

Introduction

This book originally accompanied a two-day course on using the \LaTeX typesetting system. It was extensively revised and updated for publication, so that it could be used for self-study as well as in the classroom. For those with sufficient prior knowledge of computing and authoring, it has also successfully been used as the basis for a 1-day intensive introductory course. It is aimed at users of Unix & GNU/Linux, including Apple Macintosh OS X, and Microsoft Windows systems, but it can be used with \LaTeX on any platform, including other Unix workstations, mainframes, Android and Apple smartphones, and even some older Personal Digital Assistant (PDA)s.

Who needs this book?

The course was originally designed for computer-literate but non-IT professionals in business, academic, and nonprofit organisations. You may be in a similar position, but you may also come from another background entirely; you may be a hobbyist, a school or college student, a home computer user or a volunteer worker, or you might just be interested in high-quality automated typesetting. However, it's likely that you have one or more of the following or similar objectives:

- producing consistent, typeset-quality formatting;
- formatting long *or* complex *or* highly-structured *or* repetitive *or* automatically-generated documents;
- saving time and effort by automating common tasks;
- gaining independence from expensive and restrictive proprietary hardware, software, or file formats;
- creating robust, durable documents which will survive changes in technology.

Skills needed

\LaTeX is a very easy system to learn, and requires no specialist knowledge to get started, although it's useful if you understand something about writing, formatting, and readability. However, you do need to be

completely familiar with using your computer, which means knowing the following topics thoroughly. Note that none of these is in any way specialist; they're all basic, fundamental, standard computer skills that everyone should know:

- 1.1.4–1.1.6,
1.3.1, 1.4.1–
1.4.2, **Using the mouse:** how to point and click with your mouse to select text and pick from a menu (or to use keyboard shortcuts to do the same)
- 2.3 **Handling files:** how to create, open, save, close, rename, copy, move, and delete files and folders (directories);
- 3.2.1.2 **Handling characters:** where to find all 95 of the printable `ASCII` characters on your keyboard and what they mean, and how to type accents and symbols, if you need them;
- 2.1.3 **Using an editor:** how to use a good plaintext editor⁵;
- 7.1.6
[7.3.1.6] **Downloading files:** how to use your Web browser and/or file transfer program to download and save files from the Internet;
- 2.3.8 **Unzip files:** how to uncompress and unwrap (unzip or detar) compressed files;
- 2.6 **Install software:** how to install software, both manually and using automated installers;
- 1.7 **RTFM:** how to read and follow instructions and how (and where) to ask for help.

If you don't know how to do these things yet, it's important to go and learn them first. Trying to become familiar with basic computer skills *at the same time* as learning `LaTeX` is not going to be as effective as doing them in the right order.

It is important to understand that these are *not* specialist skills — they are standard for anyone who uses a computer, and they form a fundamental part of the basic knowledge of computers. With the exception of software installation, they were included in the European Computer Driving Licence (ECDL) course: the relevant module and

⁵ *Not* a wordprocessor like *OpenOffice*, *Lotus Notes*, *Corel WordPerfect*, or *Microsoft Word*, and *not* a 'dumb' editor like *Apple TextEdit* or *Microsoft Notepad*.

section numbers of the [ECDL](#) syllabus are noted in parentheses or in the margin above (Kelly & O'Connor, 2005).

Objectives of this book

By the end of this book, you should be able to undertake the following tasks:

1. use your editor to create and maintain your documents;
2. use \LaTeX markup to identify your document structure and formatting requirements;
3. typeset \LaTeX documents, correct simple formatting errors, and display or print the results;
4. identify and use additional \LaTeX packages (using the Internet for downloading where necessary and installing them);
5. recognise the limitations of procedural markup systems and choose appropriate generic markup methods where appropriate.

Synopsis

The original course covered the following topics as separate sessions. Earlier versions of this document kept to this structure in the book as chapters, but the current version has moved Installation (originally chapter 1) to Appendix A and merged it with the details of configuration; and Typesetting, viewing, and printing (originally chapter 4) to a new Appendix; as the procedure in both cases has been so much simplified that the previous level of detail is no longer needed in most cases.

