family we get the following messages on screen:

```
The PostScript font bembo (bb) is now ready for use
in TeX. This can be done by declaring the font, e.g.
(TeX:) \font\ownfontname=bbr
(LaTeX:) \newfont{\ownfontname}{bbr scaled \magstep2}
The font is activated e.g. by the command
        {\ownfontname This is a test}.
```

```
This font family can be used as follows:
        \fontfamily{bb}\selectfont
```

```
This family is used as default for roman/sansserif like this:
(LaTeX 2.09:) \documentstyle[bembo]{article}
(LaTeX 2e:) \usepackage{bembo}
```

Note: If you don't use this key before returning to the main menu the font will *not* be installed.

R Return to main menu

Returns you to the L^AT_EX's main menu. Equivalent keys are **Q** and **Esc**. After pressing this key all temporary files will be deleted.

Part 5

Editors and compilers

5.1 The editors

\LaTeX works in combination with any ASCII editor. The installation script gives you a list of editors from which you can pick your favorite. If you choose an editor of the TSE (The Semware Editor) family (TSE JUNIOR 4.0, TSE 2.5 or TSE 2.8 you can benefit from several extra features. TSE JUNIOR 4.0 is a shareware editor, formerly known as QEDIT (the executable is still called q.exe). TSE 2.5, the professional version with more features and a much more powerful macro language. On the CD-rom you will find a “test drive” version, which is fully functional but will remind of its status now and then. TSE 2.8 is the 32 bits version of TSE 2.5 and only runs on MS-WINDOWS 95 and NT. Most notable extras are its support for ‘long file names’, the Windows clipboard and syntax highlighting. There is a “test drive” version of 2.8 available, but you will have to install it yourself because it is time-limited: after 60 days it stops working.

For these editors \LaTeX supports a set a macros that we will discuss below.

In TSE JUNIOR 4.0 $\boxed{F1}$ invokes a help screen which shows the meaning of the most important keys. In TSE 2.x on pressing $\boxed{F1}$ a menu appears, containing submenus on general as well as \LaTeX subjects. Below we describe the specific \LaTeX features of both editors. Unless stated otherwise they work in both editors.

\TeX -specific commands:

- By default the main \TeX file as well as the include file are loaded in the editor. By pressing $\boxed{Alt} \boxed{N}$ one can switch between the files that are loaded.
- $\boxed{Alt} \boxed{Tab}$ find matching symbol for {, }, [,], (or). In Windows $\boxed{Alt} \boxed{Tab}$ already has a global meaning. In that case use $\boxed{Alt} \boxed{F3}$.
- $\boxed{Alt} \boxed{key}$ where \boxed{key} is one out of the following list, produces the indicated output. The cursor is positioned in between the two symbols.

$\boxed{(}$	$\boxed{)}$
$\boxed{\{}$	$\boxed{\}}$
$\boxed{\$}$	$\boxed{\$\$}$
$\boxed{\&}$	$\boxed{\&\&}$
$\boxed{=}$	$\boxed{\backslash\{}$

- $\boxed{Alt} \boxed{\backslash}$ choose a \LaTeX environment. A selection list containing many standard \LaTeX environments is presented. Use the cursor keys and $\boxed{Enter} \boxed{\leftarrow}$ to select. In TSE 2.x speed-search can be used: by typing the initial characters of the desired environment the cursor jumps to the intended line.

To use your own list of environments you should set the environment variable `ENVIRONMENT` in `texuser.<os>`.

- `Ctrl I` insert `\index{...}` behind the word at the cursor position. For example, if the cursor is positioned at the word `gnu` then pressing `Ctrl I` results in `gnu\index{gnu}`.
- `Alt %` display a menu of sample T_EX files. Use cursor keys and `Enter ←` to select one. This file will be copied into the current file at the cursor position.
- `Ctrl B` run BIBDB, the program to manage the BIBT_EX database files. See Section 7.1.2 for a detailed description.
- `F3` show the on-line T_EX documentation in the directory `δ:\emtex\doc`. First a list of files containing the documentation is presented. Use the cursor keys and `Enter ←` to select one. The documentation file is displayed by an appropriate program, depending on the type of file. Press `Esc` twice to return to the editor. *Note:* This command is also available in all 4T_EX menus. *Warning:* The documentation is on a very technical level and not always easy to read. You may prefer to consult a more experienced user instead.
- `Ctrl G` replace the word at the cursor position by a string from `gloss.txt`.¹ For example, if the file `gloss.txt` contains a line `be \begin{eqnarray}` then on typing `\be` followed by `Ctrl G` this will be replaced by `\begin{eqnarray}`. Once this facility has been used in an editing session, the file `gloss.txt` will remain in the editor and new lines can be added.
- `Alt /` show a list of all labels in the current bibliographic database (TSE 2.x only). Use the cursor keys and `Enter ←` to select a label. Output of the macro is the selected label in braces as used in `\cite` and `\nocite` commands.
- `F10` compile and view the current block. 4T_EX recognizes two special cases when `F10` is pressed:
 - if no block is defined then the current line is treated as if it were a block;
 - if the current format is SLIT_EX then the whole slide containing the current block is compiled and viewed.

To ensure that all necessary and macros are included, 4T_EX creates and then compiles a temporary file containing

- i) all lines of the main T_EX file up to and including the line containing the string `\begin{document}`;
- ii) the current block (see above);
- iii) the line `\end{document}`.²

- `Alt F10` compile and view the current file. It is the same as `F10` with the current block equal to the current file.
- `Ctrl N E` (TSE 2.X only) find next error. To be used to locate and correct errors reported by the T_EX compiler. All errors in all files (also e.g., `.sty` files) are handled in the order in which they appeared. Each time, the error message generated by T_EX (e.g., `Undefined control sequence`.) is displayed on the top line of the screen.
- `F5` user defined utilities. This is the same feature as defined in all 4T_EX menus.
- `Alt F1` or `Ctrl F1` check/correct the spelling of the word at the cursor position. For a detailed description of the spell-checker `AMSPELL` see the discussion on the command `check Spelling` and Section 7.3. Press `Ctrl F1` to check spelling in the default language (see Section 4.1) of the current main T_EX file. Press `Alt F1` to select a language first. If the word at the cursor position is correctly spelled, you will be immediately returned to the editor. How to proceed if a spelling error was found is described in Section 7.3.

1. If this file does not exist then it will be created and loaded into the editor.

2. Obviously, this procedure assumes you are using the L^AT_EX or SLIT_EX format. Plain T_EX users can ‘trick’ 4T_EX by including the text `% \begin{document}` in their main file. The T_EX compiler will not stumble on it because it is a comment. The `\end{document}` command inserted by 4T_EX should work normally, provided that `\end` has not been redefined.

- `Alt =` search for the next occurrence of the word at the cursor position. `Ctrl =` will search for the previous occurrence of that word. These two macros are only supported by TSE 2.x.
- `Alt S` will display several tables (\LaTeX commands and result) of mathematical symbols.
- `Ctrl Ins` (TSE 2.x only) will scan the main document and all include files for labels. A list of all labels is displayed and you are prompted to select one label. The output of the macro is that the selected label is entered in your document. This macro makes it possible to construct a reference to a label without exactly knowing/remembering the label name.
- `Alt X` save and exit all files. Just before this is done, a marker is placed at the cursor position in each file that is loaded in the editor.³ This marker is used to restore the cursor position on reentrance.

5.2 The compilers

The $\text{EM}\TeX$ \TeX compiler for MS-DOS comes in six different versions:

<code>tex86.exe</code>	for 8088/8086 processors
<code>tex186.exe</code>	for 80186 and better processors
<code>btex86.exe</code>	for 8088/8086 processors and very large documents
<code>btex186.exe</code>	for 80186 and better processors and very ‘difficult’ documents
<code>tex386.exe</code>	for 80386 and better processors; this is a ‘big’ version
<code>htex386.exe</code>	for 80386 and better processors; this is a ‘huge’ version. Its capacity is even higher than ‘big’ \TeX . There is no huge \TeX compiler available for processors lower than 80386.

However, in $\text{4}\TeX$ only the 80386 versions are used, i.e., `tex386.exe` and `htex386.exe`.

These programs use the EMX dos-extender by default. If they detect there is no VCPI-host, but there *is* a DPMI-host, they will use the RSX dos-extender instead. This situation is likely to occur if you run $\text{4}\TeX$ in an MS-DOS-box under MS-WINDOWS or OS/2.

If your machine doesn’t have at least 3MB of memory, `[h]tex386` may run very slowly. You’ll need even more memory if a large part of it is already occupied (RAM disks, operating system).

Memory settings

In case you find that \TeX cannot compile a file because its capacity is exceeded (it will write such a message on screen and in the log file), and you are sure the \TeX code is correct, you may need to use the huge version of the \TeX compiler (see above).

The \TeX compilers have default values for memory settings which can be changed to some extend by setting the environment variable `EMTEXOPT`. In section 5.2.2 some settings are listed. See the $\text{em}\TeX$ documentation (use `F3`) for more details and the `Ctrl E` key in the main menu (Section 4.1).

The variable `EMTEXOPT` can be changed by pressing `Ctrl E` in the main menu. $\text{4}\TeX$ will show the current setting and all possible parameters and their ranges. You can edit the current setting. Press `Enter` when you have finished editing, and you will return to the main menu.

3. This is why you may find a line containing only the string `%<TeX_Marker>` in a `.tex` file if it is viewed outside $\text{4}\TeX$.

Note: The variable EMTEXOPT is automatically reset to its default by \LaTeX each time you choose a new main file or format file.

5.2.1 Preparation

The remainder of this section is of interest mainly for those who want to set up their own system. Or maybe you are just curious about what happens behind the scenes.

In order to make installation easier we define the environment variable EMTEXDIR which denotes the root of the complete \LaTeX system, e.g. if you have installed \LaTeX on the c: drive you set EMTEXDIR=C:\EMTEX. In case you run \LaTeX from CD-rom and your CD-rom drive is drive D: you set EMTEXDIR=D:\EMTEX.⁴

\TeX also needs some environment variables that tell it where to look for its files:

▷ set TEXTFM=%EMTEXDIR\tfm
for .tfm files that contain font metrics.

▷ set TEXINPUT=%EMTEXDIR\texinput
for styles and macros which are not in the format file; see below.

▷ set BTEXFMT=%EMTEXDIR\btexfmts
for .fmt files, that contain hyphenation patterns and macros.

The big \TeX version `tex386` looks for the parameter BTEXFMT:

▷ set BTEXFMT=%EMTEXDIR\btexfmts
The huge version (`htex386`) looks for HTEXFMT.

Note: When \TeX tries to find a any \TeX text or macro input file it first searches the current directory, then the directories listed in TEXINPUT. We suggest that you run \TeX from the directory where you keep your text files. However, it is also possible to add a directory to the TEXINPUT variable, e.g.,

▷ set TEXINPUT=%EMTEXDIR\inputs;c:\texfiles\texinput

\LaTeX will do that for you if you set MYTEXINPUT in `texuser.<os>` to such a directory. Similar remarks apply to the [B|H]TEXFMT variable and .fmt files.

The $\EM\TeX$ compilers can automatically search subdirectories of a given path. E.g., if you specify TEXINPUT=%EMTEXDIR\inputs! (note the exclamation mark), the compiler will search %EMTEXDIR\inputs and all its subdirectories (e.g., %EMTEXDIR\inputs\pstricks and %EMTEXDIR\inputs\bibtex) for input files. Note that the order in which subdirectories are searched is not determined. One exclamation mark stands for searching one level deeper, no more. If you want the compiler to search an the entire branch (subsubdirectories and deeper) you should supply two exclamation marks after the path name.

Beware that that currently not all programs that use these variables are capable of subdirectory searching, and some programs use a different syntax. \LaTeX deals with these incompatibilities where necessary.

5.2.2 Command line

The syntax of the compilers is

```
[H]TEX386 [options] [&<format>] [input file]
```

4. On a Novell network the EMTEXDIR variable can be used in combination with a mapping command to access the $\EM\TeX$ files:

▷ set EMTEXDIR=t:\
▷ map root t:=sys:appls\emtex

In the on-line documentation you can find a complete listing of options. Only a few options need to be described here (see also `Ctrl | E` in the main menu; Section 4.1):

- `/maxxx` parameter stack (macro parameters). You can increase this value (to some extent) if your \TeX job requires a larger stack than the default.
- `/mf#` font memory (font metric data). You can increase this value (to some extent) if your \TeX job requires a larger font memory than the default.
- `/mm#` main memory. You can increase this value (to some extent) if your \TeX job requires a larger main memory than the default.
- `/mn#` semantic nest size (mode nesting). You can increase this value (to some extent) if your \TeX job requires a larger semantic nest size than the default.
- `/mp#` pool size (strings). You can increase this value (to some extent) if your \TeX job requires a larger pool size than the default.
- `/msxxx` sets the save size (values saved by grouping). You can increase this value (to some extent) if your \TeX job requires a larger save size than the default.
- `/mt#` sets memory available for hyphenation patterns to # bytes; the maximum allowed is `/mt:65000`, the minimum is `/mt:5000`. The amount you specify must always be at least as much as was specified when generating the format file you are using (see also Section 4.8).
- `/b` (batchmode) tells \TeX to not to prompt the user for input when errors occur, but to continue as best as it can. Error messages etc. are written to `<filename>.log`, where `<filename>` is the input filename you gave.
- `/axxx` is for automatically calling an editor from within \TeX . See the next section for a description of this feature.

The `&<format>`⁵ parameter indicates a format file with preloaded macros and hyphenation patterns. These files reside in the directory indicated by `[B|H]TEXFMT`. For plain \TeX the format is `plain`; for \LaTeX `latex2e`. For the (obsolete) \LaTeX version 2.09 the format is `lplain`. Most \LaTeX formats provided by $\4TeX$ contain hyphenation patterns for English, French, Dutch and German—which is why extra memory is claimed for hyphenation patterns. Format files are generated by running the \TeX compiler with the `/i` option. This process is described in Sections 4.8 and 5.2.4.

`[input file]` should represent the name of a text file. An extension `.tex` is assumed if none is given. If no input file is given, \TeX will run interactively, i.e., it will show the \TeX prompt from which you can enter text.

Thus, a \TeX command line might look something like

```
▷ tex386 /mt:27000 &latex2e mytext
```

This command processes `mytext.tex`, using the (precompiled) \LaTeX macro package `latex2e.fmt` and creates `mytext.dvi` from which the appropriate printer program can generate output. Error messages, statistics etc. will be written to the logfile `mytext.log`.

5. Note that your command interpreter may already have given the ‘&’ symbol a special meaning. E.g., in OS/2 it is defined by default to be the command separator, allowing you to specify multiple commands on one input line. The same applies to 4NT (4DOS for MS-WINDOWS NT). $\4TeX$ redefines it so you don’t have to worry about that, but if you want to run \TeX from you own command line you will have to either redefine ‘&’ or ‘escape’ it. See the manual of your command line interpreter for details.

Any selection of options can be put in the environment variable `EMTEXOPT`—as is done by Δ TeX. For example:

```
▷ set EMTEXOPT=/mt:27000 &plain
```

This would simplify the above command line to

```
▷ tex386 mytext
```

For details on running TeX, see your favorite TeX book. Remember: if you get the ? prompt, then you can terminate the TeX compilation by typing `[X]` followed by `[Enter ←]`.

In case TeX can't find a file and asks you to type in a name, you can stop TeX by entering `[Ctrl][Z]`, or cause it to skip a file by entering `null`. You can interrupt TeX by pressing `[Ctrl][Break]`.

5.2.3 Connecting an editor

TeX files can be prepared using any ASCII editor. It is very convenient if the editor can display line numbers, as we will discuss below. While PC Magazine's TED won't do, the NORTON EDITOR, EMACS, VDE and WORDPERFECT's PE are all fine. If you don't mind changing editors, we suggest using TSE if you run Δ TeX on MS-DOS, OS/2 or MS-WINDOWS 3.x. If you run MS-WINDOWS 95 or MS-WINDOWS NT PFE would be an excellent choice.

When TeX encounters an error and prints a questionmark, one of the answers you can give is `[E]` (from Editor). If the system is set up properly, this answer results in TeX being terminated and the editor of your choice being called.

Here is how it works: if TeX was called with an option `/a<filename>` and you entered `[E]` after the ? prompt, TeX will write certain information to `<filename>`. The environment variable `EMTEXED` controls what information. `EMTEXED` may contain parameters %1 for the line number, %2 for the input file and %3 for the logfile. For TSE JUNIOR 4.0 you might set `EMTEXED` as follows:

```
▷ set EMTEXED=c:\tsejr40\q.exe %2 -n%1
```

Then TSE would load %2, which is the input file, and jump to line %1, which is the line where the error occurred. For PFE

```
▷ set EMTEXED=pfe32 /s /g %1 %2
```

would have a similar effect. For VDE:

```
▷ set EMTEXED=vde %2,%3
```

With VDE you can't enter a line number on the command line; all you can do is load both the input file and the logfile.

Note: In a batch file (e.g., `autoexec.bat`) you'll need double percent signs:

```
▷ set EMTEXED=c:\tsejr40\q.exe %%2 -n%%1
```

Now we can load TeX from a batch file such as the following:

```
@echo off
if exist e.bat del e.bat
tex386 /mt:65000 /ae.bat &plain %1
if exist e.bat e.bat
```

When the compiler has finished, it may have written the batch file `e.bat`. If this batch file exists, it is called. `e.bat` in its turn will call your editor.

5.2.4 Format files

Δ TeX version 4.71 as you find it on the CD-rom supports the following format files:

- \LaTeX 2 ϵ (`latex2e.fmt`)

- plain \TeX (plain.fmt)
- extended plain \TeX (eplain.fmt)
- \LaTeX 2.09 (lplain.fmt and lplainn.fmt with NFSS)
- $\SL\TeX$ (splain.fmt and splainn.fmt with NFSS)
- Greek \TeX (greek.fmt)
- MeX Polish plain \TeX (mex.fmt)
- MeX Polish \LaTeX 2.09 (lamex.fmt)
- Polish \LaTeX 2 ϵ (plllatex.fmt)
- Greek \LaTeX 2.09 (lgreek.fmt)
- Cyrillic AMS- \TeX (lhamste.fmt)
- Cyrillic \LaTeX 2.09 (lhllatex.fmt and blhlnfss.fmt with NFSS)
- Cyrillic plain \TeX (lhplain.fmt)
- Czech/Slovak \LaTeX 2 ϵ (cslatex.fmt)
- Slovakian \TeX (slovak.fmt)
- BLUe \TeX (blue.fmt)
- \TeX sis (texsis.fmt)
- \TeX info (texinfo.fmt)
- The Lollipop format file (lollipop.fmt)
- Slav \TeX \LaTeX 2.09 (sllplain.fmt)
- Slav \TeX plain \TeX (slplain.fmt)
- Con \TeX t (contxen.fmt, contxnl.fmt and contxde.fmt)

Except for CON \TeX t, which is only available as ‘huge’ format, all formats are available in both ‘big’ and ‘huge’ format.

For documentation on plain \TeX , \LaTeX , and others we refer to the books listed in the bibliography and the on-line documentation.

Babel

With \TeX 3.0 it has become possible to load hyphenation patterns for more than one language. To take advantage of this new feature some extra book keeping is needed. BABEL is a \TeX program that provides a simple way to generate \TeX format files (see below) with multiple languages, and some control sequences to switch languages within one \TeX document.

\LaTeX supports hyphenation patterns for the *UK English, US English, Dutch, German, French, Russian, Portugese, Greek, Italian, Spanish, Polish, Turkish* and several other languages, some even in different flavors. This means that you can generate a format (see Format menu, Section 4.8) that supports hyphenation for one or more of the languages above. From within a \LaTeX document you can switch to and from any of these languages, provided you have included the matching \LaTeX style files such as *français* or *dutch* (see documentation of BABEL).

Generating new format files

On the CD-rom (nearly) all format files are available in two versions: one ‘big’ version (for Big \TeX) and one ‘huge’ version (for Huge \TeX). The first resides in the directory indicated by BTEXFMT, and is generated by TEX386. The second resides in the directory indicated by HTEXFMT, and is generated by HTEX386.

The easiest way to generate new formats is to start \LaTeX and run the Format utility (see Section 4.8). Below we will describe step by step how you can generate a format file from the command line. All the

environment variables that are needed are set by Δ TeX, so the easiest way to generate formats is from a MS-DOS shell *on top of* Δ TeX. Use F9 from the main menu to start the shell.

