INTRODUCTION TO TEX AND TUG FOR NEW USERS

Ron Whitney

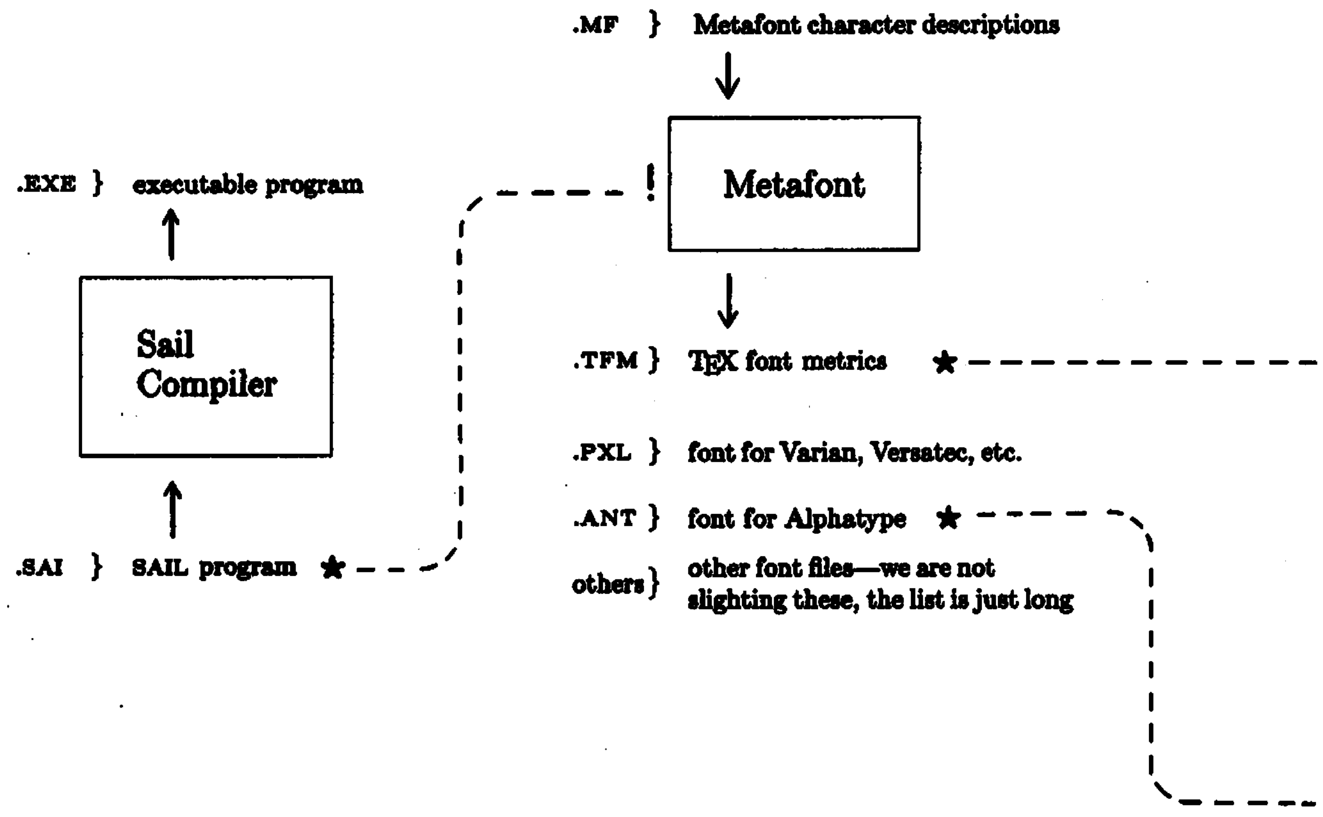
The schematic diagram on the following pages gives a picture of the files and programs involved in the TEX typesetting system. The names used are the standard extensions seen on DEC10's and 20's, but they are sufficiently mnemonic to be representative of all systems. Movement along arrows shows the general flow of files through programs. Particular files and programs within one flow may be the result of particular input from another, and these connections are indicated by paths (- - - -) from input (*) to output (!). A prospective user should be aware that this scheme is somewhat different than the pre-TEX82 setup and that the METAFONT side will change in the next year or so when that program is written in WEB. Otherwise, the following generalities should be of some help in discerning what one needs to start a system and what jargon will help in asking questions.