1. How to create \LaTeX documents (with a Quick-Start Guide for the impatient);
2. Basic document structures (the Document Class Declaration and its layout options; the `document` environment with sections and paragraphs);
3. Using packages and [CTAN](#) to adapt formatting to your needs;

4. Other document structures (lists, tables, figures, images, and verbatim text);
 5. Textual tools (footnotes, marginal notes, cross-references, indexes and glossaries, and bibliographic citations);
 6. Typographic considerations (white-space and typefaces; inline markup and font changes; extra font installation and automation);
 7. Programmability and automation (macros and modifying \LaTeX 's behaviour);
 8. Conversion and compatibility with other systems ([XML](#), *Word*, etc).
- A Where to get and how to install \LaTeX (using the \TeX Users Group's \TeX Collection DVD);
 - B How to install new fonts;
 - C Typesetting, viewing, and printing (largely obsolete now that editors are better integrated with viewers and printers);
 - D User groups and the benefits of membership;
 - E The [ASCII](#) character set;
 - F The [GNU Free Documentation License](#).

I have made a few other changes in the transition to printed and online form, but the basic structure is the same, and the document functions as a workbook for the course as well as a standalone self-teaching guide.

Where's the math?

Please understand that this document *does not cover* mathematical typesetting, complex tabular material, the design of large-scale macros and document classes, or the finer points of typography or typographic design, although it does refer to these topics in passing on a few occasions.

There are several other guides, introductions, and 'get-started' documents on the Web and on [CTAN](#) which cover these topics and more in great detail. Among the more popular are:

- *Getting Started*, where all beginners should start;
- *A (Not So) Short Introduction to L^AT_EX 2_ε* is a good beginner's tutorial;
- *Gentle Intro* is a classic tutorial on Plain T_EX (not L^AT_EX);
- *Imported graphics* shows you how to do (almost) anything with graphics: side-by-side, rotated, etc;
- *Short Math Guide* gets you started with the American Math Society's extensions;
- *The T_EX Symbol List* shows over 2,500 symbols available.

This list was taken from the [CTAN](#) search page. There are also lots of books published about T_EX and L^AT_EX: the most important of these for users of this document are listed at the end of the [on page xi](#).

Availability of L^AT_EX systems

The standard implementations of T_EX and related systems are in the T_EX Collection, published annually on DVD by the [T_EX Users Group](#). These are all derived from Knuth's master versions, and adapted for all major platforms (Unix & GNU/Linux, including Apple Mac OS X; and Microsoft *Windows*). The DVD is sent free to all [TUG](#) members and can be obtained from your local user group. You can also download the ISO image file from [CTAN](#) to burn your own copy.

Commercial implementations are listed in 'Commercial implementations' [on page xxviii](#).

Systems included on the T_EX Collection DVD

ProT_EXt (Windows): This is the popular *MiK_TE_X* implementation plus the *T_EXstudio* editor.

MacT_EX (OS X): This is *T_EX Live* plus the *T_EXshop* editor (the Mac's built-in *Preview* is used for the [WYSIWYG](#) display).

T_EX Live (Unix & GNU/Linux, including Apple Macintosh OS X, and Windows): Generic *T_EX Live* for systems without a built-in package distribution.

Users of Red Hat/CentOS, Debian/Ubuntu, and derivative systems should normally use their system-supplied package manager (eg *yum*, *rpm*, or *apt-get*; or *Synaptic* or *Software Installer* etc) to install the RPM or DEB packaged versions provided in their supplier's repositories, as these provide tighter integration with other software such as editors and viewers; but the generic *T_EX Live* can be used where the absolute latest bleeding-edge features are required (in which case any system-installed version SHOULD be removed first to avoid conflicts and unresolved dependencies — see the notes in Appendix 1 starting [on page 183](#)).