First we will show how to generate the standard *plain* format.

1. Enter
 \triangleright set TEXPROGRAM=tex386
to specify the TeX compiler.
2. Change directory to `c:\texfiles\4system`. This is where the format file will be written. If all goes well you can move it to the correct directory later.
3. Enter
 \triangleright %texprogram /i /c437.tex plain
to generate the plain TeX format using the codepage 437 definition. If you don't want to load a specific codepage you should use /8 instead to allow for pure 8 bit input.
4. After a few seconds the compiler will ask you to enter the name of the file that contains the hyphenation patterns. You can enter:
`ushyphen.tex`
to load US English hyphenation patterns.
5. A little later the compiler will display its prompt (*). Now you can enter
`\dump`
The file `plain.fmt` will be written to disk.
6. Now move this format file to the directory δ : `\emtex\btexfmts`.

The 'huge' version is generated in much the same way, with these exceptions:

- You must use the huge version of the compiler: use `%hugetexprogram` in stead of `%texprogram`.
- You must move the format file to the directory δ : `\emtex\htexfmts`.

Next we will generate the standard \LaTeX format.

1. Enter
 \triangleright set TEXPROGRAM=tex386
to specify the TeX compiler.
2. Change directory to `c:\texfiles\4system`. This is where the format file will be written. If all goes well you can move it to the correct directory later.
3. Enter
 \triangleright %texprogram /i /8 latex.ltx
to generate the \LaTeX format using 8 bit input.
4. After a few seconds the compiler will ask you to enter the name of the file that contains the text font information. You should enter:
`fonttext.lx`
to get the standard definitions.
5. A few seconds later the compiler will ask you to enter the name of the file that contains math font information. You should enter:
`fontmath.lx`
to get the standard definitions.
6. Then the compiler will ask you to enter the name of the file that contains font preloading information. You should enter:
`preload.lx`
to get the standard definitions.
7. Now the compiler will read the file `hyphen.cfg` which enables the Babel multiple language system. This system will load the file `language.dat` (if available) in which you must have specified which languages you will be using with this format.

8. The file `latex.fmt` will be written to disk.
9. Now move this format file to the directory δ : `\emtex\btexfmts`.

The ‘huge’ version is generated in much the same way, with these exceptions:

- You must use the huge version of the compiler: use `%hugetexprogram` in stead of `%bigtexprogram`.
- You must move the format file to the directory δ : `\emtex\htexfmts`.

Generating the \LaTeX 2.09 format files with the New Font Selection Scheme takes a few more steps.

1. Enter
 - \triangleright `set TEXPROGRAM=tex386`
to specify the \TeX compiler.
2. Change directory to `c:\texfiles\4system`. This is where the format file will be written. If all goes well you can move it to the correct directory later.
3. Enter
 - \triangleright `%texprogram /i /c437_tex lplain`
to generate the plain \TeX format using the codepage 437 definition.
4. After a few seconds the compiler will ask you to enter the name of the file that contains the hyphenation patterns. You should enter:
`babel.hyp`
to get the multi-language version.
5. A few seconds later the compiler will ask you to enter the name of the file that contains font definitions. You should enter:
`lfonts.nfs`
to get the standard definitions of NFSS.
6. Then the compiler will ask you to enter the name of the file that contains the set of font definitions that you require. You should enter:
`fontdef.ori`
7. You must also enter the name of the file that defines which fonts will be preloaded. You should enter:
`preload.ori`
8. Next you enter the name of the style file that determines the default font selection mechanism. You should enter:
`newlfont.sty`
9. A little later the compiler will display its prompt (*). Now you can enter
`\dump`
The file `lplain.fmt` will be written to disk.
10. Rename this format file to `lplainn.fmt`.
11. Move the format file to the directory δ : `\emtex\btexfmts`.

The ‘huge’ version is generated in much the same way, with these exceptions:

- You must use the big version of the compiler: use `%hugetexprogram` in stead of `%texprogram`.
- You must move the format file to the directory δ : `\emtex\htexfmts`.

The $\text{SL}\TeX$ formats (`splain.fmt`) are generated in almost the same way as the *plain* formats.

Part 6

Print and preview programs

6.1 \TeX dvi-drivers

\TeX supports a variety of output devices. There are separate print or preview programs for separate groups of output devices:

DVIDOT	for dot-matrix printers and bitmapped output such as PCX picture files (e.g., for fax machines).
DVIHPLJ	for LaserJet, DeskJet and other PCL printers.
DVIPS	for PostScript printers. This print program will be described in Section 6.2.
GHOSTSCRIPT	for printing PostScript output on non PostScript printers or the screen (previewing). This program will be described in Sections 6.2.6 and 6.2.5.
DVIDJC	for color DeskJet printers. This print program will be described in Section 6.3.
DVI2TTY	for printing or previewing on non graphic/ASCII output devices. It simply emulates the page layout as if you were using a type writer.
DVISCR	is not a print but a preview program; we describe it here because it is very similar in structure to the print programs. See Section 6.1.6 on how to operate DVISCR interactively.
DVIVGAC	for previewing colored output for DeskJet color printers. This program will be described in Section 6.3.1.
DVIWIN	is a WINDOWS previewer program that can also be used to print. See Section 6.4 for a detailed description.

All these programs take a `.dvi`-file generated by a \TeX -compiler as input. We refer to them collectively as *DVI-drivers*.

The programs DVIDOT, DVIHPLJ and DVISCR are part of the $\text{EM}\TeX$ package written by Eberhard Mattes and share similar interfaces. These programs exist in two versions: one standard, and one that needs a mathematical coprocessor. The latter will run only on machines equipped with a 80x87 coprocessor, and they will run a little faster. These programs have a '7' appended to their file name, e.g., `dvidot7.exe`. \TeX will use the coprocessor versions automatically if it detects a coprocessor in your computer.

DVISCRS is an alternative to DVISCR that uses less memory at the expense of performance. This program is especially handy if DVISCR issues out of memory errors due to loading many (high resolution) fonts.

DVIHPLJ3 is a 32 bits version of DVIHPLJ which runs faster and can print larger pages (e.g., A3) at high resolutions (up to 1200 dpi).

DVI2TTY was written by S. Lindahl and extended by M. Mol. It is used only for quick viewing/printing dvi files and thus has a much more primitive interface.

DVIPS was written by T. Rockiki.

The color DeskJet drivers were written by F. Sowa.

GHOSTSCRIPT was written by Alladin Software Inc.

The WINDOWS previewer GSVIEW that uses GHOSTSCRIPT as its 'rendering engine' was written by Russell Lang.

DVIWIN was written by H. Sendoukas.

6.1.1 Fonts

Whereas the \TeX -compiler only needs to know the font *metrics*, the preview and/or print programs usually require *bitmaps* of each character of each font used, and must be told where to find these.

Fontfiles `.pk` can be organized into font libraries `.fli`, which can be read directly by print and preview programs. The program FONTLIB (see Section 4.7) manages font libraries.

The CD-rom version of \LaTeX does not use font libraries to store bitmapped fonts because not all of the supported DVI-drivers can read font libraries: so far only the $\text{EM}\TeX$ drivers and DVIPS can. The current version of the color DeskJet dvi drivers are even more limited. They can only read bitmapped fonts from one directory tree (e.g., `\emtex\texfonts\deskjet`) and its subdirectories. All other fonts you need should be accessible from the current directory.

There are no separate screenfonts; the $\text{EM}\TeX$ previewer uses printer bitmaps (`.pk` files) of any resolution.

6.1.2 $\text{Em}\TeX$ DVI-drivers command line

A few comments before we embark on the description of the command line:

- You may wonder how you can fit even the required options on a command line.¹ The answer is that you don't have to: you can put options in environment variables and *response* files instead. This is described in Section 6.1.3, where we also give complete examples.
- This section describes the simpler options. These will probably suffice if you want to set up your own system. Section 6.1.5 describes the \LaTeX response files. These are considerably more complex, in part because they support additional options, but mainly because they must accommodate many users, each with their own preferences and directory setup.
- Options not described here include: scaling of your output; printing 2-up; font substitution if a specified font cannot be found; changing the order in which pages are printed; moving the printed area by a horizontal and/or vertical offset; options for inclusion of graphics (these are described in Section 7.4); support for alternative directory configurations; automatic font generation (see Section 6.5.3) etc. Complete information is available in `\emtex\doc\programs\dvidrv.zip`.
- Parameter files, configuration files and font substitution files are located in the directory `%emtex-dir\data`.

1. MS-DOS allows 127 characters; 4DOS 255.

The syntax is

DVIxxx [*parameter file*] <*input file*> [*output file*] [*options*]

where *DVIxxx* is the print or preview program of choice; if necessary with drive and/or directory specification.

[*parameter file*] (e.g., *fx80.dot*) is required only for *DVIDOT*.

<*input file*> is a required parameter. An extension *.dvi* is assumed if none is given.

[*output file*] is normally *prn*; for *DVIscr*, it should be omitted. A filename can be entered instead of *prn*. The resulting file (e.g., *mytext.lj*) can be copied to a printer in a separate step by the DOS command

▷ copy /b mytext.lj prn

Options may occur anywhere on the command line, except that for *DVIDOT* they should not precede [*parameter file*]. The options include:

/? display help
/?? display extensive help

dvihplj-specific options

/opxxx printer class. Default is *laserjet*, other classes are e.g.: *deskjet*.
/oi+ reset printer before printing (default)
/om# printer memory (see below)

dviscr-specific options

/oi+ inverse: white on black
/s# scaling: map #×# printer dots to one screen pixel
/o# as /s#, but, if possible, with grayscaling
/oa19 Olivetti M24 640 × 400 display
/oa8 Tseng Labs Super VGA graphics card
Super VGA modes:

Acumos AVGA2, AVGA3	/oa103:800:600:58
Advance Logic AL2101	/oa103:800:600:2B
Ahead V5000	/oa103:800:600:71
ATI VGAWONDER, Graphics Ultra etc.	/oa103:800:600:54
Chips and Technologies	/oa103:800:600:6A
Cirrus Logic CL-GD 500/600	/oa103:800:600:64
Cirrus Logic GD 5422	/oa103:800:600:58
Compaq VGA	/oa103:800:600:59
CTI	/oa103:800:600:6A
Diamond Stealth	/oa103:800:600:4F02:0102
Everex	/oa103:800:600:70:02
Genoa 5xxx, Sigma VGA	/oa103:800:600:29
Genoa 6xxx	/oa103:800:600:6A
MXIC MX 68010	/oa103:800:600:55
NCR 77C22	/oa103:800:600:58
OAK Technologies OTI-067, OTI-077, OTI037C	/oa103:800:600:52
OAK Technologies OTI037C w/ NEL BIOS	/oa103:800:600:5B

Orchid Prodesigner	/oa103:800:600:29
Paradise	/oa103:800:600:58
Poach	/oa103:800:600:6A
Primus	/oa103:800:600:2A
Realtek RT 3106	/oa103:800:600:1F
S3	/oa103:800:600:4F02:0102
Tecmar	/oa103:800:600:16
Trident 8900	/oa103:800:600:5B
Tseng ET-3000, ET-4000	/oa103:800:600:29
VEGA	/oa103:800:600:29
VESA compatible (800 × 600)	/oa21
VESA compatible (1024 × 768)	/oa22
VESA compatible (1280 × 1024)	/oa23
Video 7 SVGA	/oa103:800:600:62
WD90C11	/oa103:800:600:5C
Western Digital	/oa103:800:600:58

In `texuser.<os>` you can specify e.g., `SCRTYPE=103:800:600:58` if your computer is equipped with a Paradise graphics adapter. `SCRTYPE=8` is sufficient for a Tseng ET-3000 or ET-4000. Note that the `/oa` part must be omitted in `texuser.<os>`. If your graphics adapter is not listed, try one of the VESA modes.

page selection

- `/b#` start printing at page #; hierarchical page numbers are also possible, e.g., `/b3.2`
- `/e#` stop printing after page #

page dimensions (see below):

- `/h#` set page height to #; the default unit of measurement is inches, but you can also use, e.g., millimeters: `/h297mm` is equivalent to `/h11.69` and to `/h=11.69in`
- `/w#` set page width; `/w8.27` is equivalent to `/w210mm`

resolution parameters (see below):

- `/r#` set horizontal and vertical printer resolution to # DPI (Dots Per Inch)
- `/rx#` set horizontal printer resolution
- `/ry#` set vertical printer resolution

fonts

- `/plxxx` specifies font libraries (see below)
- `/ft#` font-size tolerance in % (default 0.2%; maximum 20%); this option allows the DVI-driver to substitute slightly smaller or larger fonts when the exact size is unavailable

Above, ‘#’ stands for a number, possibly followed by a dimension indicator (e.g., inches: `in` or millimeters: `mm`); ‘xxx’ for any text string. The option may be separated from its parameter by either `=`, `:`, `:=` or by nothing at all; a space in between is not allowed. Examples: `/b1`, `/b:1`, `/b=1` and `/b:=1` are all legal.

dvihplj-specific parameters: if you use e.g. a LaserJet 4M that has a lot of extra memory, printing will be more efficient if DVIHPLJ is told about it by the addition of a parameter `/om6000`. Make sure you never specify more memory than is available in your target printer—better be on the safe side. For

LaserJets, the default is 384. For DeskJets, the memory parameter is ignored; a memory cartridge won't do you any good as far as DVIHPLJ is concerned.

Page-dimension parameters: `/h#` and `/w#` are best put in a response file; see below. Note that page dimensions are only used to determine how much of the page can be printed; a wrong page height won't result in 'creeping' pages since formfeeds are used to move to a new page.

Resolution parameters: a LaserJet II/III or DeskJet requires `/r=300`. An ordinary Epson or IBM compatible 9-pin dot-matrix printer such as most inexpensive Star printers requires resolution parameters `/rx=240` and `/ry=216`. Resolution parameters are also best put in a response file. Remember to set the `/rf` parameter identical to `/r` if you want to make use of the automatic font generation features.

Font libraries: a LaserJet or DeskJet would require

```
/pl=c:\emtex\texfonts\lj_{0,1,2,3,4,5,6,7,8,slj}
```

This is shorthand for

```
/pl=c:\emtex\texfonts\lj_0,c:\emtex\texfonts\lj_1,...
```

An Epson-compatible printer would require

```
/pl=c:\emtex\texfonts\fx_{0,h,1,2,3,4,5}
```

6.1.3 Response files

Even with font libraries, it is not practical to enter all font files on the command line. Therefore, *response files* are used which contain names of font libraries and other options. An option `@<filename>` on the command line causes the DVI-driver to read options and parameters from `<filename>` before processing the remainder of the command line.

EMT_EX expects to find response files you refer to to be located in a subdirectory called `data` directly beneath the path indicated by the environment variable `EMTEXDIR`, e.g., `δ:\emtex\data`. However, if you supply a path with the response file name (e.g., `c:\texfiles\ownprt\myprint.cnf` in stead of just `myprint.cnf`) it will read the file you specified.

Since the previewer DVISCR uses printer fonts, it makes sense for response files to be usable for both printing and previewing.

An example response file for a LaserJet (lines starting with % are comments):

```
% A4 paper size
/w=210mm
/h=297mm
% printer resolution in Dots Per Inch
/r=300
/rf=300
% font libraries
/pl=c:\emtex\texfonts\lj_{0,1,2,3,4,5,6,7,8,slj}
```

With such a response file, named e.g., `dvi.cnf`, a file `mytext.dvi` could be previewed by entering

```
▷ dviscr mytext @dvi.cnf /s3
```

and printed on a LaserJet with

```
▷ dvihplj mytext /oi+ /om2048 @dvi.cnf prn
```

For DeskJet users:

```
▷ dvihplj texfile /oi+ /od @dvi.cnf prn
```

A response file for a Star or Epson-compatible 9-pin printer:

```
/w=8in
% 12" tractor paper
/h=12in
% printer resolution x and y
/rx=240
/ry=216
/pl=c:\emtex\texfonts\fx_{0,h,1,2,3,4,5}
```

Commands for previewing and printing for Star printer users:

```
▷ dviscr mytext @dvi.cnf /s3
```

```
▷ dvidot fx80.dot mytext @dvi.cnf prn
```

The first argument, `fx80.dot`, is the parameter file required for Epson-compatible 9-pins dot-matrix printers.

6.1.4 Environment variables

Before scanning the command line, options will be read from environment variables; first from `DVIDRV`, next from an environment variable named after the DVI-driver (e.g., `DVISCR` or `DVIHPLJ`, e.g., given environment variables

```
▷ set DVIDRV=@dvi.cnf
```

```
▷ set DVISCR=/s3
```

```
▷ set DVIHPLJ=/oi+ /om2048
```

command lines for previewing and printing a file `mytext.dvi` can be simplified to

```
▷ dviscr mytext
```

```
▷ dvihplj mytext prn
```

Options in environment variables can be overruled by options on the command line, or in response files: if options conflict, the option processed last will override options processed earlier.

6.1.5 The \LaTeX response files

The examples above will work just fine if you set up your own system. However, the \LaTeX response files `*.cnf` are a lot more complex. The principal reason is that they must be shared by many users, who may not have the same directory-setup. They also support some options not described above. In this section, we shall explain the additional options and syntactical constructs used in these response files.

Here comes `δ:\emtex\data\hplj.cnf`, the \LaTeX response file for printing on a LaserJet II or III:

```
% hplj.cnf
/pd={,$TEXDVI:,$DVIDRVINPUT:}@i
/pl=$OWNFLI:ownfli
/pf={{$DVIDRVFONTS:,$EMTEXDIRSLASH:texfonts\laserjet\}@Rrdpi\,}@f.{pk,pxl}
/pg={,$DVIDRVGRAPH:}@PBf{@Ef,.msp,.pcx,.bmp}
/pv=${$VFFONTS:}@f.vf
```

```

/ps=hplj.sub
/r=300
/rf=300
/rg=300
/fs=1
/fd=2
/fb
/fl=-1

```

and now `fx.cnf`, the response file for Epson FX80-compatible printers:

```

% fx.cnf (240x216 DPI using FX fonts)
/pd={,$DVIDRVINPUT:}@i
/pl=$OWNFLI:ownfli
/pf=${DVIDRVFONTS:,$EMTEXDIRSLASH:texfonts\epsonfx\}@Rrdpi\@f{.pk,.pxl}
/pg={,$DVIDRVGRAPH:}@Rrdpi\,@PBf{@Ef,.msp,.pcx,.bmp}
/pv=${VFFONTS:}@f.vf
/ps=p6h.sub
/rx=240
/ry=216
/rfx=240
/rfy=216
/fs=1
/fd=2
/fb
/fl=-1

```

The following options were not described in the previous section and can be found in the documentation of the DVI-drivers:

- `/pmxxx` .tfm (T_EX Font Metric) files. If the DVI-driver can't find a font, a .tfm file for that font may help the DVI-driver to improvise and generate *some* output.
- `/pdxxx` DVI-file. This option can be used to let the DVI-driver search some additional directories.
- `/pfxxx` font files; for fonts which are not stored in libraries.
- `/pgxxx` graphic files.
- `/psxxx` font substitution file.
- `/pvxxx` path and filename format for virtual fonts.
- `/rfx#` horizontal font resolution; required if you want missing fonts to be automatically generated.
- `/rfy#` vertical font resolution.
- `/rgxxx` graphics resolution.
- `/fs#` font scaling.
- `/fmxxx` METAFONT mode. Note that the .cnf files do not specify /fm. \LaTeX will add this parameter dynamically in case automatic font generation is required (see Section 6.5.3).
- `/fd#` maximum drift.

Note also the use of environment variables. Here, they start with '\$' and end with ':', e.g., \$TEXDVI:. The DVI-driver will replace such an environment variable by its values plus a backslash. If the environment variable is not set, an empty string will be used. In this context, environment variables (or, more accurately, their values) may not contain the characters '{', '}' or '\$'. See the \LaTeX configuration file `texuser.<os>` in the `\delta:\emtex\btm` subdirectory or in your own work directory for a description of the environment variables. If you want to print a complete list of the environment variables used by \LaTeX , enter a DOS command

```
▷ set > prn
```

from the \LaTeX main menu.

Environment variables are not the only variables that can be used in the specification of a file or group of files. The \LaTeX response files also use:

@i is to be replaced by the `.dvi` file

@f is to be replaced by a fontname or graphics file name

@Pbf as @f, but without extension

@Rrdpi is to be replaced by the integer part of the font resolution in dpi, e.g., 300dpi, 329dpi or 360dpi

@Ef as @f, but here only the extension is retained (dot included)

As an example, consider the line

```
/pd={,$TEXDVI:,$DVIDRVINPUT:}@i
```

If `DVIxxx` had been called with `mytext [.dvi]` as input file, then `DVIxxx` would look for `mytext [.dvi]` in the current directory² and in the directories specified in the environment variables `TEXDVI` and `DVIDRVINPUT`.

