# 1 The TEX Family

## 1.1 A Little History

TEX is a typesetting system with a long history and a bright future. TEX is, like `HTML` and `XML` a markup language. As we have seen in earlier chapters plain text with commands (tags if you prefer) interspersed make up a TEX source document. TEX was created before the invention of the personal computer. The first versions ran on text terminals attached to mainframes and minicomputers.

TEX was created by the eminent mathematician and computer theorist Donald Knuth. Although hundreds have participated in the extension of the system the original work was the product of a single individual, and it stands today as a remarkable personal achievement.

When TEX was created there were no type standards like *PostScript* or *TrueType*. So Knuth designed his own computer type system called *Metafont*. With no standardized way of converting a document to printable pages he invented a device independent intermediate format `dvi` with the idea that any vendor of a specific printing system could write a conversion program.

## Free Publishing Software

He also designed the typesetting program so that it contained only simple commands (called *primitives*) with the idea that macro commands (combinations of primitives) could be created to perform typesetting tasks. He provided a mechanism whereby groupings of these macro commands could be in effect preprocessed to speed up the translation of source text to the dvi format. These preprocessed bundles are called formats. He wrote the first such bundle himself as an example to others (plain.tex.) If this sounds suspiciously like computer programming it is because that is what TeX is—a specialized programming language. Instead of the result of a TeX pass being an executable program the result is a printable document. But the basic concepts are identical. Programming language compilers are themselves programs. That means they are written in a programming language and compiled before they can be used. Most compilers in the 1970's were written in a very low level language called an Assembler. Knuth invented his own language for creating the TeX program, called `Web`.

TeX was written by a mathematician and so it is not surprising that it has excellent facilities for expressing mathematical formulae.No other typesetting system has quite matched it in this arena. But it can be used for almost any printing task, from novels to arm bands.

## 1.2 The people of TeX

Although Knuth is preeminent as the creator of TeX the development of the system has been carried forward by a small army of volunteers, most of whom are academics. They do this work because they can exercise their skills without reporting to anyone except the TeX community as a whole. This author cannot hope to list them all with their accomplishments. Here are a few names, in alphabetic order, of people who matter in the world of TeX. They include macro writers, archivists, authors of books, creators of fonts, and programmers who have provided ancillary packages like bibtex and makeindex:

David Arseneau
Nelson Beebe
Karl Berry
David Carlisle
Pehong Chen
Michael Doob
Robin Fairbairn
Denis Girou
Hans Hagen
Taco Hoekwater
Leslie Lamport
Sebastian Rahtz
Hàn Thề Thành
Timothy Van Zandt

Why is this list important? If you post a question on

/`comp.text.tex`

or one of the many mailing lists dealing with TeX then you know that these names represent the voice of experience. In fact you may be corresponding with the author of the package you are attempting to use! You don't get that kind of personal service from commercial software houses. Just remember to be polite. Flames are not welcome. Individuals can be grumpy at times, but don't respond in kind.

`The major \TeX variants are included in the distributions (miktex, tetex) available from www.miktex.org or www.tug.org respectively.`

## 1.3 TeX Variants

The original TeX program was designed with variations in mind. As mentioned before Knuth got the ball rolling with an initial format package called plain tex. Leslie Lamport created a much more structured and abstract approach through a package called LaTeX which has become the *de facto* standard implementation for most users of TeX. But there are many other implementations as well as auxiliary packages that run with more than one implementation. An early task for you the user is to study these variants and determine which meets your needs and current state of knowledge.

A special feature of TeX is that the combination of the basic kernel and the major macro package is executed by entering the name of the major package. The table below shows the major variants and how they are in fact called on the command line. Here are a few notes on the table:

- Variants above the horizontal line produce *dvi* output only. Variants below it can produce either *dvi* or *pdf* output.

## The TEX Family—TEX Variants

TEX Variants and commands.

| **Package** | **command** | **Comments on Variant** |
|---|---|---|
| Plain TEX | `tex` | Unstructured, terse, stable |
| LATEX | `latex` | Structured, verbose, popular |
| Texsis | `texsis` | Plain plus tables etc. |
| etex | `etex` | Plain plus right-to left |
| pdftex | `pdftex` | Plain plus interactivity |
| pdfLATEX | `pdflatex` | LaTeX plus interactivity |
| pdfetex | `pdfetex` | Combines pdftex, etex |
| pdfeLATEX | `pdfelatex` | Combines pdfLATEX, etex |
| ConTEXt | `texexec` | Verbose, full-featured, new |

**Table 1.1** TEX variants and calling commands