Because the T_EX program (the internal 'engine' which does the actual typesetting) is independent of any other software, it doesn't have its own editor like a wordprocessor does. Instead, you get to choose whichever editor you prefer: there are lots available, and you can switch between them to find one you like: see 'Graphical interfaces (editors)' and [section A.3 on page 216](#) for details.

Graphical interfaces (editors)

Most users run L^AT_EX with a graphical plaintext editor which has a toolbar and menus like other windowing applications. These usually include all the common formatting features of L^AT_EX plus writing tools like spellchecking, thesaurus, indexing, and bibliographic citation, and generally all work in a very similar way. Text-only interfaces are available for use on servers and automated production systems (see 'Command-line interfaces' [on page xxvii](#)).

The Windows and Mac systems described in 'Availability of L^AT_EX systems' [on page xxv](#) come with a recommended editor (*T_EXStudio* and *T_EXshop* respectively), but you can install any other suitable editor you prefer (see [section A.3 on page 216](#)). The Unix & GNU/Linux distribution does not install any editor because these systems usually have their own software repositories with suitable editors already available for installation, such as *Emacs*, *vi*, *T_EXStudio*, or *Kile*.

Fully synchronous typographic interfaces (editors) are available as commercial products: see the list item 'Synchronous typographic displays' [on page xxviii](#).

Command-line interfaces

While you would use a graphical interface to *set up* an automated system like a web server or e-commerce environment, it is useless where systems have to run in the background, unattended, with no human to click on buttons. In fact, the \TeX typesetting engine is a Command-Line Interface (CLI) program, which can be used from any script or console or 'Command' window. You can type the command `xelatex` followed by the name of your document file (see [Figure C.2 on page 238](#) for an example).

Commands like these let you run \LaTeX in an automated or scripted environment like a Common Gateway Interface (CGI) script on a web server or a batch file on a document publishing system. All the popular distributions for all systems, both free and commercial, include this CLI interface as standard.

WYSIWYG displays

\LaTeX usually displays your typeset results in a separate window such as a PDF viewer, updated automatically every time the document is retypeset, because the typesetting is kept separate from the editing. This is called an 'asynchronous' display. Some systems, however, can format the typesetting while you type each character, like a wordprocessor, although at the expense of some flexibility. These are called 'synchronous' displays.

Asynchronous typographic displays: The WYSIWYG display is updated when the document is reprocessed, rather than *while* you are still typing, as it would with a wordprocessor. To update the display, just click on the button which reformats the document. You are probably already familiar with this idea if you have used HTML, where you reload the page in a browser to see it, or if you have used a spreadsheet, where the ReCalc button (F9) does something similar.

\TeX systems typeset the whole document at one go, including all indexing, cross-references, tables of contents, bibliographic citations, and the placement of figures and tables. \TeX also formats whole paragraphs at a time, rather than line-by-line as wordprocessors do, in order to get the quality of spacing,

hyphenation, and justification right. This approach makes it much faster than a wordprocessor in dealing with typical complex documents, as it can be done without holding the whole document in memory.

Synchronous typographic displays: The WYSIWYG display is the editing window, and it updates while you type, like a wordprocessor. Some popular examples are *L_AT_EX* (all platforms), *Textures* (Mac), *BaKoMa T_EX* (Windows), and *Scientific Word* (Windows) (see [Table 1 on page xxix](#)).

With a synchronous display you get Instant Textual Gratification™, but like a wordprocessor, your level of control is restricted to that of the system you use, which cannot provide access to everything that *L_AT_EX* can do. For complete control of complex material you may still need to use separate editing and display windows as for asynchronous implementations.

Near-synchronous displays: There are a few systems for very-close-to-synchronous WYSIWYG display. These include Jonathan Fine's *Instant Preview* with the *T_EX* daemon, and David Kastrup's [preview-latex](#) package for embedding typographic fragments from the typeset display back into the editor window.