6.1.6 Interactive operation of DviScr and DviScrs

The following are the more important keyboard commands when you are previewing with `DVISCR` and `DVISCRS`:

- Move around the page with the arrow keys.
- Pressing `C` (coarse) causes the arrow keys to make bigger 'steps'; `F` (fine) to make smaller 'steps'. Can be repeated.
- Page up and down: `PgUp` and `PgDn`.
- First and last page: `Ctrl PgUp` and `Ctrl PgDn`.
- Zoom in and out: `+` and `-`. Can be repeated.
- Go to page *n*: `G` or `P` plus *n* `Enter`.
- Quit: `Q` or `Ctrl C`.
- Search for text: `S`.
- Toggle the ruler display: `R`.
- Change the dimension unit forward: `U`. Dimension units are used in the status line and the ruler display.
- Change the dimension unit backward: `Ctrl U`.
- Reverse the image (black becomes white; white becomes black): `I`.

2. The comma preceding `$TEXDVI:` stands for an empty string.

6.1.7 Font substitution

Whenever the $\text{EMT}_{\text{E}}\text{X}$ DVI-drivers cannot find a font file they will consult a *font substitution* file. In this file fonts can be remapped to other fonts or other magnifications.

A font substitution file usually has two parts: one part that states which fonts are *always* substituted by others, another part that states what fonts are substituted only in case a font cannot be found.

As an example we show a font substitution file for the Laserjet III:

```
gray*      -> graylj*
MTEX *     -> mtex *
MTSY *     -> mtsy *
MTMI *     -> mtmi *

$b

pcti* *    -> plti* *
pcsl* *    -> plsl* *
pcr* *     -> plr* *

* 328      -> * 329
* *        -> * * /w
* *        -> rptmr * /r
```

The first lines tells DVIHPLJ to always substitute `graylj*` for `gray*` for any magnification.

`$b` means that the first part is finished.

The second part states that for all fonts of resolution 328 dpi the 329 dpi fonts can be used. In case no font file at all is available, but the font metric file is, then all characters will be left blank (the `/w` parameter). I even no font metric file is available, the font file `rptmr` (Times Roman) will be used, but all characters will be printed as black rectangles (the `/r` parameter).

6.2 Using PostScript and DviPS

For professional typesetting, PostScript has become a *de facto* standard. Any serious DTP program, wordprocessor or illustration package can generate PostScript output, from which any print shop can generate high-resolution output.³

Even if you plan to have your files professionally typeset,⁴ you should make proofs on a local printer. If you do not have a PostScript printer, you can use GHOSTSCRIPT, a public domain PostScript interpreter, to print your PostScript files on an arbitrary printer. GHOSTSCRIPT can also ‘print’ to the screen (see Subsections 6.2.6 and 6.2.5).

3. This is the theory; in practice, there may be problems with fonts, or your file may be more complex than the target typesetter can handle, or hardware and software don’t support exactly the same version of PostScript.

4. Note that for generating output for a high-resolution typesetter, high-resolution bitmapped fonts and high-resolution bitmapped graphics are required. \LaTeX provides only a small set of ready-made bitmapped fonts in resolutions higher than 600dpi; you may have to generate your own, using METAFONT or PS2PK. See Section 6.2.2 for details on the use of scalable PostScript fonts.

PostScript support for T_EX is provided by DVIPS, written by T. Rokicki. This program generates PostScript output from a .dvi file, and offers mechanisms to include PostScript graphics in a T_EX document and to use scalable PostScript fonts.

We shall first explain how you can make use of PostScript graphics and PostScript fonts in your documents; then we describe the command line interface and configuration of DVIPS.

6.2.1 Including PostScript graphics

Encapsulated PostScript (file type .eps) is a restricted PostScript format suitable for inclusion in other PostScript files, and is widely used as a graphics file format. Many word-processing and DTP programs support .eps graphics for a PostScript target printer simply by passing the code on to the printer, without trying to interpret it themselves. DVIPS basically follows the same approach. DVISCR won't display a .eps graphic; but you can run DVIPS and then use GHOSTSCRIPT or GSVIEW for previewing the PostScript file. As can be seen in Section 6.6, GHOSTSCRIPT can be selected as a viewer type within L^AT_EX.

If you want to include a .eps graphic, you'll need a macro package. Section 7.4.5 describes how to use the style files psfig and figures. An alternative is epsf, which is from the same author as DVIPS. L^AT_EX users can include epsf as a 'package', e.g.,

```
\documentclass{report}
\usepackage{epsf}
```

In general T_EX users can place a statement

```
\input epsf
```

somewhere early in their T_EX file.

To include a graphic file picture.eps at its 'natural' size, at the current position, use a statement⁵

```
\epsfbox{picture.eps}
```

You can also scale the picture:

```
\epsfxsize=2in\epsfysize=1.5in\epsfbox{picture.eps}
```

\epsfxsize sets the width, \epsfysize the height. If you specify only one dimension, then the other dimension will be computed automatically, preserving the original aspect ratio.

The DVIPS manual describes additional scaling options. Other topics not covered here include PostScript header files and color support.

L^AT_EX users are advised to use the GRAPHICS or the GRAPHICX package to include any kind of graphic picture. Here is some sample code:

```
\documentclass{article}
\usepackage{graphicx}
```

5. According to the DVIPS manual, the epsfbox macro typesets the figure as a vbox; the manual also says that L^AT_EX users may first have to enter a command \leavevmode in order to make certain environments work correctly.

```

\begin{document}

\includegraphics[width=50mm]{mypict.eps}

\end{document}

```

See the documentation on these packages for more details.

6.2.2 PostScript fonts

Another attraction of PostScript printers is that it gives you access to Adobe Type 1 scalable fonts. These fonts can be scaled to any size: different sizes don't need to be loaded separately.⁶ Mostly, you'll make use of the set of 35 internal scalable fonts,⁷ but there is also a huge collection available of external fonts, both commercial and public domain that can be used if you want to give your documents a more individual touch. You can use these fonts side-by-side with the bitmapped T_EX font libraries.

L^AT_EX users have a very easy job of accessing internal PostScript fonts in their documents: they only need to specify the appropriate font package. E.g., specifying:

```

\documentclass{report}
\usepackage[times]{psnfss}

```

In this case it can be written even simpler:

```

\documentclass{report}
\usepackage{times}

```

This will cause your document to be typeset in Times fonts. Note that text in math mode won't be affected by this package. Other ready-made styles for PostScript fonts include `palatino` for Palatino and many others; browse the `\emtex\latex\psnfss` directory for other font options. Another interesting option is `mathptm`. It works like the Times options, but it also attempts to replace all Computer Modern fonts used in mathematics by Times and other PostScript fonts.

If you use plain T_EX, or if you want to use external fonts, you'll have define fonts yourself. First, consult the file `c:\texfiles\4system\psfonts.map` for a list of available fonts. A typical entry for a resident font may look like

```

* DVIPS style:
rpagk AvantGarde-Book
* PSNFSS style:
pagk8r AvantGarde-Book "TeXBase1Encoding ReEncodeFont " <8r.enc

```

6. Scaling bitmaps produces 'jaggies'; scaling an outline preserves sharpness. However, traditionally type designers designed each size of a font separately, smaller sizes tending to be more squat than larger sizes. Designing a font for use at any size inevitably involves compromises. Knuth's METAFONT program uses a far more complex font description, which produces different shapes at different sizes. As a consequence, font generation by METAFONT is a rather slow process; it would probably be too slow for the on-the-fly font generation done inside PostScript printers.

7. These include eight complete families of four weights each: Courier; the serified fonts ITC Bookman, New Century Schoolbook, Palatino and Times Roman; the sans-serif fonts ITC AvantGarde, Helvetica and Helvetica Narrow; and single weights of three more fonts: a script font Zapf Chancery, a symbol font, and Zapf Dingbats, which is a 'font' consisting of typographic symbols and ornaments such as bullets, check marks and scissors.

AvantGarde-Book indicates the normal non-italic non-bold ‘weight’ of ITC AvantGarde. Note the initial `r` in `rpagk`, which stands for ‘raw’ as opposed to ‘cooked’. These terms will be explained later on. An entry for a non-resident or external font named Optima might look like

```
rpopr Optima <popr.pfb
```

Note that there is a *single* space between the font name and the ‘<’ sign. There should be *no* spaces between the ‘<’ sign and the font *file* name.

In your \TeX file you make AvantGarde-Book available at a point size of, say, 20pt, with a line

```
\font\avantgarde=pagk8r at 20pt
```

Note that here the initial `r` is not used. This line will allow you to use AvantGarde at 20 points with, e.g.,

```
{\avantgarde This line is typeset in AvantGarde at 20 points}
```

Similarly for Optima; here you shouldn’t use an initial `r` either.

The low-down on PostScript fonts

The use of PostScript fonts requires three or four pieces of information, depending on whether it is an internal or an external font:

- For an external font, the font description itself. In the second example, this would be the file `popr.pfb` from the Optima entry in `psfonts.map`. External fonts are stored as outlines called ‘Postscript Font Binary’ (PFB) or ‘Postscript Font Ascii’ (PFA). PFB’s and PFA’s contain identical information in binary (ASCII 0–255) or ASCII (ASCII 32–127) format. PFB can be translated into PFA and vice versa by PostScript utility programs. DVIPS ensures that these files are downloaded to the printer,⁸ which will scale the fonts as needed.
- The font metrics. This would be the file `rpagk.tfm` for ITC Avant Garde and `rpopr.tfm` for the external Optima font. Note the initial `r`. This file is a direct conversion of the original `.afm` font metric file, which came with the font. With these font metrics, the font is considered ‘raw’. Conversion from `.afm` to `.tfm` can be performed by two program: `FONTINST` and `AFM2TFM`. The first is a \TeX program that usually gives better results than the latter. `AFM2TFM` runs much faster and is suitable for a quick and almost perfect installation. See Section 4.13 and 4.14 for more details.
- The catch: a mapping between the Adobe and the \TeX character encoding. In the above examples, these are the files `pagk.vf` and `popr.vf` (`vf` for *virtual font*). Without this mapping, ordinary characters would come out as expected, but special characters such as ligatures and accented letters might not.
- The font metric file of the remapped font: `pagk.tfm` and `popr.tfm` respectively. This file will be read by the \TeX compiler. With this metric and the above remapping, the font is considered ‘cooked’ and digestible.

The configuration file `config.ps` (see Section 6.2.4) tells DVIPS where it can find all these files.

8. The `<popr.pfb` in the Optima entry in `psfonts.map` directs DVIPS to include the font outline file `popr.pfb` as a header file.

If you have Type 1 fonts of your own that you want to use in a \TeX document, the on-line documentation of DVIPS will tell you how to generate the required .tfm and .vf files. The documentation can be found in the directory $\delta:\backslash\text{emtex}\backslash\text{doc}\backslash\text{programs}$.

6.2.3 DviPS command line

The syntax of DVIPS is

```
DVIPS32 [options] <input file>
```

Note that this 32-bits program only runs on computers equipped with a 80386 CPU or higher.

Calling the program without options or parameters produces a brief summary of its syntax.

The principal options are

- o *x* output to file or device *x*; default is *<input file>.ps* for *<input file>.dvi*
- D # resolution; this information is required for bitmapped fonts, e.g., the Computer Modern family, and for bitmapped graphics.
- t *s* paper format *s*
- M No automatic font generation
- p # start ‘printing’ at page #
- l # stop ‘printing’ after page #
- a Conserve memory. Useful on PC’s with a limited amount of memory, usually not needed when using DVIPS32.

Here, ‘#’ denotes a number; it should be separated by a space from the option. Example:

```
▷ dvips32 -D 300 -t a4 -M mytext
```

reads *texttext.dvi* and writes *mytext.ps* which can be sent to a PostScript printer. `-D 300` specifies that the target printer has a 300 dpi resolution; `-t a4` specifies a paper size A4; `-M` prevents automatic generation of missing fonts.

Note that the syntax does not conform to $\text{EM}\TeX$ conventions.

Options can also be entered via the environment variable `POST_OPTION`. $\mathcal{A}\TeX$ uses

```
▷ set POST_OPTION=-D 300 -t a4 -M
```

See the example above for an explanation of the parameters. With this environment variable set, the example above can be simplified to

```
▷ dvips32 mytext
```

$\mathcal{A}\TeX$ only supports the 32 bits version of DVIPS. Actually, there are two versions of DVIPS: *dvips32.exe* and *dvihps32.exe*. The program *DVIHPS32* is an extension of DVIPS that can handle ‘HyperDVI’-files, generating ‘HyperPS’ output. See Section 4.2, DVIPS documentation and ‘hyperref’ documentation for more details. This version supports the extra option `-z` to run in ‘hyper’ mode. Because otherwise these two programs are identical, in $\mathcal{A}\TeX$ *dvihps32.exe* is renamed to *dvips32.exe* and is used for all purposes.

6.2.4 DviPS configuration

Most configuration is done in *config.ps* in the PS subdirectory of the \TeX directory. This file contains several paths, among other things. If DVIPS appears unable to find fonts, the real reason may be that it can’t find this configuration file. Try setting the environment variable `TEXCONFIG` to the directory of

config.ps :

▷ set TEXCONFIG=δ:\emtex\ps

You can also use TEXCONFIG to direct DVIPS to a version of config.ps of your own. See the on-line documentation for a description of config.ps and for the full story on configuring DVIPS. Watch the syntax of config.ps carefully: it differs from EM_TE_X conventions. \mathcal{A} T_EX uses the file config.ps for all the general PostScript settings and certain .cfg files for additional printer type specific settings.

6.2.5 Ghostscript as printer and viewer

The syntax of GHOSTSCRIPT is as follows:

```
gs[386] [-sDEVICE=devicename] [switches] [file1.ps file2.ps ...]
```

The program gs386.exe only runs on computers equipped with a 80386 CPU or higher, the program gs.exe runs on any computer.

Available devicenames are:

vga	ega	svga16	atiw	tseng	tvga	deskjet	djet500
laserjet	ljetplus	ljet2p	ljet3	ljet4	cdeskjet	cdjcolor	cdjmono
cdj550	pj	pjxl	pjxl300	epson	eps9high	ibmpro	bj10e
bj200	bjc600	bjc800	pcxmono	pcxgray	pcx16	pcx256	pcx24b
pcxcmk	tiffcrl	tiffg3	tiffg32d	tiffg4	tiffzw	tiffpack	bmpmono
bmp16	bmp256	bmp16m	tiff12nc	tiff24nc	psmono	psgray	bit
bitrgb	bitcmk	jpeg	jpeggray	pdfwrite	nullpage		

Most frequently used switches are: (you can use '#' in stead of '=')

-d<name>[=<token>]	define name as token, or true if no token given
-dNOPAUSE	don't pause between pages
-g<width>x<height>	set width and height ('geometry'), in pixels
-q	'quiet' mode, suppress most messages
-r<res>	set resolution, in pixels per inch
-s<name>=<string>	define name as string
-sDEVICE=<devicename>	select initial device
-sOutputFile=<file>	select output file: embed %d or %ld for page #, - means stdout, use command to pipe
-	read from (redirected) stdin non-interactively

For more information, see the file δ:\emtex\utils\gstools\gs\use.doc.

To print on a laserjet printer the file book.ps one can use the command:

```
gs[386] -sDEVICE=ljet3 -dNOPAUSE -sOUTPUTFILE=LPT1 4texdoc.ps quit.ps
```

6.2.6 Ghostscript as previewer

GHOSTSCRIPT can be used as previewer under MS-DOS or under MS-WINDOWS 95 or NT. In the latter case \mathcal{A} T_EX uses the program 32 bits Windows program GSVIEW.

When using GHOSTSCRIPT as previewer under MS-DOS \mathcal{A} T_EX will load an extra PostScript program called ps_view.ps, written by written by P. Pianowski and B. Jackowski, and extended by the authors of \mathcal{A} T_EX. This file will enables several keys that make it easy to browse through a PostScript file.

The following keys can be used:

<code>Home</code>	goto first page	<code>F7</code>	zoom out
<code>End</code>	goto last page	<code>F8</code>	zoom in
<code>PgDn</code>	goto next page	<code>F10</code>	redraw current picture
<code>PgUp</code>	goto previous page	<code>F12</code>	view whole picture
<code>←</code>	shift left	<code>Q</code> <code>Enter</code>	draw grid
<code>→</code>	shift right	<code>V</code> <code>Enter</code>	view picture
<code>↑</code>	shift up	<code>V</code> <code>f</code> <code>Enter</code>	flip vertically
<code>↓</code>	shift down	<code>H</code> <code>f</code> <code>Enter</code>	flip horizontally
<code>Q</code> <code>Enter</code>	quit	<code>F</code> <code>r</code> <code>Enter</code>	show frame
<code>Esc</code>	quit	<code>Del</code>	undo previous picture
<code>Alt</code> <code>X</code>	quit	<code>Ins</code>	redo undone picture
<code>F1</code>	display predefined keys (press <code>x</code> <code>Enter</code> to return to your document)		

You can set the *magnification factor* by typing a number and then pressing `M` and `Enter`

You can go to a specified page by typing the number of that page and then pressing `F4`

You can set the shift factor by typing a number and then pressing `S` and `Enter`

You can shift the picture by typing the horizontal shift, a space, the vertical shift and then pressing `Tab`

You can rotate a page by typing the rotating angle ($\text{abs}(\text{STEP}) \cdot 90$) and then typing `R` `O` and `Enter`.

Note: These keys only work correctly if the the PostScript file contains the standard page markup comments.

When running \LaTeX on WINDOWS 95 or NT \LaTeX can start GSVIEW, a true 32 bits Windows program, as a previewer. This program explains itself quite well, so we will not discuss it here in detail. Note, however, that unlike any other previewer, `PgUp` and `PgDn` do not work as you would expect. Click on the `-` and `+` Windows buttons instead.

6.3 DviDjC command line

The syntax of the DVIDJC programs `dvi500c.exe` and `dvi550c.exe` is as follows:

```
DVI5x0c.exe [ options ] dvifile
```

They support the following options:

- s<starting page> (default: 1 i.e. first in file)
- c<number of pages> (default: 9999 i.e. all)
- m<new magnification> (default: from dvi file)
- h<horizontal offset> (default: 78 (pixels))
- v<vertical offset> (default: 156 (pixels))
- f (default: /usr/TeX/lib/tex/fonts/)
- t<TFMfile directory> (default: /usr/TeX/lib/tex/fonts/)
- o<outfile> (default: prn)

Note: These DVI-drivers do *not* support subdirectory searching. Another restriction is that the font and tfm directory parameters can only take *one* path. Therefore \LaTeX calls the programs using the subdirectory `\emtex\tfmall` for tfm files and `\emtex\texfonts\deskjet` for font files. In case you need other fonts you will have to put them in the current directory, where the DVI-drivers will also scan for tfm and font files.

6.3.1 Interactive operation of DviVGAc

DVIVGAC is a screen previewer dedicated to dvi files generated for printing on color Deskjet series 500 printers. This dvi viewer is capable of showing the colors as they will appear when printed on paper.

The program supports the following keys:

P or **Backspace**: previous page

N or **Space**: next page

C: use colors (much slower)

G: use grey shades (faster)

Q: quit

The viewer supports a ‘moving magnifying glass’. It is activated by pressing one of the following keys:

Arrow keys: move the magnifying glass left, right, up or down

Home: move to the upper left corner of the page

End: move to the lower right corner

Note: Whenever you try to go to the next or previous page and your current position is either at the last or the first page the program will abort.

Note: If the viewer can’t find a font file it will display boxes for all character in that font, using its tfm file.

6.4 DviWin as previewer

DVIWIN is a true MS-WINDOWS program. It comes in two versions, a 16 bits program called DVIWIN.EXE and a 32 bits program called dviwin2.exe. Naturally \LaTeX uses the latter. The installation script installs a file dviwin2.ini in your Windows system directory.

The settings in this ini file should be sufficient for using DVIWIN in \LaTeX . By default DVIWIN will use bitmap fonts generated for printers of type HP Laserjet II, III or 4L (300 dpi). For technical reasons automatic font generation is supported only when running on MS-WINDOWS 95, not NT.

One of the problems when running DVIWIN is that it cannot handle virtual fonts. This means that the if the .dvi file references virtual fonts the .dvi file has to be devirtualized first. Not devirtualizing the output will result in missing fonts, that cannot be generated automatically. See Section 4.2 for details on devirtualizing in \LaTeX .

For fine-tuning DVIWIN you should read the detailed Windows help file on DVIWIN.

6.5 Available fonts

\LaTeX stores all standard \TeX fonts in directories named after the device they were generated for beneath the directory $\delta:\backslash\text{emtex}\backslash\text{texfonts}$. E.g., fonts generated for the HP LaserJet are stored in $\delta:\backslash\text{emtex}\backslash\text{texfonts}\backslash\text{laserjet}$. The bitmapped fonts are in ‘pk’-format and stored in subdirectories according to their resolution. E.g., a 10 points Computer Modern Roman font for a HP LaserJet (300 dpi) will be stored as $\delta:\backslash\text{emtex}\backslash\text{texfonts}\backslash\text{laserjet}\backslash\text{300dpi}\backslash\text{cmr10.pk}$.

However, fonts are also stored in font libraries. The advantage of font libraries are:

1. Font libraries take up less disk space than all the stored .pk files together. The difference depends on the cluster size of your hard disk. The larger the the cluster size, the greater the advantage of libraries.
2. Font libraries are easier to check for errors than individual fonts (see below).
3. Font libraries can store fonts with file names longer than 8 characters.

Disadvantages are:

1. Not all DVI-drivers can handle font libraries: only the EM \TeX drivers and DVIPS can.
2. If an error occurs in a font library it may be very hard to extract all other fonts from it.
3. The size of the fonts you add must be given very exactly, otherwise EM \TeX DVI-drivers may not be able to use it. Directory names (such as laserjet\329dpi) where fonts are to be found are automatically rounded, so you will not need a directory called 464.49887dpi simply because MS-DOS doesn't allow such names. 465dpi will do.
4. The .pk fonts are operating system independent. On other operating system you may not find tools for managing font libraries so the font files cannot be, e.g., copied from CD-rom to a UNIX system and unpacked.

Font libraries can be browsed and edited by the fontlib.exe program. FONTLIB has many parameters that are all documented in fontlib.doc. The most important FONTLIB commands are:

list all fonts in a font library:

▷ fontlib /l fontlib.fli

add or replace a font:

▷ fontlib fontlib.fli 300 +fontname

remove a font:

▷ fontlib fontlib.fli 300 -fontname

extract a font:

▷ fontlib fontlib.fli 300 #fontname

test the integrity of a font library:

▷ fontlib /t fontlib.fli

'300' is an example of size relative to the printer's resolution, which is 300 dpi for HP LaserJets and compatibles. This means that

▷ fontlib fontlib.fli 600 #fontname

would extract the font sized twice as big, e.g., cmr10 at 20 points.

6.5.1 Commercial fonts

On the commercial market there are many fonts available.

Whatever commercial and/or public domain PostScript font you have, with the help of the program PS2PK you can generate the PK files for every type of printer. The PostScript fonts (.pfb files) should be stored in the directory $\delta:\text{emtex}\backslash\text{ps}\backslash\text{fonts}$ (i.e. the environment variable PSFONTS). See next section for details on PostScript fonts.

\TeX is famous for the excellence of its typesetting and especially the high quality of its mathematics. Most fonts families, however, do not have a special mathematical font and therefore mathematics is often set in Computer Modern. Very popular commercial PostScript fonts that have a mathematical font are the MathTimes fonts for Times (from \TeX plorators Corporation) and the Lucida Bright fonts (from Bigelow & Holmes Inc.).

6.5.2 Using PostScript fonts

If you have access to a PostScript printer, you can use the DVIPS program to print a dvi-file, and use any internal or external PostScript font. You can still use font libraries for any font you do not have in PostScript outline format, as DVIPS can read font libraries. But mostly you will use internal fonts such as *Times-Roman* or *Palatino*. External fonts are usually stored as font outlines called ‘Postscript Font Binary’ (PFB), also called PostScript Type1. An alternative is ‘Postscript Font Ascii’ (PFA), also called PostScript Type3. PFBs and PFAs contain identical information in binary (ASCII 0–255) or ASCII (ASCII 32–127) format. PFB can be translated into PFA and vice versa by PostScript utility programs. DVIPS reads the file `psfonts.map` from `c:\texfiles\4system` to find out which font file to include when printing a document using external fonts. An important feature of PostScript fonts is scalability. PostScript fonts can be scaled to literally *any* size. No more than one font needs to be loaded to achieve this, in contrast to bitmapped fonts such as PK files. For every size you need to load another PK file. Besides, PK files are specific for a given printer resolution—PostScript fonts are not. PostScript fonts can be downloaded partially by using a program called

Choosing different fonts can be done by means of loading packages in L^AT_EX or ‘manually’ in plain T_EX. E.g., to typeset a document in PostScript Times-Roman you specify `times` as a package to be loaded. Note that the document can be printed on a PostScript printer at any resolution, but it cannot be printed on a LaserJet printer, unless you have bitmapped fonts available. 4T_EX supports all 35 standard PostScript fonts, thanks to the URW fonts supplied with GHOSTSCRIPT.

6.5.3 Automatic font generation

METAFONT is a program for designing fonts for use by T_EX. It interprets a drawing language with a syntax that slightly resembles the Pascal programming language. The input needed for METAFONT can be interactive, or from a source file. METAFONT source files have the extension `.mf`. The output of METAFONT is a GF (‘generic font’) file, extension `.gf`, which contains the bitmap. This may be compressed to a PK (‘packed’ font) file, extension `.pk`, using the program GFTOPK.

When compiling your document, T_EX does not use the bitmap fonts but only the TFM files (‘T_EX Font Metrics’). The TFM files describe the dimensions, ligatures and kerns of the font. METAFONT can make a TFM file as well as a bitmap font file.

A more detailed description of METAFONT can be found in the book ‘The METAFONT book’ (Knuth (1986)).

When using the previewer or printer drivers, EMT_EX will look for correct bitmap fonts in the font libraries (specified in the driver response file) or somewhere on a specified path given by the environment variable `DVIDRVFONTS`. If a font can not be found, a substitution table (e.g., `hplj.sub`) is used (see Section 6.1.7).

4T_EX can generate fonts on demand if you have set up your directories correctly and specified `FONTGEN=y` in your `texuser.<os>`. Many DVI-drivers will pause when one or more fonts are missing, asking you if you want to generate these fonts now. You can get a list of missing fonts by entering , enter to skip font generation, or enter to start font generation.

Naturally, font generation is only possible if font rules are available. 4T_EX supports two types of font rules: METAFONT (`.mf`) and PostScript (`.pfb`). METAFONT must be installed in the `δ:\emtex\metafont` directory; PostScript font files must be installed in directory `δ:\emtex\ps\fonts`, along with the file `psfonts.inf` that defines which fonts are available and how `.pk` files are to be generated from them, using the font rasterizer PS2PK. You can also install your own PostScript font in a directory set by the environment variable `MYPSPFONTS` in `texuser.<os>`. Likewise, you can also install your own

METAFONT sources in a directory set by the environment variable MYMF in `texuser.<os>`. Note that PS2PK will only run on computers equipped with a 80386 cpu or higher.

Bitmapped font files (.pk) will be stored on the directory specified by the environment variable MYFONTS in `texuser.<os>`, e.g., `MYFONTS=c:\texfiles\fonts`. In order to make sure the DVI-drivers use the correct fonts for any printer selected, \TeX puts font files in a subdirectory named after the current METAFONT printer mode, e.g., `c:\texfiles\fonts\laserjet`. \TeX uses the environment variable PRINTER_MODE set in the printer definition file to store fonts in the subdirectory of MYFONTS. (e.g., the printer definitions file `\emtex\prndest\hplj.prt` will set `PRINTER_MODE=LASERJET` and the fonts are stored in `c:\texfiles\fonts\laserjet`) In this directory, font files are stored in subdirectories named after the resolution of the fonts. For instance, if you need the font CMR10 at 20 points for a LaserJet III, \TeX will generate `c:\texfiles\fonts\laserjet\600dpi\cmr10.pk`, because the font is twice as big as the standard 300 dpi resolution. Note that this font is *not* the same as `c:\texfiles\fonts\ljjiv\600dpi\cmr10.pk`. The latter would be a 10 points font for a 600 dpi LaserJet 4.

6.6 Supported output devices

- Apple Image Writer matrix (160 × 144dpi)
- Adobe PDF (300dpi) through GHOSTSCRIPT
- Adobe PDF (600dpi) through GHOSTSCRIPT
- Canon Bubblejet 10e (360dpi)
- Canon Bubblejet 300 (360dpi)
- Canon Bubblejet 10e (360dpi) through GHOSTSCRIPT
- C.ITOH matrix (160 × 144dpi)
- C.ITOH matrix (80 × 72dpi)
- C.ITOH matrix (80 × 72dpi) using Laserjet 300dpi fonts
- C.ITOH matrix (160 × 144dpi) using Laserjet 300dpi fonts
- DVI to ASCII
- DVIwin 2.9 (uses HP Laserjet 300dpi)
- Epson FX 80 or RX matrix (240 × 216dpi)
- Epson FX 100 or RX matrix (240 × 216dpi)
- Epson FX 100 or RX matrix (120 × 72dpi)
- Epson FX 80 or RX matrix (120 × 72dpi)
- Epson LQ matrix / NEC P6 (360 × 360dpi)
- Epson LQ matrix / NEC P6 (180 × 180dpi)
- Epson LQ matrix (360 × 180dpi)
- Epson LQ matrix (wide) (360 × 360dpi)
- Epson LQ matrix (wide) (180 × 180dpi)
- Epson LQ matrix (wide) (360 × 180dpi)
- Epson FX matrix (240 × 216dpi) through GHOSTSCRIPT
- Epson Stylus 800 matrix (360 × 360dpi)
- FAX PCX bitmap format (204dpi)
- HP Deskjet (300dpi)
- HP Deskjet 1200c (300dpi)
- HP Deskjet 1200c color (300dpi) through GHOSTSCRIPT
- HP Deskjet 1200c/ps PostScript (300dpi)
- HP Deskjet 500c color (300dpi)
- HP Deskjet 550c color (300dpi) through GHOSTSCRIPT

- HP Deskjet 550c monochrome (300dpi) through GHOSTSCRIPT
- HP Deskjet (300dpi) through GHOSTSCRIPT
- HP Laserjet (300dpi) using Laserjet 600dpi fonts
- HP Laserjet II, III, 4L (300dpi)
- HP Laserjet IIp (300dpi) through GHOSTSCRIPT
- HP Laserjet III (300dpi) through GHOSTSCRIPT
- HP Laserjet 4, 5 or 6 (600dpi)
- HP Laserjet 4, 5 or 6 PostScript (600dpi)
- HP Laserjet 4, 5 or 6 (600dpi) through GHOSTSCRIPT
- HP Laserjet Plus (300dpi) through GHOSTSCRIPT
- HP Laserjet II, III, 4L PostScript (300dpi)
- HP Paintjet (180 × 180dpi)
- HP Paintjet XL color (300dpi) through GHOSTSCRIPT
- HP Paintjet color (180dpi) through GHOSTSCRIPT
- HP Quietjet (192 × 192dpi)
- HP Thinkjet (192 × 96dpi)
- IBM Proprinter 4201 matrix (240 × 216dpi)
- IBM Proprinter 4202 matrix (240 × 216dpi)
- IBM Proprinter 4207 matrix (180 × 180dpi)
- IBM Proprinter 4207 matrix (360 × 180dpi)
- IBM Proprinter 4208 matrix (180 × 180dpi)
- IBM Proprinter 4208 matrix (360 × 180dpi)
- Kyocera Laserjet (300dpi)
- Lexmark Optra S 2450 PCL5 (1200dpi)
- Lexmark Optra S 2450 PostScript (1200dpi)
- Lexmark Optra S 2450 PCL5 (600dpi)
- Lexmark Optra S 2450 PostScript (600dpi)
- Linotronic PostScript (2540dpi)
- Linotronic PostScript (1270dpi)
- MSP old bitmap format (300dpi)
- MSP newer bitmap format (300dpi)
- NEC P6 or Panasonic KX-P1124 matrix (360 × 360dpi)
- NEC P6 or Panasonic KX-P1124 matrix (180 × 180dpi)
- NEC P6 matrix (180 × 180dpi) using 360dpi fonts
- NEC P6 or Panasonic KX-P1124 matrix (360 × 180dpi)
- NEC P6 matrix (360 × 180dpi) using 360dpi fonts
- NEC P7 matrix (360 × 360dpi)
- NEC P7 matrix (180 × 180dpi)
- NEC P7 matrix (180 × 180dpi) using 360dpi fonts
- NEC P7 matrix (360 × 180dpi)
- NEC P7 matrix (360 × 180dpi) using 360dpi fonts
- Océ PostScript (508dpi)
- PCX bitmap format (300dpi)
- Tandy DMP-130 matrix (240 × 216dpi)
- VGA viewer (110dpi)
- Xerox Docutech PostScript (600dpi)

Most of the listed printers can also be selected as viewers. If you want to have full profit of your super VGA screen (i.e. the 800 × 600 or 1280 × 1024 pixels graphics mode) see Subsection 6.1.2 how to set the environment variable `SCRTYPE` and `VIDEO` in the file `texuser.<os>`.

Part 7

Utility programs

7.1 Bibliographies and indexes

7.1.1 BibTeX

When writing an article, book or report you often refer to other literature. At the end of the document you usually include a bibliography. With L^AT_EX and the program BIB_TE_X this becomes easy.

Using one or more bibliographic databases that contain references to all books, articles, etc. you never have to create a bibliography yourself. By simply including, e.g., the command

```
\bibliography{articles,books,reports}
```

in the document, you tell L^AT_EX to use the databases `articles.bib`, `books.bib`, and `reports.bib` to select the documents you referred to in your document and put them in a bibliography. Note that the databases must have the extension `.bib`.

Referring to other documents is done by one of the following commands: `\cite{KEYNAME}` or `\nocite{KEYNAME}`, where `KEYNAME` is the key or identification name you attached to the reference in the databases. The difference between `\cite` and `\nocite` is that `\cite` produces some output on the spot where the command is issued (e.g., the number of the document in the bibliography). The command `\nocite` produces no output and only includes the reference in the bibliography. Economists do not like to use numbers when making references but like to use the year of publication instead, and therefore often use the `\nocite` command, e.g.,

```
Knuth~(1984)\nocite{KNUTH}
```

will produce the output `Knuth (1984)` and will display the book in the bibliography. Mathematicians like to use numbers for references and they use the `\cite` command. E.g., if you type `Knuth\cite{KNUTH}`, the text `Knuth [10]` will be produced and the book will be listed as the 10th entry of the bibliography.

The layout of the bibliography is defined by `\bibliographystyle{BIBSTYLE}`, where `BIBSTYLE` is the bibliography style file with the extension `.bst`. The standard bibliography style files are `plain.bst`, `alpha.bst`, `abbrev.bst`, and `unsrt.bst`, but many publishers have defined their own bibliography layout and their own `.bst`. L^AT_EX comes with a lot of `.bst` files. Most probably one of those will satisfy your requirements.

To produce a bibliography file you have to run L^AT_EX with your document (e.g., the file `sample.tex`), which produces one or more auxiliary files (files with the extension `.aux`). Compiling the file `sample.tex` will always produce the auxiliary file `sample.aux` but when other files are included these will also produce an auxiliary file. The auxiliary files contain the cross-referencing information, e.g., the references

to mathematical formulas, sections etc. and also the reference to the `\cite` and `\nocite` commands and the arguments of the `\bibliography` and the `\bibliographystyle` commands.

After running \LaTeX for the first time you run `BIBTEX`, e.g.,

```
▷ δ:\emtex\utils\bibtex sample
```

`BIBTEX` will read the auxiliary files and will query the databases for the key or identification name of the `\cite` and `\nocite` commands. When the keys are found it will produce a file `sample.bbl` containing the \TeX commands to produce a bibliography according to the specified style file.

The second time you run \LaTeX with your `sample.tex` document the `\bibliography` command reads the `sample.bbl` file and produces a bibliography. The bibliography is created on the location where the `\bibliography` command is issued. This means that this command should be at the end of the document, although the `\bibliography` command can be issued anywhere after the `\begin{document}` command.¹

A detailed discussion how to use `BIBTEX` can be found in Lamport (1986), i.e. Section 4.3. and appendix B. We also refer to Patashnik (1988) for a discussion of how to make your own bibliography style. The following example illustrates how Knuth's \TeX book is entered in the `BIBTEX` database:

```
@BOOK{KNUTH,
  author={Knuth, D.E.},
  title={The {\TeX}{book}},
  year=1984,
  publisher={Addison-Wesley},
  address={Reading, Massachusetts}
}
```

There are many options and details to be aware of when building a bibliographic database. Appendix B of Lamport's reference manual explains how to do this, but there is a much simpler solution: `BIBDB`, the interactive `BIBTEX` bibliography database manager (see Section 7.1.2).

\LaTeX uses the big `BIBTEX` version for 80386 CPUs and higher (`bibtex32.exe`). The `BIBTEX` program can be found in the `δ:\emtex\utils` directory and the bibliography style files can be found in the directory `δ:\emtex\inputs\bibtex`. In the example given above, `BIBTEX` will look for the file `sample.aux` and the `.aux` files of all the included files in the current directory. The bibliography style file is read from the current directory, or, if not found, in the directory specified by the environment variable `BIBSTYDIR`. The database is also read from the current directory, or from the directory specified by `TEXBIB` (set in `texuser.<os>`) if the file was not found.

We will end this section with an example of a document that is set up to produce a bibliography. Note that \LaTeX has a `BibTEX` utility for maintaining and running `BibTEX` (see Section 4.3).

```
\documentclass{article}
...           % put here your macros and definitions

\begin{document}
...           % some text
```

As explained by Knuth~(1984)\nocite{KNUTH} in his well-known book

1. You have to run \LaTeX at least twice to get a bibliography typeset. This is time consuming but the only way to see which references should be included in the document. Once the `.bbl` file exists you can edit this file for corrections or add/remove some references. This saves you from editing the databases and running \LaTeX and `BIBTEX` again but you should remember that changes in the `.bbl` file do not change the citations (e.g., the numbering) and will be defeated as soon as you run `BIBTEX` again.

```

...           % more text and references (\CITE and \NOCITE)

\bibliographystyle{plain}
% plain.bst is the style file for the bibliography layout

\bibliography{articles,books,reports}
% articles.bib, books.bib, and reports.bib are the bibliography
% databases in which your references can be found.

\end{document}

```

7.1.2 BibDB

BIBDB is a program written by E. Doron for managing BIB_TE_X databases. BIB_TE_X is a good way of keeping a reference library, but managing it is a pain. Finding entries using criteria such as year or keywords, and so forth, is cumbersome, and entering data is fraught with errors. Both an MS-DOS version and an MS-WINDOWS version are used in \LaTeX .

With BIBDB you can browse through the database, extract selected entries, add and edit entries in a convenient manner. The user-friendly interface shields you from many technical details that would otherwise often cause errors.

The programs can be found in the $\delta:\text{emtex}\backslash\text{utils}$ directory and $\delta:\text{emtex}\backslash\text{utils}\backslash\text{win}$. The syntax is:

$[W]BIBDB$ [*options*] [*bibfile1* [... *bibfile8*]]. Of course *wbibdb* is the MS-WINDOWS version.

The BIB_TE_X database name should be supplied without the *.bib* extension. If the bibliography file does not include a path, BIBDB searches for the bibliography in the current directory and then in the directory specified by the environment variable *TEXINPUT*. When BIBDB is run it will look for the *bibdb.cfg* configuration file. It will first look in the current directory, then in the $\delta:\text{emtex}\backslash\text{utils}$ directory and finally on the DOS *PATH*.

For a detailed discussion how BIBDB works we refer to Doron (1994). Although BIBDB is very user-friendly you should know how BIB_TE_X works, before you can work with BIBDB properly.

7.2 MakeIndex

The finishing touch of your book or report may be an index. Creating a detailed index is very time consuming for the author but essential for the reader. With MAKEINDEX creating an index in \LaTeX documents is relatively easy, technically speaking.