What You See Is What You Get (WYSIWYG) refers to the accuracy of the typographical display. Most modern ones are pretty good, given the fact that your screen is probably only a fraction of the accuracy of your printer — between 96 dots per inch (DPI) on an old desktop screen and around 300 DPI on some handhelds; as opposed to 600 DPI on your printer, or 1200 DPI or more in photo-quality, and 3,600 DPI or more on laser-driven phototypesetters.

Commercial implementations

Although the *T_EX* Collection is available free of charge, there are several excellent commercial implementations of *T_EX* and *L_AT_EX* listed in [Table 1 on page xxix](#), with enhanced support and additional features. These companies, founders, and staff have been good friends of the *T_EX* and

Table 1: Popular commercial distributions of T_EX systems

Product	Platform	Company	URI
PCT _E X	MS-Win	Personal T _E X, Inc	www.pctex.com/
BaKoMa T _E X	MS-Win	Basil K Malyshev	www.bakoma-tex.com/
TrueT _E X	MS-Win	True T _E X	www.truetex.com/
Textures	Mac	Blue Sky Research	www.bluesky.com/
Scientific Word	MS-Win	Mackichan Software	www.mackichan.com/
VT _E X	MS-Win, Linux, OS/2	MicroPress, Inc	www.micropress-inc.com/


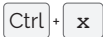




L^AT_EX communities for many years: if these versions are of benefit to you, I urge you to support them and buy their products.⁶

Symbols and conventions

There are several typographic conventions about how you represent computer-related material in print which are shown in Table 2 on page xxx. Typed commands, keywords, examples of input, and related text are in a fixed-width (monospace) font, like a typewriter, because that's how program code is usually displayed and edited (this also helps avoid ambiguities, as explained in section 4.7.1.1 on page 101). Special values, like numeric quantities represented by a name or symbol, are in italics, as in mathematics. Terms or references to products, programs, packages, and other components of L^AT_EX have their own typographic form. Finally there are some symbols like keyboard keys and menus, which are shown graphically.

⁶ Y&Y, Inc, who produced a T_EX distribution for many years, have ceased trading. Some of their add-on fonts are now being distributed by the T_EX Users Group (see Appendix 4 starting on page 253), or have been replaced by Open Source implementations, and there is a mailing list at the TUG web site for the support of former Y&Y users.

Table 2: Typographic notations used in this document

Notation	Meaning
MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL	These keywords have a specific and possibly mandatory meaning when shown in THIS STYLE (usually small capitals), according to the specification in RFC 2119 (Bradner, 1997).
<code>\command</code>	L ^A T _E X commands (control sequences) that you type which perform an action like <code>\clearpage</code> , or identify your text like <code>\footnote{...}</code>
<code>\length</code>	Control sequences which store a dimension (a measurement in units), like <code>\textwidth</code>
<i>counter</i>	Values used for counting (whole numbers only), like <i>section</i>
term	The defining instance of a new technical term
environment	A L ^A T _E X formatting or identification environment, like <code>quotation</code>
class	A L ^A T _E X document class (available from CTAN), like memoir
package	A L ^A T _E X add-on package (available from CTAN), like <code>footmisc</code>
<i>product</i>	A program or product name
<i>typewriter type</i>	Literal examples
<u>value</u>	Mnemonic examples of things you must type, where you have to supply real-life values of your own, like <code>\author{<u>your name</u>}</code> means you must replace <u>your name</u> with your own real name.
	A specific key on your keyboard
	Two keys pressed together, not separately
	Two keys pressed one after another
	An on-screen button to click
	A drop-down menu with items to select
	A normal space, just so it's visible

Production note

This document is written and maintained in `XML`, using a customized version of the *DocBook 5* DTD. Conversions are made to `HTML` and `LaTeX` using `XSLT2` scripts and Michael Kay's *Saxon* processor.

The complete source, with all ancillary files, is available online at www.ctan.org/tex-archive/info/beginlatex/src/. If you want to try processing it yourself you will need *Java* and *Saxon* in addition to a full installation of `LaTeX`.

This document is published under the terms and conditions of the `GNU Free Documentation License`. Details are in Appendix 6 starting on page 261.