Creating an index is much like creating a bibliography. First you add the package *makeidx*, e.g.,

```

\documentclass{book}
\usepackage{makeidx}

```

Then you put a *\makeindex* command in the preamble (between the *\documentclass* and the *\begin{document}* command). At the location where you want your index to appear you give the command *\printindex* (usually right before the *\end{document}* command). When this is done you have to specify the entries you want in the index and MAKEINDEX will find the correct page numbers and produces an index table. If you want, e.g., the word *gnat* to be included in the index you put the

command `\index{gnat}` right after the word `gnat` in your document, i.e., `gnat\index{gnat}`. There are several other options to include words in an index. For a detailed discussion about all possibilities we refer to Lamport (1987).

Suppose that the document you are writing is called `sample.tex`. When you run \LaTeX for the first time, the style option `makeidx` together with the command `\makeindex` will produce a file called `sample.idx`. In this file you can find all the index entries. Then you run `MAKEINDEX2`, e.g.,

```
▷ δ:\emtex\utils\mkidx32 sample.idx
```

The result is a file called `sample.ind` that contains the index entries with the corresponding page numbers. The second time you run \LaTeX the `\printindex` command will include the file `sample.ind` so your index gets typeset.

The program `MAKEINDEX` can be found on the directory `δ:\emtex\utils`. It reads the `.idx` file from the current directory.

If you want to create multiple indices (e.g., a normal index, a list of notations used, and a name index) you could use the package `index.sty`. The example below explains how multiple indices are created and printed at the end of your document.

```
%% Create multiple indexes.
%% This is a sample file for INDEX.STY.
%% Here is how to compile a text with multiple indexes:
%% - compile the TeX file
%% - goto MakeIndex menu (Utilities)
%% - change index file extension to *.dx
%% - run MakeIndex
%% - return to main menu
%% - compile the TeX file again
\documentclass{book}
\usepackage{index}
%      ~~~~~ required package for multiple indices!

% start a 'standard' index:
\makeindex
% (this is equivalent to '\newindex{default}{idx}{ind}{Index}'
% note:                ^      ^

% start an index called 'Name Index':
\newindex{aut}{adx}{and}{Name Index}
% note:                ^      ^

% start an index called 'List of Notation':
\newindex{not}{ndx}{nnd}{List of Notation}
% note:                ^      ^

\begin{document}

Here is some text.\index{subject}
% 'subject' will appear in the standard 'Index'
```

2. Again, there is a 32 bits version that \LaTeX uses. It's called `mkidx32.exe`.


```

Here is some more text.\index[not]{notation}
% 'notation' will appear in the 'List of Notation'

Who wrote this?\index[aut]{Someone else}
Here is yet more text.\index[aut]{Knut}
% 'Knut' and 'Someone else' will appear in the 'Name Index'

\printindex[not]
% prints the 'List of Notation'

\printindex[aut]
% prints the 'Name Index'

\printindex
% prints the standard 'Index'

\end{document}

```

7.3 Spell-checking

7.3.1 Concepts

\TeX uses the public domain program `AMSPELL` to check and correct spelling in \TeX documents. `AMSPELL` is basically a spell-checker for plain ASCII files, with some special features for dealing with \TeX files. The program was written by A. Merckens in `TURBO PASCAL` and runs on any MS-DOS computer.

The basic idea behind this program is to make spell-checking easier by

- providing the context of the possibly misspelled word;
- offering alternatives;
- offering facilities for editing the word;
- automatically replacing misspelled words in your document;
- learning new words.

Currently, dictionaries for UK English, US English, Dutch, German, French, Spanish, South-African and Italian are available.

7.3.2 \TeX features

`AMSPELL` does not require \TeX commands to be removed from your document (de \TeX ing). In fact, it will even interpret the standard accenting commands like `\", \', \'` and will automatically use them while replacing incorrectly spelled words. When checking a \TeX file, `AMSPELL` will ignore all text between `$`'s and `$$`'s. Furthermore, `AMSPELL` will ignore parameters of the \LaTeX commands `\ref`, `\pageref`, `\cite`, `\nocite`, `\label`, and all text between `\begin{equation}`, `\begin{eqnarray}`, `\begin{eqnarray*}`, `\[` and their counterparts like `\]`. You can change or expand the lists by means of environment variables. If, e.g., you only want the parameter of `\index` and `\ref` to be ignored, you should set the environment variable `SPL_COM` in the file `texuser.<os>` as follows:

```
SPL_COM=index,ref
```

However, if you want to *add* these commands to the list, you should use

is a simple, menu driven environment which will let you perform the following operations on graphic files: View\index{view}, Convert between any two formats, Print, Dither colour pictures to black and white, Reverse, Rotate, Flip, Scale, Reduce the number of colours Sharpen, Soften, Adjust the brightness, contrast and colour balance of the bitmaps\index{bitmap}.

[F1 help]

	colour	
choler	color	colore
cooler	calory	coiler
collar	calor	cholera
colorer		

colour

Figure 7.1: Correcting misspelled words

SPL_COM=+index,ref

Note that the list must be comma-delimited. Spaces are not allowed, and all characters will be converted to lower case.

If you only want the contents of the environments of, e.g., `figure`, `table` and `verbatim` to be ignored, you should set the environment variable `SPL_ENV` as follows:

SPL_ENV=figure,table,verbatim

However, if you want to *add* these environments to the list, you should use

SPL_ENV=+figure,table,verbatim

The syntax of this environment variable equals that of `SPL_COM`. Both lists can be up to 255 characters long. A list can be emptied by specifying a comma:

SPL_ENV=,

All words starting with a backslash are ignored.

Naturally `AMSPELL` will conform to the capitalization in each word.

7.3.3 The user interface

When you start `AMSPELL`, it will first read your text and look up all the words it found in the dictionaries and your own (e.g., US English) jargon file (e.g., `correct.us`). Then it will search its dictionaries again to find alternatives for words it does not know.

For each unknown word it displays the context, and provides alternatives (if possible). Figure 7.1 shows what the screen looks like when `AMSPELL` encounters the word 'colour', using the US English dictionary.

the Trident graphics card for viewing in super VGA mode. You can resize the picture to any length and width. You can also calculate the BoundingBox and convert\index{convert graphics} the Postscript picture to a PCX bitmap. This PCX bitmap can be used with the DVI drivers\index{dvi-driver} of \emtex\ to view and print the picture i

[F1 help]

Information on special keys when correcting spelling

Enter	: Accept word for this time
^Enter	: Learn word <appended to file LEARN SPL>
F10	: Stop correcting and write changes to file
Left, Right, End, Home	: Keys used for editing selected word
Ins	: Toggle Insert/Overwrite Mode
Del, BS	: Delete character On/Before cursorposition
Down, Up	: Move to Next/Previous word
Pg-Dn, Pg-Up	: Move to word one line Down/Up
^T, ^Y	: Remove rest of line
^BREAK	: Abort

[Press any key to continue]

Figure 7.2: Help information on special keys.

The word ‘colour’ is displayed in reverse video. You can choose one of the alternatives or edit the word yourself. Pressing F1 will show all keys AMSPELL accepts (see Figure 7.2).

Note that unfortunately AMSPELL sometimes gets confused and skips parts of your document. As explained before AMSPELL skips math environment, but in some cases it may fail to recognize the end of a math environment. In case of doubt, e.g., after a complex math construction, add a line like this:

```
% \]
```

In case AMSPELL was in ‘skipping mode’ it will now resume spell-checking your text. Note that AMSPELL does not know that a percent sign starts a comment.

7.4 Graphics

TEX has been developed with the idea that it should run on any operating system (MS-DOS, VMS, VM-CMS, UNIX etc.). Another feature of TEX is that documents can be freely exchanged between operating systems, because documents are written in standard ASCII. Graphics, however, are often machine dependent, so the possibility to include graphics in TEX or L^ATEX depends on the operating system and the DVI-driver you are using. This means that including graphics in TEX or L^ATEX is often not an easy job.

The solution often adopted for the inclusion of graphics is the incorporation of PostScript pictures in the document using the \special command. The \special command is ignored/passed on by the TEX compiler but the PostScript DVI-driver will use the \special command to insert the PostScript picture at the right place and in the right size in your document. The advantage of this method is that for all operating systems there are PostScript DVI-drivers and that PostScript files are usually also written in standard ASCII, therefore you can transfer both text file and graphics to any operating system. The disadvantage of this method is that you can *only* include PostScript pictures in your document and that you need a PostScript printer to produce output, or GHOSTSCRIPT, a PostScript interpreter. The standard EMTEX screen previewer will not display PostScript pictures.

If you have a PC there are other ways to incorporate pictures in $\text{T}_{\text{E}}\text{X}$ documents. Before we discuss them, we will explain different types of pictures. In principle there are two types of pictures, namely bitmap and vector pictures. A bitmap picture is a matrix with the entries corresponding to points with a color. The dimension of the matrix specifies the height and width of the picture. Because of the fixed matrix, manipulating the picture is difficult and resizing the picture often leads to less satisfactory results. However, many graphic packages produce pictures in a bitmap format. These bitmap files come in many different types, mostly as a result of different compression and color encoding techniques. Examples of bitmap pictures are: GIF (Compuserve), TIFF, PCX (PC Paintbrush), BMP (MS-Windows), IFF (Amiga), LBM (Amiga), IMG (Ventura), CUT (Dr Halo), PCL (Hewlett Packard) and JPEG (Joint Photographic Experts Group).

A *vector* picture is specified as a device-independent mathematical description and is therefore easy to manipulate/resize. However, the problem with vector pictures is that most DVI-drivers cannot handle them. Examples of vector pictures are: HPGL (Hewlett Packard Graphics Language), PS (PostScript), and EPS (Encapsulated PostScript).

In the sequel of this section we will discuss the programs GRAPHIC WORKSHOP, PAINTSHOP PRO, HP2XX, BM2FONT, PCLTOMSP, and GHOSTSCRIPT. We will end this section by describing how \LaTeX combines the strength of all these programs to incorporate graphics in $\text{T}_{\text{E}}\text{X}$ documents.

7.4.1 $\text{emT}_{\text{E}}\text{X}$ and the $\backslash\text{special}$ command

When the $\text{T}_{\text{E}}\text{X}$ compiler encounters a $\backslash\text{special}$ command, it will expand its parameter and pass it as such to the DVI file. Nothing else will happen. The DVI-driver is supposed to understand the meaning of the command when it reads the DVI file. DVI-drivers may support different $\backslash\text{special}$ commands. Therefore the $\backslash\text{special}$ command is output device dependent and it should be ignored if the DVI-driver does not support that specific function. The advantage of the $\backslash\text{special}$ command is that you can use device-dependent instructions to produce output, e.g., you can use a PostScript printer to include PostScript pictures in your $\text{T}_{\text{E}}\text{X}$ document.

$\text{emT}_{\text{E}}\text{X}$ defines some $\backslash\text{special}$ commands to include PCX (PC Paintbrush), MSP (Microsoft Paint), black and white BMP (MS-Windows) bitmap pictures and PCL (Hewlett Packard) pictures. The syntax of the $\backslash\text{special}$ command is:

```
 $\backslash\text{special}\{em:graph [path]<bitmap file>\}$ 
```

where $[path]$ is an optional path and $<bitmap file>$ is a PCX, MSP, or black and white BMP bitmap file. The upper left corner of the graphic file is located at the reference point of a character. Run length encoded BMP files and 4 color CGA-mode PCX files cannot be used. All non-white pixels of a PCX file are printed (assuming the standard palette). The width of the graphic must not exceed 32760 pixels, the height must not exceed 32766 pixels. The viewer and the printer drivers of the $\text{emT}_{\text{E}}\text{X}$ package will show and print the bitmap. However, it is not possible to manipulate the picture. This means that different drivers will produce different sized pictures (as a result of the resolution of the device driver and the fixed resolution of the bitmap graphic). For a detailed discussion about $\backslash\text{special}$ commands see Mattes (1991).

When you want to manipulate the picture (e.g., resizing the picture or color reduction), you can for instance use the shareware program Graphic Workshop. If you use MS-WINDOWS 95 or NT we suggest you use the shareware program PAINTSHOP PRO to do the operations described below.

Graphic Workshop is a program for working with bitmapped graphic files. It will handle most of the popular bitmap formats. Graphic Workshop is a simple, menu driven environment which will let you perform the following operations on graphic files:

- View bitmap pictures

- Convert between any two bitmap formats
- Print the bitmap on almost all popular printers
- Dither color pictures to black and white
- Reverse the colors
- Rotate the picture
- Flip the picture
- Scale the picture to any size
- Select a part of the graphic as a new picture
- Reduce the number of colors
- Sharpen and Soften the picture
- Adjust the brightness, contrast and color balance of the bitmaps.

If the picture has more than 32 colors and you use the $\text{EMT}\text{E}\text{X}$ special command to include a graphic an error will appear when viewing or printing the picture. Too many colors will result in too dark and unfocussed pictures. In that case it is much better to use $\text{BM2}\text{FONT}$ to incorporate the picture into a TEX document.

The problem with bitmap files is their fixed dimension, i.e. the size of the picture will depend on the DVI-driver used. Suppose the bitmap file `golfer.pcx` has dimensions 550×770 dots. Using the picture and a 300 dpi DVI-driver the picture will have a width of $550/300 = 1.83'' = 47\text{mm}$ and a height of $770/300 = 2.57'' = 60\text{mm}$. Other DVI-drivers will result in different sizes. The bitmap file is now printed using:

```

\begin{figure}[htbp]           %1
\begin{center}                %2
  \setlength{\unitlength}{1mm} %3
  \begin{picture}(47,65)       %4
    \put(0,65){%              %5
      \special{em:graph golfer.pcx}} %6
  \end{picture}
\end{center}
\caption{This is an example how to
         use pictures in \emtex} %7
\label{fig:emtexexample}      %8
\end{figure}

%1 For the figure environment see Lamport (1986) pages 59, 176
% [htbp] lists admissible locations: h=here, t=top of page,
% b=bottom of page, p=on separate page
% With here.sty you have the extra option H=forced here
%2 Center the picture horizontally
%3 We use millimeters as the unit of measurement
%4 For the picture environment see Lamport pages 101-111
%5 Normally the picture golfer.pcx will be printed from the
% left upper corner of the picture box, but it should be printed
% from the left lower corner, therefore we need \put(0,65)
%6 \special{...} is the emTeX special command to include a
% bitmap picture
%7 Produces a numbered caption on the place where it is issued
%8 Label the figure for future referencing Lamport (1986)
% pages 71, 186

```

The result is shown in figure 7.3.



Figure 7.3: This is an example how to use pictures in EMT_{EX}

There are many macro packages that can help you insert PostScript files in your document, e.g., PSFIG, a macro package written by T.J. Darrel (see Section 7.4.5). With the help of a PostScript DVI-driver, figures are automatically scaled and positioned on the page, and the proper amount of space is reserved.

L^AT_EX users are advised to use the `graphics` or the `graphicx` package. These packages provide a more or less unified interface for incorporating several kinds of graphics. See the L^AT_EX Companion (Goossens et al. (1994)) or the L^AT_EX Graphics Companion (Goossens et al. (1997)) for an in depth discussion of its features, or see the on-line documentation.

7.4.2 BM2Font

BM2FONT is a program written by F. Sowa and is used to convert bitmap pictures to T_EX fonts. These T_EX fonts can be read by the DVI-drivers and are used to view and print pictures. BM2FONT can convert the following bitmap pictures: PCX, GIF, BMP, IFF, LBM, TIFF, IMG, and CUT. For a detailed discussion on how BM2FONT works and all the possible parameters we refer to the manual. Note that BM2FONT can produce several T_EX fonts (i.e. bitmap fonts (extension `.pk`) and T_EX font metric files (extension `.tfm`)) and that the bitmap fonts are resolution dependent. Therefore, the T_EX file becomes resolution dependent.

The program BM2FONT can be found in the `δ:\emtex\utils` directory. Its command syntax is `BM2FONT <bitmap file> [options]`

The result of `bm2font example.pcx` is one or more font files, plus a file called `example.tex`.

All these files can be found in the directories specified by the environment variables `TEXINPUT`, `TEXFONTS` and `DIRPXL`. With the environment variable `TEXINPUT` we specify the path where BM2FONT searches for the bitmap file. The `TEXFONTS` specifies the path where BM2FONT stores the `.tfm` files. The environment variable `DIRPXL` specifies the path for the `.pk` files, i.e. if `DIRPXL=δ:\EMTEX\TEXFONTS\PXL` and we have specified a resolution of 300 dpi the `.pk` files are stored in the directory `δ:\emtex\texfonts-\pxl300`. If these environment variables do not exist the files are written to the current directory. The file `example.tex` (written to the current directory) uses the picture fonts and defines a macro called `\setexample` (i.e. consisting of the the word 'set' and the filename 'example' (without file extension)). The picture is now produced simply by giving the command `\input{example.tex}` and the command `\setexample` on the location where you want the picture.



Figure 7.4: This is the file kitten.bmp converted to \TeX fonts

We will end this section with an example. Suppose we have a BMP bitmap file `kitten.bmp` and we want to convert this bitmap to \TeX fonts for a 300 dpi printer. Running `BM2FONT`

```
▷ bm2font kitten.bmp -h300 -v300 -m50 -n50
```

will result in one \TeX font metric file (`kittena.tfm`), one font (`kittena.pk`), and the \TeX file `kitten.tex` containing the macro `\setkitten`. The parameters `-h` and `-v` are the horizontal and vertical resolution of the printer, the parameters `-m` and `-n` are the width and height you want the picture to be in millimeters. To produce the picture in the \TeX document we can use, e.g.:

```
\begin{figure}
\centerline{\input{kitten}\setkitten}
% the file kitten.tex defines the macro \setkitten
\caption{This is the file kitten.bmp converted to \TeX\ fonts}
\end{figure}
```

`BM2FONT` gives excellent results and is easy to use. A disadvantage however is that you need to generate picture fonts for every printer you use. For example when you use a 300dpi laserprinter as well as a 600dpi PostScript printer, `BM2FONT` will give you `.pk` and `.tfm` files with the same name but for different printers. To avoid name clashes and wrong picture sizes \LaTeX does some bookkeeping: the font files produced by `BM2FONT` are stored in directories named after the printer for which they were generated.

Note: If the length of the filename of the picture file is eight characters, the last character will be omitted for the construction of the `.tfm` and `.pk` files (because of the addition of the font numbers `a`, `b`, ...). For instance `scrndump.pcx` will produce `scrnduma.tfm`, `scrnduma.pk`, and `scrndum.tex` and the macro is called `\setscrndum`.

Note: No digits are allowed in the picture filename. E.g., `screen1.pcx` has to be renamed to `screeni.pcx`. This, of course, is the result of the \LaTeX macro name constructed by `BM2FONT` using the filename, i.e. in \TeX it is not allowed to use digits in macro names.

Note: JPEG pictures are not supported by `BM2FONT`, so \LaTeX converts them first to GIF format, using e.g., `GRAPHIC WORKSHOP` or `PAINTSHOP PRO`.

Note: If you have an (Enhanced) Windows Metafile (`.wmf` or `.emf`) you can use a program like `PAINTSHOP PRO` or `IRFANVIEW` to convert the Windows Metafile to one of the graphic formats as described above.

If you have a PCL bitmap file (Hewlett Packard LaserJet and DeskJet graphic output) you can convert this file to a MSP or a PCX bitmap. The conversion can be done by E. Mattes's conversion program `pcltomsp`. After conversion you can use the special command to include the graphic (see Section 7.4.1) or the program `BM2FONT` to generate \TeX fonts. By entering e.g.,

```
▷ pcltomsp -qop graph17.lj graph17
```

you convert the PCL file `graph17.lj` to the PCX file `graph17.pcx`, suppressing the program title and any warnings.

7.4.3 HPGL files and HP2xx

Some graphics programs produce graphic files in the *HPGL* format (Hewlett Packard Graphic Language). These are vector pictures specifically made for Hewlett Packard plotters. Because we can only use PostScript and certain bitmap pictures in \TeX documents we need to convert such files. The extensions often used for HPGL files are `.hpp`, `.plt`, and `.hpg` and are produced by, e.g., the programs `MATLAB`, `GAUSS`, and `HARVARD GRAPHICS`. To convert HPGL plotter files \TeX uses the program `HP2XX`.

`HP2XX` is a free software program written by H. Werntges that can be used to print, view and convert HPGL plotfiles. It can be found in the `δ:\emtex\utils` directory. We refer to the documentation for a detailed discussion of all the possibilities (e.g., rotation, picture size, pen color, magnification etc.). \TeX uses `HP2XX` to convert HPGL files to *PCX* bitmap files and *EPS* (Encapsulated PostScript) files. `HP2XX` uses no environment variables, it reads and writes the files from the current directory. `HP2XX` can also display graphs on the screen. It supports super VGA modes. `HP2XX` is easy to use and its output is of excellent quality. You can convert HPGL pictures to *MF* (*METAFONT* format), *CAD* (to be used with `TEXCAD`), *EM* (`EM \TeX` specials), *EPIC* (the Enhanced Picture style), *IMG-*, *PBM-*, *PCL-* and *PCX-*bitmaps and *EPS* pictures.

Suppose we have a file `example.hpg`. We can convert this file to a *PCX* bitmap file `example.pcx` with height 100 mm (the width is automatically calculated, preserving the aspect ratio) using the command

```
▷ hp2xx -mpcx -d300 -h100 -f example.pcx example.hpg
```

`example.pcx` can now be used on a 300 dpi output device. Instead of converting the picture to a *PCX* bitmap we can also convert it to an Encapsulated PostScript file (use `-meps` and `-f example.eps` instead of `-mpcx` and `-f example.pcx`). After conversion you can proceed as discussed in Section 7.4.5.

As an example we show a *LOTUS 1-2-3* picture that is generated as a HPGL file and then converted to a *PCX* bitmap and an *EPS* picture using the commands described above. After conversion we used the style file `figures.sty` (see Section 7.4.5) to print the picture in Figure 7.5.

7.4.4 PostScript and Ghostscript

If you want to view, print and manipulate PostScript files and you do not have a PostScript printer (or commercial software), we suggest you use the free software program `GHOSTSCRIPT` from Aladdin Enterprises. Using `GHOSTSCRIPT` you can view and print `.ps` and `.eps` files on almost any screen or printer. `GHOSTSCRIPT` also supports the Tseng ET-4000 and the Trident graphics card for viewing in super VGA mode. You can resize the picture to any height and width. You can also calculate the `BoundingBox` and convert the PostScript picture to a bitmap in e.g., *PCX* format. This *PCX* bitmap can be used by the `EM \TeX` DVI drivers to view and print the picture in \TeX documents on non-PostScript output devices (see Section 6). `GHOSTSCRIPT` looks for the specified files in the current directory. If it cannot find a file it will look in the directory specified by the `GS_LIB` environment variable. For a detailed discussion how `GHOSTSCRIPT` works we refer to the `GHOSTSCRIPT` documentation. `GHOSTSCRIPT` with all

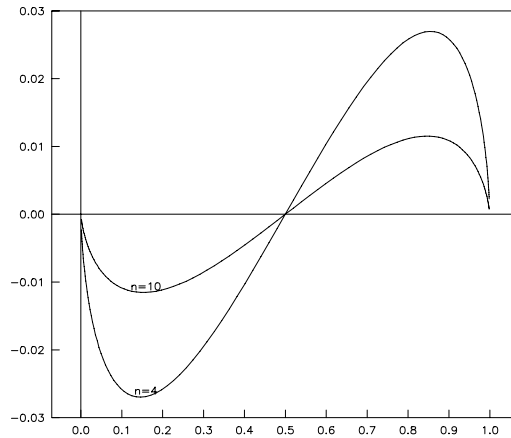


Figure 7.5: An HPGL picture converted to PCX or EPS

its files takes a lot of disk space and is placed in a separate directory $\delta:\text{\emtex\utils\gstools\gs}$. WINDOWS 95/NT users are advised to use GSVIEW to view/print .ps, .eps, and .pdf files.

Note: GHOSTSCRIPT is free software, is regularly updated and gives good results. The only disadvantage is perhaps that it is not very user friendly and poorly documented for beginners.

7.4.5 The FIGURES and PSFIG style files

PSFIG is a macro package written by T.J. Darrell to incorporate PostScript figures in a \TeX document. With the help of a PostScript DVI driver, figures are automatically scaled and positioned on the page, and the proper amount of space is reserved. To include a PostScript picture, simply input PSFIG at the top of your plain \TeX document:

```
\input psfig
```

\LaTeX users will include PSFIG as follows:

```
\documentclass{article}
\usepackage{psfig}
```

and, when you wish to include a figure, invoke the macro with, e.g., (in \LaTeX):

```
\begin{figure}
\begin{center}
\psfig{figure=tiger,width=50mm,height=50mm}
\end{center}
\caption{The use of PSFIG to insert a PS picture}
\end{figure}
```

The result is shown in Figure 7.6.

Note that spaces in the arguments of the macro are not allowed. For a detailed discussion of all possibilities (e.g., rotation, scaling etc.) we refer to Darrel (1992) and Goossens (1993). The PSFIG-macro will generate some \special commands to claim the correct space and size, and with a PostScript DVI driver the picture will be printed correctly.

The PSFIG files can be found in the directory $\delta:\text{\emtex\inputs\misc}$. PSFIG will look for the specified picture files in the current directory, if not found it will look in the directory specified by the \TeX



Figure 7.6: The use of PSFIG to insert a PS picture

command `\psfigurepath`. PSFIG will first check if the specified file is a PostScript file, i.e. it will check whether the file starts with `%!` . If not it will generate a warning message. Then it will look for a *BoundingBox* (i.e. the PostScript command that specifies the location and size of the picture) in the specified picture file, e.g., `example.ps`. If not found it will look for the `BoundingBox` in the file `example.bb`, and if it does not contain a `BoundingBox`, an error message is displayed. The PSFIG-macro will generate some `\special` commands to claim the correct space and size, and with a PostScript DVI driver the picture will be printed correctly.

The style file `FIGURES` is a modification of the PSFIG style file and is used in \LaTeX . It combines the possibility to print/view PCX pictures with the `EMTeX` special commands, and to print PostScript files as with PSFIG. By default (or when using the command `\pcx`) `FIGURES` will try to use PCX picture files. If not found or if you used the `TeX` command `\postscript`, `FIGURES` will look for PostScript files (`.eps` and `.ps`). This makes it possible to view/print pictures using the `EMTeX` special commands and to print the pictures in a `TeX` document on any printer. To include a picture include the `FIGURES` package at the top of your document:

```
\documentclass{article}
\usepackage{figures}
```

and, when you wish to include the figure (`example.pcx` or `example.eps`), call the macro like this:

```
\putfigure{figure=example,width=2in,height=3in}
```

Note that the extension of the picture file is not specified. All commands defined in the style file `PSFIG` are also available (see e.g., Goossens (1993)). Some names of the macros of PSFIG are changed, e.g., `\psfig` is changed in `\putfigure` and some extra macros are added, e.g., `\pcx` (use PCX files), `\figurefull` (the same as `\psfull`); `\figuredraft` (the same as `\psdraft`).

\LaTeX users who prefer to use the `graphics` or `graphicx` package might write the following code:

```
\documentclass{article}
\usepackage[dvips]{graphicx}
\begin{document}
\includegraphics[width=2in]{example.eps}
\end{document}
```

Note that here the file extension (`eps` in this case) must be supplied.

7.4.6 T_EXcad

T_EXCAD is a drawing program written by G. Horn for producing drawings in L^AT_EX documents. It allows the objects available in the L^AT_EX picture environment to be drawn and edited. Its output is a sequence of L^AT_EX picture commands which can be inserted into a L^AT_EX document to generate the drawing. A mouse is not required but strongly recommended. If used, the mouse driver must be loaded before starting T_EXCAD.

The program can be found in the directory $\delta:\backslash\text{emtex}\backslash\text{utils}$. When started it will read the parameter file `texcad.opt`. It will first look in the current directory for this parameter file and if not found it will look in the directory containing `texcad.exe`. T_EXCAD supports the special commands of EMT_EX for line drawing (very useful for drawing lines at any angle) but you should not forget to include the style file `emlines2.sty` in your document. Likewise, if you use *bezier* curves, you must include the style file `bezier.sty`. For a detailed discussion how to install and use T_EXCAD we refer to Horn (1990).

7.4.7 L^AT_EXcad

For L^AT_EX users the program L^AT_EXCAD by John Leis might be interesting. It is a powerful drawing program that can output a L^AT_EX picture environment, that can be included in any L^AT_EX document with little effort. Note that the `latexcad.sty` is required to compile such a picture.

7.4.8 Mayura Cad

This MS-WINDOWS program is another fine drawing program, written by Mayura Software. It can output (“export”) pictures in EPS, TIFF, Adobe Acrobat PDF, Adobe Illustrator, BMP bitmap and Windows Metafile WMF. Note that if Adobe Type Manager is not installed on your system, the program will warn you that you will not be able to use text in your pictures. All other drawing tools will work, though.

7.4.9 GNUplot

GNUPLOT is a general plotting program written by C. Kelly and T. Williams that comes in both an MS-DOS and an MS-WINDOWS version. It is capable of plotting curves from mathematical expressions, or from data files. It can export pictures in a variety of ways. Most important for T_EX users are eps (PostScript), L^AT_EX picture environment (with or without EMT_EX specials to allow for lines at any angle).

7.4.10 QFig

QFIG is a drawing program written by T. Iwakuma for producing drawings in T_EX or L^AT_EX documents. Its output is either in PICT_EX, EEPIC, or EEPICEMU format. In order to use such a picture you must load the appropriate package before including the picture.

A mouse is not required but strongly recommended. If used, the mouse driver must be loaded before starting QFIG. Remember that you can exit the program only by pressing Ctrl Q.

The program can be found in the directory $\delta:\backslash\text{emtex}\backslash\text{utils}$.

7.4.11 Metafont

METAFONT can not only be used to produce fonts. By loading the MFTOEPS package created by B. Jackowski, P. Pianowski and M. Ryćko you can generate PostScript output (Encapsulated PostScript), readable by e.g., CORELDRAW! and ADOBE ILLUSTRATOR.

See also Section 4.11.

7.4.12 Metapost

The METAPOST system implements a picture-drawing language very much like Knuth's METAFONT except that it outputs PostScript code instead of bitmaps. METAPOST is a powerful language for producing figures for documents to be printed on PostScript printers. It provides easy access to all the features of PostScript and it includes facilities for integrating text and graphics. The output is in Encapsulated PostScript format. We recommend that you read J. Hobby's METAPOST manual.

See also Section 4.12

7.4.13 Summary

In the preceding sections we have discussed several ways to incorporate graphics in T_EX documents.

The main problem when including a picture is to know which program to use and which parameters and commands are needed to get reasonable output. The 4T_EX workbench tries to shield you from these dirty bits.

4T_EX's GRAPHICS utility helps you incorporate pictures. All the programs discussed above are used, but a user does not need to remember all the (program specific) parameters. Simply choosing from an options list you can specify the parameters. In many cases this is not necessary because most parameters are set automatically (e.g., the printer resolution). The user need only specify the size of the picture and then *convert* the picture. Converting means that the correct programs are called and that the result is a T_EX file that can be used to insert the picture in your document. When for instance you want to include the picture acad.hpp, 4T_EX will take care of the necessary conversion and tell you when it is finished that the picture can be inserted simply by adding the style file figures.sty and using the statements:

```
\begin{figure}
  \centerline{\input{ACAD}}
  \caption{your title}
\end{figure}
```

This way a uniform approach of inserting pictures is reached, i.e. it does not matter if you use HP2XX, BM2FONT, PCLTOMSP, or GHOSTSCRIPT. After conversion only one simple input statement will suffice.

4T_EX also makes it possible to view and manipulate the picture. Viewing is also possible in super VGA modes. For instance a PostScript picture can be rotated, a BoundingBox can be calculated and 4T_EX magnifies the picture so that it will fit the specified size exactly. The same holds for HPGL pictures. Bitmap pictures can be manipulated using e.g., GRAPHIC WORKSHOP or PAINTSHOP PRO.

All the necessary bookkeeping is done by 4T_EX. For instance the fonts (*.pk and *.tfm) are stored in the correct directories and the conversion file is stored in the current working directory.

The conversion of pictures is done in such a way that it allows you to view and print the T_EX document with the pictures on any screen or printer. When you want to use a PostScript printer you need to include the command \postscript in the document.

7.5 L^AT_EX Help

When writing a document in L^AT_EX you sometimes or perhaps often need to look up a control sequence, symbol or other information.

Unfortunately Lamport's L^AT_EX book (and many other L^AT_EX related books) are often too elaborate, or you simply can't find what you are looking for because it's not in the index. Finding documentation can also be hard if the index has 5 or more entries to the subject, as is often the case in the T_EX book.

A more convenient way to get help on L^AT_EX related items is a MS-WINDOWS hypertext help system that can be started from the L^AT_EX utilities menu. The help file describes L^AT_EX 2_ε, not L^AT_EX 2.09. The help was adapted from the old L^AT_EX 2.09 help file by T. Martinsen.

7.6 L^AT_EX Mac

L^AT_EX MAC is an MS-WINDOWS program that can make writing a T_EX file more WYSIWYG (What You See Is What You Get). This program, written by J.M. Aguirregabiria offers several 'toolbars' with e.g., all mathematical operators or all Greek symbols. By 'clicking' on e.g., a Greek lambda the program will insert the T_EX code `\lambda` in your text. Likewise, it can insert complete L^AT_EX environments, and more. Text is inserted at the current cursor position, so beware that the cursor in your editor is at the right position when you choose an item from L^AT_EX MAC's toolbar.

Note that L^AT_EX MAC can only communicate with your editor after you have told L^AT_EX MAC what your editor is. You can pick an application that L^AT_EX MAC will write to from the output destination selector. Note also that your editor itself does not need to be a Windows editor. In fact, keys can be pasted into any application.

7.7 T_EXchk

T_EXCHK is a program from the public domain that checks a L^AT_EX file for errors in matching {}, [], environments (`\begin{xxx}` – `\end{xxx}`), formulas etc. However, this program is far from perfect. Many errors are not detected, and some errors that it finds are spurious.

The syntax is: `TEXCHK [-v -c -r -2] [file1 file2 ...]`

The program is documented in `texchk.doc`. L^AT_EX uses only the `-v` switch (verbose output). This switch is very helpful when L^AT_EX complains that 'environment *x* ended by environment *y*'. T_EXCHK clearly shows the nesting of environments.

Part 8

Overview of software used by \LaTeX

8.1 Free software

Free software is software *free* of charge and may be used by anyone who wishes to, including commercial/educational environments. However, you are not allowed to change the software in any way, to remove the copyright statement, to sell the programs, or bundle it with other commercial goods. You may give them to anyone you like, but only in their original form with the complete documentation.

In some cases software is free only for non-commercial and/or educational use. You should always read the documentation to make sure your particular use is in accordance with the licensing statement of that piece of software.

The \LaTeX Workbench uses the free software programs described in Table 8.1.

Table 8.1 Free software

Program	Author(s)	Description
abc.exe	A. Merckens	split file into separate files according to the word initials, used with <code>amsPELL.exe</code>
amsPELL.exe	A. Merckens	ASCII / \TeX spell-checker
ansi.com	M. Mefford	emulate ANSI functions for screen and keyboard
asc2tex.exe	T. Götz	translate ASCII screen dump to \LaTeX picture environment
asc2wp.exe	G. Johannsen	translate ASCII files with ‘hard returns’ to ‘soft returns’ for WORDPERFECT input
b5conv.exe	W. Lemberg	preprocess raw Big-5 text to CJK
bibclean.exe	N. Beebe	clean/check bib \TeX files
[w]bibdb.exe	E. Doron	BIB \TeX database manager
bibtex[32].exe	O. Patashnik	E. Mattes’ MS-DOS version of BIB \TeX
bm2font.exe	F. Sowa	converts bitmap graphics in \TeX fonts
bounding.exe	4U	finds bounding box in PS and EPS files
change.exe	B. Guthrie	replace strings in text files
chi2tex.exe	I. Zakharevich	convert CHIWRITER files to \TeX
chi2ltx.exe	M. Gomulinski	convert CHIWRITER files to \LaTeX
concopy.exe	C. Dunford	run a program and direct screen output both to standard output and a file
crlf.exe	K. Hartnegg	change CarriageReturn in CarriageReturn + LineFeed
detex.exe	D. Trinkle	strip \TeX commands from a <code>.tex</code> file
dvi2tty.exe	S. Lindahl	convert DVI file to formatted ASCII text

Table 8.1 (continued)

Program	Author(s)	Description
dvi5x0.exe	F. Sowa	print DVI file on HP DeskJet 500 series color printers
dvidvi.exe	Radical Eye Software	pagination tricks for .dvi files
dvi[h]ps*.exe	Radical Eye Software	print DVI file on PostScript printer
dvivgac.exe	F. Sowa	view DVI file prepared for HP color DeskJet printer on VGA screen
dvitomp.exe	J. Hobby	part of the METAPOST program
EM \TeX	E. Mattes	\TeX compilers, dvi-drivers, METAFONT
emx.exe	E. Mattes	DOS memory extender using VCPI
expand.i.exe	4U	expand dictionaries of ISPELL, used with AMSPELL
filt.exe	L.G. Institut Fourier	convert 8 bits ASCII text to \TeX code and vice versa
gawk.exe	Aho - Kernigham - Weinberger	pattern processing language
[w]gnuplot.exe	C. Kelly and T. Williams	plotting program
grep.com	Borland International	search for strings in text files
gs[386].exe	Aladdin Enterprises	print/display PostScript files on non-PostScript devices
gsview32.exe	Russell Lang	display PostScript files
hp2xx.exe	H. Werntges	view/convert/print HPGL pictures
html2lat.exe	N. Torkington	convert \LaTeX to HTML
i_view.exe	I. Skiljan	Windows graphics viewer
inclist.exe	4U	trace all input and include statements in a \TeX file
kill.exe	L. Kahn	kill Windows processes
latexcad.exe	John Leis	draw pictures
latex.hlp	T. Martinsen	Windows help file on \LaTeX
latexmac.exe	J.M. Aguirregabiria	paste \LaTeX commands to editor
mkidx32.exe	P. Chen	MS-DOS version of the \LaTeX makeindex program
mp.exe	J. Hobby	the METAPOST program (also dvitomp.exe and mptotex.exe)
mptotex.exe	J. Hobby	part of the METAPOST program
midi2tex.exe	H. Kuykens	convert MIDI files to music \TeX
musixflx.exe	D. Taupin & R. Mitchell	improve music \TeX output
nospace.exe	4U	remove trailing blanks from a text file
numoff.com	??	turn keyboard 'numlock' off
opera.exe	E. Young	WWW browser
pcltomsp.exe	E. Mattes	convert PCL files to MSP or PCX
pcwtex.exe	J. Breen	convert PCWRITE files to \LaTeX
ps.exe	L. Kahn	show Windows processes
ps2pk.exe	P. Tutelaers	produce .pk files from PostScript fonts
qfig.exe	W. Ofosu-Amach & L. Ben-Brahim	graphics drawing program for (La) \TeX
qe32.exe	J. Maddock	Windows editor

Table 8.1 (continued)

Program	Author(s)	Description
rsx.exe	R. Schnitker	DOS memory extender using DPMI
rtflatex.exe	D. Taupin	convert RTF text to L ^A T _E X
selektie.exe	4U	file selection program
sort.exe	T. Ohl	sort files
spatieer.exe	4U	add a space between a persons initials and name, used for BIB _T E _X database files
spcomp.exe	A. Merckens	compress dictionary files for use with AMSPELL
spdecomp.exe	A. Merckens	decompress dictionary files for use with AMSPELL
strpcrlf.exe	K. Hartnegg	convert MS-DOS text to UNIX or MACINTOSH format
t1binary.exe	L. Hetherington	convert PostScript .pfa font file to .pfb format
tex4ht.exe	E. Gurari	convert T _E X to HTML
texcad.exe	G. Horn	drawing program for L ^A T _E X documents
texchk.exe	J. Massar	check L ^A T _E X file for correct syntax
texutil.exe	H. Hagen	solve references for ConT _E Xt format
tgrind.exe	??	convert ASCII program listing to T _E X
ttb.exe	U. Fastenrath	convert TIB bibliography files to BIB _T E _X
untex.exe	M. Staats	strip T _E X commands from a .tex file
unzip386.exe	Adler - Wales - Gailly - Rommel	decompress 'zipped' archives
viewpcl.exe	4U	view PCL pictures
wordlist.exe	4U	list words in T _E X files
wd2latex.exe	C.J. Thomas	convert MS-WORD files to L ^A T _E X
wp2latex.exe	R.C. Houtpen, G. Geers, and M. Covington	convert WORDPERFECT 5.1 text files to L ^A T _E X files
wp2x.exe	??	convert WORDPERFECT 4.2 text files to T _E X or L ^A T _E X

8.2 Shareware

You may use shareware software for an evaluation period, usually 21 to 30 days. The purpose of this evaluation period is to allow you to determine whether the program meets your needs before purchasing it. Once the evaluation period has ended, you agree to either purchase a registered copy of the program, or to stop using it. If you have ordered a registered copy of the program from a dealer, you may continue to use your shareware copy beyond the end of the evaluation period until your registered copy arrives. Note that when the evaluation period ends a program may stop working or may start nagging you more and more.

While you are evaluating the program, you may use it on as many computers as are required to perform your evaluation. Your evaluation period begins when you first install the program on one or more computers for evaluation purposes.

You may make copies of shareware programs to give to others, as long as you include all of the files that you originally received with your shareware copy, as listed in the documentation file included with your shareware copy. When you give a shareware copy of the program to another person, you agree to inform them that their copy is to be used for a time limited evaluation period, and that they must purchase a

Table 8.2: Shareware software

Program	Author(s)	Description
4DOS	JP Software Inc.	replacement of standard command interpreter
ACDSEE	D.S. Hopper	Windows graphics viewer/convertor
GWS	Alchemy Mindworks	display/print/convert . . . bitmap pictures
CSHOW	CompuShow	view bitmap pictures
CONVDW	CrossCourt Systems	convert DISPLAYWRITEfiles to T _E X
HTMLCON	S. Township	convert HTML to ASCII
MAKURY DRAW	Makury Software	draw pictures
MMA2LTX	G. Ghibó	convert Mathematica picture to L ^A T _E X
PAINTSHOP PRO	Jasc, Inc.	bitmap graphics program

registered copy if they continue to use the program once the evaluation period has ended. You agree not to sell shareware copies of the program or distribute them to others for any kind of compensation or fee.

Note that distribution sets of all freeware and shareware programs supported by 4T_EX can be found on the second CD-rom in directory \DISTRIB.

The 4T_EX workbench uses the shareware programs described in Table 8.2.

8.3 Editors

4T_EX offers you an extensive choice of editors that we think are very suitable for writing your T_EX files.

The installation script allows you to select an editor, but on the CD-roms you will find even more editors. Here is a list of all editors included.

On the first CD-rom you will find pre-installed versions of many of these editors in directory \EDITORS. You can find complete distribution sets of all these editors on the second CD-rom in directory \DISTRIB-\EDITORS.

Editor	Author(s)	Runs on*	Free/shareware
AURORA 3.0	nuText Systems	D	S
BREEZE 5.6	K. Solway	D	S
EDDI4TEX 2.01	U. Jahnz	D	S
EDITPAD 3.02	J. Goyvaerts	W	F
EMACS 19.34	???	W	F
GWD TEXT EDITOR 1.5	V. Gaco	W	S
MULTI-EDIT 7.00J	American Cybernetics, Inc.	D/W	F
NOTGNU 1.7	J. Melbin	D/W	F
OEMACS 19.30	???	D	F
PEDIT 2.0	P. Brand	D	F
PFE 7.001	A. Phillips	W	F
PROLIX 2.0	E. Schreiber	W	F
QUICKEDIT 1.1	J. Maddock	W	F
REDIT 2.10	M. Rogalla	D	F
SUPER NOTETAB 2.63	E. Fookes	W	F
TE 0.7	P. Swatzki	D	F
TSE JUNIOR 4.0	Semware Inc.	M	S
TSE 2.5	Semware Inc.	M	“testdrive”
TSE 2.8	Semware Inc.	W	“testdrive”
VEDIT 4.0	J. O’D Rooney	D	S
WINEDT 1.4	A. Simonic	W	S
ZEUS 2.50	J. Jumppanen	W	S

* D stands for MS-DOS; W stands for MS-WINDOWS 95 or NT.

Part 9

T_EX support

Because T_EX is not a commercial program, but completely free, there is no official support person that will help you in case of trouble. This may seem like a serious drawback, but can also be regarded as an advantage.

All over the world T_EX users have formed a network of *user groups* on an informal basis. These user groups consist of enthusiastic T_EX users who share their problems and solutions with anyone who wants to join the T_EX community. They usually communicate through electronic mail (*e-mail* for short) and often produce a printed periodical with contributions from members.

The periodicals and discussions on e-mail are essential for users who want to be informed about the latest developments. E-mail is important for getting information on, e.g., what is the newest version of program *x*, where do I find a macro for problem *y*, how do I install program *z*. For this purpose, a list of frequently asked questions ('FAQ') is maintained, and is distributed regularly.

On the second CD-rom you will find lots of additional information such as 'FAQ's, courses on T_EX/L^AT_EX in several languages and documentation on T_EX related software.

9.1 T_EX user groups around the world

Below we have listed all T_EX user groups currently know to us. On the second CD-rom you will find a directory \USERGRPS in which you will find much more information on many of them.

AsT_EX (French-speaking)

Michel Lavaud, President
Association pour la diffusion de logiciels scientifiques lies à T_EX Association AsT_EX
BP 6532
45066 Orleans cedex 2
France
Tel: 33 2 38 64 09 94
e-mail: astex-admin@univ-orleans.fr
discussion list astex@univ-orleans.fr

CsTUG (Czech and Slovak Republics)

Petr Sojka, President
Československé sdružení uživatelů T_EXu
CsTUG, c/o FI MU
Botanická 68a

CZ-602 00 Brno
Czech Republic
e-mail: cstug@cstug.cz
WWW page: <http://www.cstug.cz/>

CyrTUG (Russia)

Irina Makhovaya, Executive Director
Associaiaia Pol'zovatelei Kirillicheskogo T_EX'a
Mir Publishers
2, Pervyi Rizhskii Pereulok
Moscow 129820
Russia
Tel: +7 95 286 0622, 286-1777
FAX: +7 95 288 9522
e-mail: cyrtug@cemi.rssi.ru
WWW page: <http://www.semi.rssi.ru/cyrtug/>

Dante e.V. (German-speaking)

Joachim Lammarsch, President
Deutschsprachige Anwendervereinigung T_EX e.V.
Postfach 101840
D-69008 Heidelberg
Germany
Tel: +49 6221 29766
FAX: +49 6221 167906
e-mail: dante@dante.de
WWW page: <http://www.dante.de>

Estonian User Group

Enn Saar, Tartu
Astrophysical Observatory, Toravere
EE 2444 Estonia
e-mail: saar@aai.ee

Greek T_EX Friends Group (Greek speaking)

Apostolos Syropoulos, President
366, 28th October Str.
GR-671 00 Xanthi
Greece
Tel: +30 541 28704
e-mail: apostolo@platon.ee.duth.gr
WWW page: <http://obelix.ee.duth.gr/eft/>

GUST (Poland)

Tomasz Przechlewski (President)
Polska Grupa Uzytkownikow Systemu T_EX
Instytut Matematyki Uniwersytetu Gdańskiego
ul. Wita Stwosza 57
80 - 952 Gdańsk
Poland
e-mail: ekotp@univ.gda.pl
WWW page: <http://www.gust.org.pl/>

GUTenberg (French-speaking)

Michel Goossens, President
Groupe francophone des Utilisateurs de T_EX

Association GUTenberg
BP 10
F-93220 Gagny principal
France
Tel: +33 1 44 32 37 96
FAX: +33 1 44 32 20 80
e-mail: gut@irisa.fr
WWW page: <http://www.ens.fr/gut/>

GUTH (Grupo de Usuarios de T_EX Hispanoparlantes)

No formal user group yet.
Public mailing list: spanish-tex@eunet.es (Send subscription requests to this list).
Julio Sanchez
GMV SA
Isaac Newton 11
PTM Tres Cantos
E-28760 Madrid
Spain
e-mail: jsanchez@gmv.es
WWW page: <http://gordo.us.es/Actividades/GUTH/>

ITALIC (Irish)

No formal user group yet.
Public mailing list: ITALIC-L@irlearn.ucd.ie (send subscription requests to listserv@irlearn.ucd.ie).
Peter Flynn
Computer Centre
University College Cork
Ireland
e-mail: pflynn@www.ucc.ie

iT_EXnici (Italian)

(Unofficial) Italian T_EX Users Group
Giovanni Pensa
e-mail: pensa@dsi.unimi.it

JTUG (Japan)

Nobuo Saitoh, Chairman
Japan T_EX Users' Group
Faculty of Environmental Information
Keio University
5322 Endo, Fujisawa-shi
JP-252 Japan
Tel: +81 466 47 5111
e-mail: ns@keio.ac.jp

Lithuanian T_EX Users Group

Vytas Statulevicius, Chair
Akademijos 4
LT-2600 Vilnius
Lithuania
Tel: +370 2 359 609
FAX: +370 2 359 804
e-mail: statulevicius@mii.lt

Nordic T_EX Users Group (Scandinavian countries)

Dag Langmyhr, Chair
Nordic T_EX Users Group
Department of Informatics
PO Box 1080 Blindern
University of Oslo
N-0316 Oslo
Norway
Tel: +47 22 85 24 50
FAX: +47 22 85 24 01
e-mail: dag@ifi.uio.no
WWW page: <http://www.ifi.uio.no/~dag/ntug/ntug.html>

NTG (Dutch-speaking)

Erik Frambach, Chair
Nederlandstalige T_EX Gebruikersgroep
Postbus 394
NL-1740 AJ Schagen
The Netherlands
e-mail: ntg@nic.surfnet.nl
WWW page: <http://www.ntg.nl/>

T_EXCeH (Slovenian T_EX User Group)

Vladimir Batagelj
Jadranska 19
SI-61111 Ljubljana
Slovenia
e-mail: texceh@uni-lj.si

Tirant lo T_EX (Grup d'usuaris catalanoparlants de T_EX)

No formal user group yet.
Mailing list: catala-tex@aligna.cesca.es
e-mail: valiente@lsi.upc.es
WWW page: <http://www-lsi.upc.es/~valiente/tug-catalan.html>

TUG (International user group)

Mimi Jett, President
T_EX Users Group
1466 NW Front Avenue, Suite 3141
Portland, OR 97209
USA
Tel: +1-503-223-9994
FAX: +1-503-223-3960
e-mail: tug@tug.org
WWW page: <http://www.tug.org/>

UK TUG (United Kingdom)

Robin Fairbairns, Chairman
UK T_EX Users' Group
For information:
Peter Abbott
1 Eymore Close
Selly Oak
Birmingham B29 4LB
England

e-mail: uktug-enquiries@tex.ac.uk
WWW page: <http://www.tex.ac.uk/UKTUG/>

9.2 Mailing lists

Many of the user groups listed above communicate through e-mail, by means of *mailing lists*. A mailing list is an electronic mail system that anyone can subscribe to. All mail that is posted to the list is automatically distributed to all subscribers. These lists are often used to ask for help and to discuss problems, new developments etc.

Here is a list of some active mailing lists:

TEX-L	e-mail: TEX-L@FRULM11.BITNET
TEXHAX-L	e-mail: TEXHAX-L@JPNKSCI.BITNET
TEX-D-L	e-mail: TEX-D-L@DEARN.BITNET
TEX-NL	e-mail: TEX-NL@NIC.SURFNET.NL
UKTEX	e-mail: UKTEX@FRULM11.BITNET
\LaTeX	e-mail: 4TEX@NIC.SURFNET.NL

You can subscribe to a mailing list by sending a message to the list server, LISTSERV@NIC.SURFNET.NL or (if you know) where the list is maintained, e.g., LISTSERV@FRULM11.BITNET. The message should contain just one line:

```
subscribe tex-nl Foo Bar
```

where ‘Foo Bar’ is your real name. Note that you should *not* send your request for subscription to, e.g., TEX-NL@NIC.SURFNET.NL, because this is the mailing list itself. In that case your request would be distributed to all subscribers, which will not be appreciated.

Once your request has been accepted you will receive an introductory greeting mail, explaining how to use the list.

To give you an impression of what kind of discussions you can expect on certain mailing lists we have included messages from several lists on the second CD-rom (directory `\DIGESTS`). Of course these can also be helpful in finding solutions to problems you are having, and which have already been discussed (and hopefully solved) on these lists.

9.3 \LaTeX Support

If you have trouble installing \LaTeX or need more information you can send e-mail to

```
4TeX-support@eco.rug.nl
```

However, don’t expect an answer within the hour. We will try to help you as soon and as best as we can, but \LaTeX is an ‘after-hours’ project.

\LaTeX users can join the \LaTeX mailing list. On this list, users can pose/answer questions regarding \LaTeX . New or desired developments and features are also announced and discussed on this list.

Subscribing to this list is very easy. Send the following message to `listserv@nic.surfnet.nl`

```
subscribe 4tex Foo Bar
```

where 'Foo Bar' is your real name. At the moment, about 280 people from 36 countries have joined the list.

Updates and bug fixes will be available through 'anonymous ftp' from the server `ftp://frambach.eco.rug.nl/4tex/`.

9.4 Bulletin Boards

For those who are not connected to the Internet, *Bulletin Boards* are a major source of software.

In the Netherlands, NTG's Bulletin Board FGBBS is dedicated to \TeX and friends. Its collection of \TeX related material is available to anyone who has installed a *modem* in his or her computer. You can call the FGBBS at +31 26 3217041. There are no additional costs and you are free to download as many files as you like. The speed at which files are transferred may not be very high, because of limitations in telephone line traffic.

9.5 File servers

To aid the archiving and retrieval of \TeX -related files, a TUG working group developed the Comprehensive \TeX Archive Network (CTAN). Each CTAN site has identical material, and maintains authoritative versions of its material. These collections are extensive; in particular, almost everything mentioned in this document is archived at the CTAN sites, even if not explicitly stated.

The participating hosts in the Comprehensive TeX Archive Network are:

`ftp.dante.de` (Deutschland)

- anonymous ftp, directory `/tex-archive (/pub/tex /pub/archive)`
- gopher on node `gopher.dante.de`
- e-mail via `ftpmail@dante.de`
- World Wide Web access on `http://www.dante.de/`
- Administrator: `ftpmaint@dante.de`

`ftp.tex.ac.uk` (England)

- anonymous ftp, directory `/tex-archive (/pub/tex /pub/archive)`
- gopher on node `gopher.tex.ac.uk`
- NFS mountable from `nfs.tex.ac.uk:/public/ctan/tex-archive`
- World Wide Web access on `http://www.tex.ac.uk/tex-archive`
- Administrator: `ctan-uk@tex.ac.uk`

`tug2.cs.umb.edu` (Massachusetts, USA)

- anonymous ftp, directory `/tex-archive (/pub/archive)`
- World Wide Web access on `http://tug2.cs.umb.edu/ctan/`
- Administrator: `ftpmaint@mail.tug.org`

In order to reduce network load, it is recommended that you use the Comprehensive TeX Archive Network (CTAN) host which is located in the closest network proximity to your site. Alternatively, you may wish to obtain a copy of the CTAN via CD-rom (see help/CTAN.cdrom for details).

Known mirrors of the CTAN reside on (alphabetically):

<code>cis.utovrm.it</code> (Italia)	<code>/TeX</code>
<code>ctan.math.mun.ca</code> (Newfoundland, Canada)	<code>/tex-archive</code>
<code>ctan.unsw.edu.au</code> (NSW, Australia)	<code>/tex-archive</code>
<code>dongpo.math.ncu.edu.tw</code> (Taiwan)	<code>/tex-archive</code>
<code>ftp.belnet.be</code> (Belgium)	<code>/packages/TeX</code>
<code>ftp.ccu.edu.tw</code> (Taiwan)	<code>/pub/tex</code>
<code>ftp.cdrom.com</code> (West coast, USA)	<code>/pub/tex/ctan</code>
<code>ftp.comp.hkbu.edu.hk</code> (Hong Kong)	<code>/pub/TeX/CTAN</code>
<code>ftp.cs.rmit.edu.au</code> (Australia)	<code>/tex-archive</code>
<code>ftp.cs.ruu.nl</code> (The Netherlands)	<code>/pub/tex-archive</code>
<code>ftp.cstug.cz</code> (The Czech Republic)	<code>/pub/tex/CTAN</code>
<code>ftp.duke.edu</code> (North Carolina, USA)	<code>/tex-archive</code>
<code>ftp.funet.fi</code> (Finland)	<code>/pub/TeX/CTAN</code>
<code>ftp.gwdg.de</code> (Deutschland)	<code>/pub/dante</code>
<code>ftp.heai.ie</code> (Ireland)	<code>/pub/ctan/tex</code>
<code>ftp.jussieu.fr</code> (France)	<code>/pub4/TeX/CTAN</code>
<code>ftp.kreonet.re.kr</code> (Korea)	<code>/pub/CTAN</code>
<code>ftp.loria.fr</code> (France)	<code>/pub/unix/tex/ctan</code>
<code>ftp.mpi-sb.mpg.de</code> (Deutschland)	<code>/pub/tex/mirror/ftp.dante.de</code>
<code>ftp.nada.kth.se</code> (Sweden)	<code>/pub/tex/ctan-mirror</code>
<code>ftp.oleane.net</code> (France)	<code>/pub/mirrors/CTAN/</code>
<code>ftp.rediris.es</code> (España)	<code>/mirror/tex-archive</code>
<code>ftp.rge.com</code> (New York, USA)	<code>/pub/tex</code>
<code>ftp.riken.go.jp</code> (Japan)	<code>/pub/tex-archive</code>
<code>ftp.tu-chemnitz.de</code> (Deutschland)	<code>/pub/tex</code>
<code>ftp.u-aizu.ac.jp</code> (Japan)	<code>/pub/tex/CTAN</code>
<code>ftp.uni-augsburg.de</code> (Deutschland)	<code>/tex-archive</code>
<code>ftp.uni-bielefeld.de</code> (Deutschland)	<code>/pub/tex</code>
<code>ftp.unina.it</code> (Italia)	<code>/pub/TeX</code>
<code>ftp.uni-stuttgart.de</code> (Deutschland)	<code>/tex-archive (/pub/tex)</code>
<code>ftp.univie.ac.at</code> (Österreich)	<code>/packages/tex</code>
<code>ftp.ut.ee</code> (Estonia)	<code>/tex-archive</code>
<code>ftpserver.nus.sg</code> (Singapore)	<code>/pub/zi/TeX</code>
<code>joshua.smcvt.edu</code> (Vermont, USA)	<code>/pub/tex</code>
<code>src.doc.ic.ac.uk</code> (England)	<code>/packages/tex/uk-tex</code>
<code>sunsite.auc.dk</code> (Denmark)	<code>/pub/tex/ctan</code>
<code>sunsite.cnlab-switch.ch</code> (Switzerland)	<code>/mirror/tex</code>
<code>sunsite.icm.edu.pl</code> (Poland)	<code>/pub/CTAN</code>
<code>sunsite.unc.edu</code> (North Carolina, USA)	<code>/pub/packages/TeX</code>
<code>wuarchive.wustl.edu</code> (Missouri, USA)	<code>/packages/TeX</code>

Known partial mirrors of the CTAN reside on (alphabetically):

ftp.adfa.oz.au (Australia)	/pub/tex/ctan
ftp.fcu.edu.tw (Taiwan)	/pub2/tex
ftp.germany.eu.net (Deutschland)	/pub/packages/TeX
ftp.gust.org.pl (Poland)	/pub/TeX
ftp.jaist.ac.jp (Japan)	/pub/TeX/tex-archive
ftp.uu.net (Virginia, USA)	/pub/text-processing/TeX
sunsite.dsi.unimi.it (Italia)	/pub/TeX
sunsite.snu.ac.kr (Korea)	/shortcut/CTAN

To find software at a CTAN site, use anonymous ftp to the host, and then execute the command ‘quote site index <search-term>’.

9.6 Learning T_EX

There are several ways to become proficient in T_EX or L^AT_EX. Some of the user groups mentioned above offer courses in T_EX or L^AT_EX at levels ranging from absolute beginners to advanced macro writers or METAFONT users.

Many books have been written on the subject, and there are some introductory texts available. M. Doob’s ‘*A Gentle Introduction to T_EX, a Manual for Self-study*’ is a fine article to start with. It can be obtained free of charge from many *file servers* and is also available on the second CD-rom. It is even available in several languages! See the directory \TEXCOURS for a complete list of available courses and introductions to T_EX on the CD-rom.

If you plan to use L^AT_EX you may want to read J. Warbrick’s ‘*Essential L^AT_EX*’, or G. Maltby’s ‘*An introduction to T_EX and friends*’. Both can be obtained from many file servers and the CD-rom. But after you have mastered the essentials of L^AT_EX you will soon feel the need for more documentation. This is provided by the L. Lamport’s ‘*L^AT_EX, A Document Preparation System*’, which is commercially available. The most complete book on L^AT_EX is ‘*The L^AT_EX Companion*’ by M. Goossens, F. Mittelbach and A. Samarin. We will mention three other recommendable books on L^AT_EX: Kopka, H. and P.W. Daly: ‘*A guide to L^AT_EX*’, Kopka’s ‘*L^AT_EX: Erweiterungsmöglichkeiten*’ in German, and R. Seroul and S. Levy’s ‘*A Beginner’s Book of T_EX*’.

Advanced T_EX and L^AT_EX users will often write their own macros and need much more insight. Their reference manual is ‘*The T_EXbook*’ by D. Knuth, the author of T_EX. This is the most comprehensive work on T_EX, but some consider it not very easy to read and understand for beginners. Another commendable book on T_EX is V. Eijkhout’s ‘*T_EX by Topic*’, which is not for the novice, but for users with basic understanding of T_EX, who want to explore the full potential of T_EX. S. von Bechtolsheim’s multi-volume opus ‘*T_EX in Practice*’ is also a good choice.

In the bibliography you will find several more references to books about T_EX.

Bibliography

- [1] P.W. Abrahams, K.A. Hargreaves, and K. Berry. *T_EX for the impatient*. Addison-Wesley, 1990.
- [2] Adobe Systems Incorporated. *PostScript Language Tutorial and Cookbook*. Addison-Wesley, 1985.
- [3] Adobe Systems Incorporated. *PostScript Language Reference Manual*. Addison-Wesley, 1990.
- [4] S. von Bechtolsheim. *T_EX in Practice: 1. Basics*. Springer-Verlag, 1993.
- [5] S. von Bechtolsheim. *T_EX in Practice: 2. Paragraphs, Math and Fonts*. Springer-Verlag, 1993.
- [6] S. von Bechtolsheim. *T_EX in Practice: 3. Tokens, Macros*. Springer-Verlag, 1993.
- [7] S. von Bechtolsheim. *T_EX in Practice: 4. Output Routines, Tables*. Springer-Verlag, 1993.
- [8] N.H.F. Beebe. Bibliography prettyprinting and syntax checking. *TUGBoat* 14, no. 4, pages 395–419, 1993.
- [9] K. Berry. Filename for fonts. *TUGBoat* 11, no. 4, pages 517–520, 1990.
- [10] P. Bloemen. Shells for T_EX. *MAPS*, 93.2, pages 155–159, 1993.
- [11] A. Borde. *T_EX by Example, a beginner's guide*. Academic Press Professional, 1992.
- [12] J. Braams. Babel, a multilingual style-option system for use with L^AT_EX's standard document styles. *TUGBoat* 12, no. 2, pages 291–301, 1991.
- [13] R. de Bruin, C.G. van der Laan, and J.R. Luyten. *Publiceren met L^AT_EX*. Rekencentrum Universiteit Groningen, 1988.
- [14] P. Chen and M.A. Harrison. *Index Preparation and Processing*. available from public file-servers, 1990.
- [15] T. Darrel. *PSFIG 1.8 users guide*. available from public file-servers, 1992.
- [16] W. Dol. The ease of including graphics in T_EX documents using 4T_EX. *MAPS*, 93.2, pages 171–176, 1993.
- [17] W. Dol. Frequently asked questions about 4T_EX. *MAPS*, 93.2, pages 128–130, 1993.
- [18] W. Dol. Some notes about T_EX and MS-Windows. *MAPS*, 93.2, pages 160–162, 1993.
- [19] W. Dol, E.H.M. Frambach, and M.H. van der Vlerk. 4T_EX, a T_EX workbench for ms-dos pc's. *MAPS*, 93.1, pages 53–56, 1993.
- [20] M. Doob. *A Gentle Introduction to T_EX, A Manual for Self-study*. Addison-Wesley, 1984.
- [21] E. Doron. BibDB, an Interactive BibT_EX Bibliography Manager. available from public file-servers, 1994.
- [22] V. Eijkhout. *T_EX by Topic*. Addison-Wesley, 1991.
- [23] V. Eijkhout. *Lollipop Unwrapped*. available from public file-servers, 1992.
- [24] R. Furuta, J. Scofield, and A. Shaw. Document formatting systems: Survey, concepts, and issues. *Computing Surveys*, 14, no. 3, pages 417–472, 1982.
- [25] M. Goossens. Postscript en L^AT_EX, de komplementariteit in de praktijk. *MAPS*, 93.1, pages 101–112, 1993.
- [26] M. Goossens, R. Rahtz, and F. Mittelbach. *The L^AT_EX Companion*. Addison-Wesley, 1994.
- [27] M. Goossens, R. Rahtz, and F. Mittelbach. *The L^AT_EX Graphics Companion*. Addison-Wesley, 1997.
- [28] G.D. Greenwade. The comprehensive T_EX archive network (ctan). *TUGBoat* 14, no. 3, pages 342–351, 1993.

- [29] D. Henderson. Introduction to METAFONT. TUGBoat 10, no. 4, pages 467–480, 1989.
- [30] J.D. Hobby. The Metapost System. available from public file-servers, 1995.
- [31] A. Hoenig. An introduction to \TeX for new users. MAPS Nederlandstalige \TeX Gebruikersgroep, pages 91–96, 1991.
- [32] B.K.P. Horn. complete bitmap-free \TeX packages! Y&Y, 1993.
- [33] G. Horn. \TeX cad Version 2.8. available from public file-servers, 1990.
- [34] N. Kempson. An Implementation of Bib \TeX for MS-DOS. available from public file-servers, 1989.
- [35] D.E. Knuth. The \TeX book. Addison-Wesley, 1984.
- [36] D.E. Knuth. A course on METAFONT programming. TUGBoat 5, no. 2, pages 105–118, 1984.
- [37] D.E. Knuth. The METAFONTbook. Addison-Wesley, 1986.
- [38] H. Kopka. \LaTeX : Erweiterungsmöglichkeiten mit einer Einführung in METAFONT. Addison-Wesley, 1990.
- [39] H. Kopka. \LaTeX , eine Einführung. Addison-Wesley, 4th revised edition, 1993.
- [40] H. Kopka and P.W. Daly. A guide to \LaTeX . Addison-Wesley, 1993.
- [41] C.G. van der Laan. Publishing with \TeX —The BLUes book. available from public file-servers, 1995.
- [42] L. Lamport. MakeIndex: An Index Processor for \LaTeX . available from public file-servers, 1987.
- [43] L. Lamport. \LaTeX , a Document Preparation System (updated for \LaTeX 2 ϵ). Addison-Wesley, 1994.
- [44] S. Levy. A Beginner’s Book of \TeX . Springer Verlag, 1991.
- [45] E. Mattes. How to run \LaTeX with $\text{EM}\TeX$. available from public file-servers, 1989.
- [46] E. Mattes. Frequently asked questions about $\text{em}\TeX$. MAPS, 93.2, pages 126–127, 1993.
- [47] E. Mattes. $\text{EM}\TeX$ 1.4a manual. available from public file-servers, 1995.
- [48] E. Mattes. METAFONT 2.718 manual. available from public file-servers, 1995.
- [49] E. Mattes. FONTLIB 1.2b manual. available from public file-servers, 1995.
- [50] E. Mattes. MAKEDOT 1.2e manual. available from public file-servers, 1995.
- [51] E. Mattes. \TeX CHK manual. available from public file-servers, 1995.
- [52] E. Mattes. DVIDOT 1.6g manual. available from public file-servers, 1997.
- [53] E. Mattes. DVIDRV 1.6g manual. available from public file-servers, 1997.
- [54] E. Mattes and L. Lamport. Using \LaTeX with $\text{EM}\TeX$. available from public file-servers, 1989.
- [55] F. Mittelbach and C. Rowley. \LaTeX 2.09 \leftrightarrow \LaTeX 3. TUGboat 13, no. 1, pages 96–101, 1992.
- [56] F. Mittelbach and R. Schöpf. With \LaTeX into the nineties. TUGboat 10, no. 4, pages 681–690, 1989.
- [57] F. Mittelbach and R. Schöpf. Towards \LaTeX 3.0. TUGboat 12, no. 1, pages 74–79, 1991.
- [58] F. Mittelbach and R.M. Schöpf. The new font family selection—user interface to standard \LaTeX . TUGboat 11, no. 2, pages 297–305, 1990.
- [59] H. Partl. Layout-Änderungen mit \LaTeX . EDV-Zentrum der TU Wien, 1988.
- [60] O. Patashnik. Bib \TeX ing. available from public file-servers, 1988.
- [61] O. Patashnik. Designing Bib \TeX Styles. available from public file-servers, 1988.
- [62] G.C. Reid. PostScript Language Program Design. Addison-Wesley, 1988.
- [63] G.C. Reid. Thinking in PostScript. Addison-Wesley, 1990.
- [64] T. Rokicki. DVIPS: A \TeX Driver (5.58). available from public file-servers, 1997.
- [65] D. Salomon. Advanced \TeX course: Insights and Hindsight. NTG, Dutch \TeX usergroup, 1992.
- [66] D. Salomon. The Advanced \TeX book. Springer Verlag, 1995.
- [67] A.L. Samuel. First Grade \TeX , A Beginner’s \TeX Manual. Stanford Department of Computer Science, 1983.
- [68] A.L. Samuel. First Grade \TeX : A Beginner’s \TeX manual. \TeX User Group, 1985.
- [69] N. Schwartz. Introduction to \TeX . Addison-Wesley, 1990.
- [70] R. Seroul and S. Levy. A Beginner’s Book of \TeX . Springer Verlag, 1992.

- [71] R. Smedinga. Hoe met \LaTeX een boek kan worden gemaakt. MAPS Nederlandstalige \TeX Gebruikersgroep, pages 97–101, 1991.
- [72] American Mathematical Society. AMS- \LaTeX Version 1.1, User's Guide. American Mathematical Society, 1991.
- [73] American Mathematical Society. User's Guide to AMS- \TeX , Version 2.1. American Mathematical Society, 1991.
- [74] F. Sowa. Benutzershandbuch BM2FONT. available from public file-servers, 1991.
- [75] F. Sowa. Bitmaps and halftones with BM2FONT. TUGBoat 12, no 5, pages 534–538, 1991.
- [76] M. Spivak. The Joy of \TeX . American Mathematical Society, 1986.
- [77] The Chicago Manual of style. . University of Chicago Press, 13th edition, 1982.
- [78] G. Tobin. METAFONT for beginners. available from public file-servers, 198?
- [79] M. Urban. An Introduction to \LaTeX . \TeX User Group, 1986.
- [80] E. van Herwijnen. Practical SGML. Wolter-Kluwer Academic Publishers, 1994.
- [81] N. Walsh. Making \TeX work. O'Reilly, 1994.
- [82] J. Warbrick. Essential \LaTeX . available from public file-servers, 1991.
- [83] H. Wright. SGML frees information. Byte 17, no. 6, pages 279–286, 1992.

Index

- .afm, 64, 67, 92
- .ai, 115
- .aux, 101, 102
- .bib, 45
- .bmp, 47, 108, 110, 115
- .bst, 101
- .btm, 27
- .cut, 47, 110
- .emf, 48, 111
- .eps, 47, 61, 90, 112–114, 119
- .fli, 82
- .fmt, 74, 75, 77–79
- .gf, 61, 98
- .gif, 47, 110
- .hpg, 47, 112
- .hpp, 47, 112
- .idx, 104
- .iff, 47, 110
- .img, 47, 110
- .ind, 50, 104
- .jpg, 47, 111
- .lbm, 47, 110
- .lj, 44, 83
- .log, 39, 44, 75
- .mf, 59, 61, 98
- .mfj, 59
- .mod, 30
- .mp, 62
- .msp, 108, 120
- .pcl, 47, 120, 121
- .pcx, 47, 108–110, 112, 114, 120
- .pdf, 113, 115
- .pfa, 64, 92, 98
- .pfb, 64, 92, 97, 98
- .pfm, 64
- .pk, 43, 82, 96, 98, 110
- .plt, 47, 112
- .prd, 32
- .prn, 44
- .ps, 47, 92–94, 112–114, 119
- .set, 31, 32
- .tfm, 64, 67, 74, 87, 92, 93, 98, 110
- .tif, 47, 110, 115
- .vf, 64, 87, 92, 93
- .wmf, 48, 111, 115
- _4tex_, 40
- 2-up printing, 42, 82
- 4-up printing, 42
- 4DOS, 26, 27
- 4dospath, 32
- 4tex.btm, 33
- 4texpars.<os>, 35, 40
- 4texpath, 32
- 80x87 coprocessor, 81
- accents, 92, 105
- Adobe, 91, 92
- afm2tfm, 64, 65, 67
- alias, 29
- aliasfil, 40
- amSpell, 39, 72, 105, 119
- append output, 44
- AsT_EX, 125
- asc2tex, 119
- asc2wp, 119
- ascii, 25, 31, 76, 92
- ascii editor, 31, 39
- Aurora, 123
- autoexec.bat, 27, 76
- Babel, 77, 78
- backup, 40
- backup_path, 40
- batch file, 27, 32
- batchmode, 75
- bezier, 115
- bib files path, 46
- BibT_EX, 101
- bibclean, 119
- BibDB, 46, 72, 102, 103, 119
- bibinput, 46
- bibliography, 101, 102
- bibliographystyle, 101, 102
- bibstydir, 102

big T_EX, 73, 74, 77
 binary file, 43, 44
 bitmap, 47, 50, 82, 91, 93, 98, 108, 110–112, 119, 122
 block compile, 39, 72
 BM2FONT, 109, 110, 119
 bounding box, 49, 114, 119
 Breeze, 123
 btxfmt, 75, 77
 bulletin boards, 130

 capacity exceeded, 73
 capt*.prd, 32
 check spelling, 39
 chi2ltx, 119
 chi2tex, 119
 ChiWriter, 119
 CM fonts, 93
 codepage, 78, 79
 color printing, 90, 108
 command.com, 26, 27
 commercial software, 91, 97
 compile and view block, 72
 compile and view file, 72
 convert graphics, 112
 convert picture, 48
 coprocessor, 81
 CrLf, 119
 cross-referencing, 101
 CsTUG, 125
 CTAN, 130
 CyrTUG, 126

 Dante e.V., 5, 126
 deT_EX, 119
 default language, 40, 72
 deskjet, 81, 85, 86
 color, 81, 82
 device-dependency, 108
 dirpxl, 110
 DisplayWrite, 122
 distribution sets, 122
 dot-matrix, 81, 85, 86
 DPMI, 73
 dvi-driver, 81, 84–88, 107, 108
 dvi2tty, 81
 dvidjc, 81
 dvidot, 44, 81, 83
 dvidrv, 86
 dvidrvfonts, 98

 dvidrvinput, 88
 dvihplj, 44, 81, 83, 84, 86
 dviHps, 93
 dvips, 44, 81, 90, 92–94, 98
 dviscr, 41, 42, 44, 81, 83, 85, 86, 88, 90
 dviscrs, 41, 42, 44, 82, 88
 dvivgac, 81
 dviwin, 81

 e-mail, 125, 129
 Eddi4T_EX, 123
 EditPad, 123
 eePic, 115
 eePicEmu, 115
 Emacs, 123
 emacs, 76
 emlines, 115
 emtexed, 76
 emtexopt, 73, 76
 EMX, 73
 encapsulated PostScript, 90, 115, 116
 encapsulated postscript, 62
 epsf style, 90
 epson, 85–87
 Estonian User Group, 126

 FAQ, 125
 FGBBS, 130
 figures style, 113
 file servers, 130
 find error(s), 72
 font generation, 93, 98
 font memory, 41
 font metrics, 74, 82, 92, 98
 FontInst, 67, 69, 92
 fontlib, 55, 82, 97
 fontmap, 65
 format files, 31, 58, 75–77
 format input path, 57
 free software, 112, 119
 FTP, 132

 GFtoPK, 61, 98
 Ghostscript, 44, 63, 81, 89, 90, 112
 GNUplot, 115
 Graphic Workshop, 50, 111, 116, 122
 graphics package, 110
 graphicx package, 110
 grayscale, 83
 Greek T_EX Friends Group, 126
 GSview, 113

GUST, 126
 GUTenberg, 126
 GUTH, 127
 GWD Text Editor, 123

 histfile, 40
 HP2xx, 112, 120
 HPGL, 47, 112, 120
 htexfmt, 74, 75, 77–79
 html2lat, 120
 HTMLcon, 122
 huge T_EX, 39, 73, 74, 77–79
 hyper-dvi, 93
 hyperPS, 93
 hyphenation patterns, 58, 74, 75, 77, 78

 iT_EXnici, 127
 index file, 50
 index making, 50
 index.sty, 104
 IrfanView, 111
 ITALIC, 127

 job path, 59
 JPEG, 111
 JTUG, 127

 keepenv.set, 40

 LaMeX, 77
 language support, 58, 77
 laserjet, 81, 85, 86, 97, 98
 L^AT_EX picture environment, 115
 L^AT_EX3, 5
 L^AT_EXcad, 115
 L^AT_EX Help, 117
 L^AT_EX Mac, 117
 list of references, 46
 Lithuanian T_EX Users Group, 127
 logfile, 39, 44, 51, 76
 lplain, 77
 Lucida Bright, 97

 mailing lists, 129
 main memory, 41, 75
 MakeIndex, 50, 103
 matching symbol, 71
 MathTimes, 97
 Mayura Cad, 115
 memory settings, 73
 metafont, 59, 60, 98, 132

 Metapost, 62, 116
 MeX, 77
 MFtoEPS, 61, 62, 116
 midi2tex, 120
 mma2ltx, 122
 monitor type, 27
 Multi-Edit, 123
 mytexinput, 74

 network, 32, 33
 network printers, 32, 44
 network.btm, 32
 newlfont style, 79
 nfss, 79
 Nordic T_EX Users Group, 127
 NotGNU, 123
 Novell Netware, 74
 NTG, 128
 NTS, 5

 Olivetti, 83
 on-line help, 26
 options file, 40
 outline, 92

 page-dimensions, 85
 Paintshop Pro, 50, 108, 111, 116, 122
 paper size, 42
 parameter stack, 41
 pattern memory, 41
 Pedit, 123
 PFE, 29, 76, 123
 pfm2afm.exe, 64
 PicT_EX, 115
 plain, 75
 plotter, 47, 112
 pool size, 41, 75
 PostScript, 43, 47, 49, 62, 81, 89, 92, 98, 107,
 108, 112–114, 116
 preamble, 37
 print destination, 42, 43
 print options, 44
 printer port, 43
 printer type, 42–44, 48
 Prolix, 123
 PS encoding, 92
 ps2pk, 65, 97, 98, 120
 psfig style, 90, 113
 psfigurepath, 114
 psfonts.inf, 65, 98
 psfonts.map, 65, 92, 98

psnfss, 91
 QFig, 115
 QuickEdit, 123
 Redit, 123
 resident fonts, 91
 RSX, 73
 sample T_EX files, 72
 save size, 41, 75
 scalable fonts, 90, 91
 semantic nest size, 41, 75
 serial port, 43
 shareware, 121, 122
 shortcuts, 40
 SliT_EX, 77, 79
 \special, 107, 114
 spell-check, 39
 spelling, 105
 SPL_COM, 105
 SPL_ENV, 106
 splain, 77
 subdirectory search, 74
 subscribe to mailing list, 129
 Super NoteTab, 123
 super VGA, 83, 112
 syntax checking, 117
 TE, 123
 T_EX courses, 132
 T_EX documentation, 72
 T_EX encoding, 92
 tex-d-l, 129
 tex-l, 129
 tex-nl, 129
 texbib, 102
 T_EXcad, 115
 T_EXCeH, 128
 T_EXchk, 117
 texconfig, 93
 texdvi, 88
 texfiles, 40
 texfonts, 110
 texhax-l, 129
 texinput, 74, 103, 110
 texprint.prn, 44
 textfm, 74
 texuser.<os>, 84, 88, 102
 Tirant lo T_EX, 128
 TSE, 29, 39, 71, 76, 123
 TSE Junior, 39
 TSE junior, 39
 Tseng, 83
 TUG, 128
 Type1 fonts, 43, 98
 Type3 fonts, 98
 UK TUG, 128
 UKtex, 129
 URW fonts, 98
 user groups, 125
 VCPI, 73
 Vedit, 123
 VESA, 83
 view options, 44
 viewer type, 42, 49
 virtual fonts, 87, 92
 WinEdt, 123
 word list, 51
 Zeus, 123