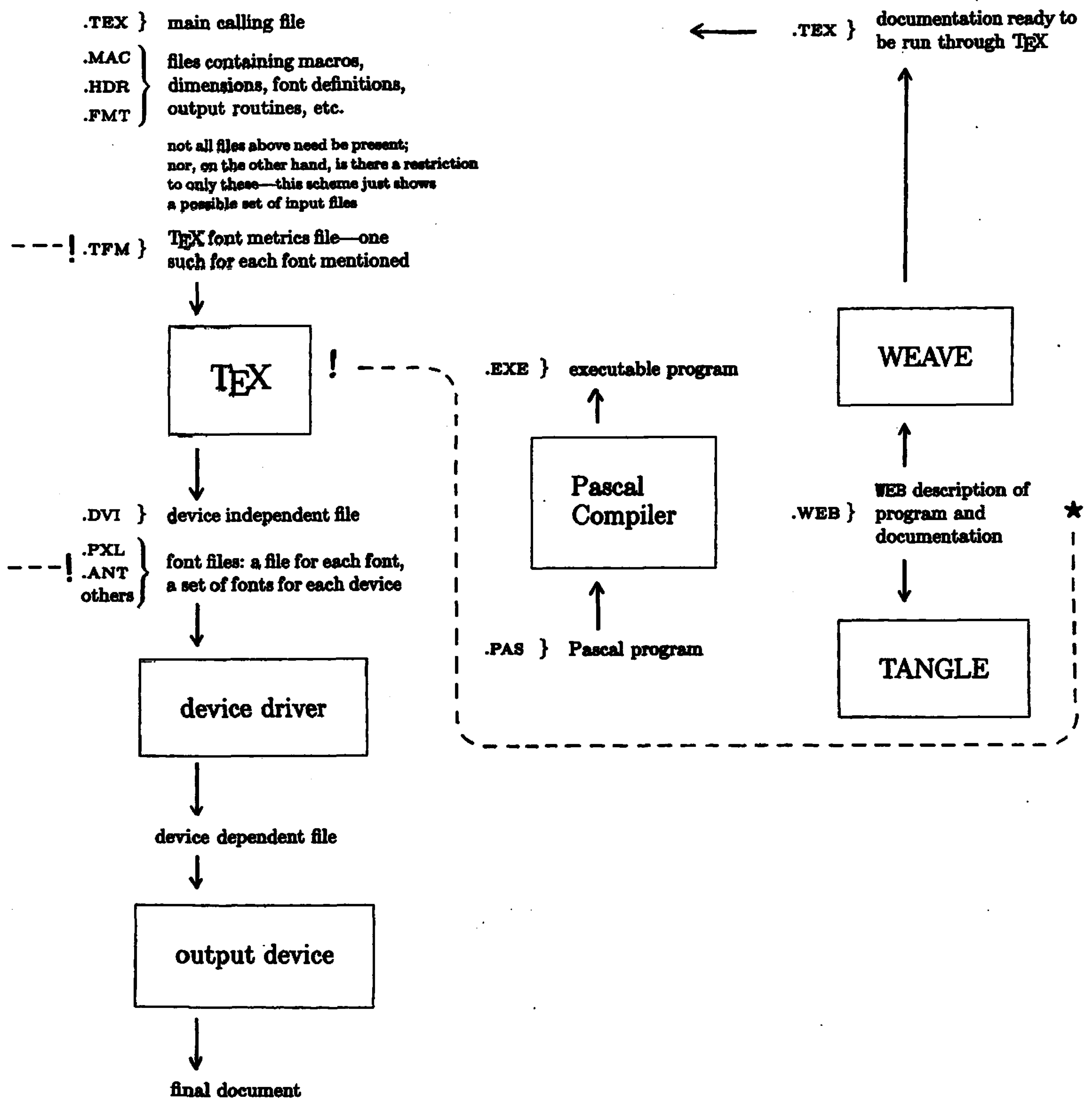
The column containing the TEX program comprises a minimal set of programs and files needed to produce a typeset document. The files used as input to TEX contain the text to be set as well as the instructions which determine where characters are placed on a page. The most comprehensive reference for the TEX language is, of course, the TEX manual (*TEX and METAFONT*, Digital Press, 1979; the TEX82 manual is in preparation and will be published by Addison-Wesley sometime soon). Raw TEX provides complete control over the final page at the possible expense of excruciating detail in specification. If the user wishes to follow the style of another author (either to cut down on one's own work or to produce a paper with a certain format), it is possible to encode that style in "macros" whose definitions can be input from .MAC, .HDR, and .FMT files. Each macro may call many TEX primitives, so a well constructed set of ancillary macro, header and format files will create an environment where (typographically) complicated papers can be put out fairly easily. We will not go into the differences traditionally associated with the "kinds" of input files (as with other specifics, one should consult issues of TUGboat and the TEX manual for help), but simply note that this is an area which deserves great attention once TEX is up and running. It is also worth noting in this context that AMS-TEX and other macro packages are *not* alternative forms of TEX, but are auxiliary files input with other matter to be run under TEX.

When a proper collection of statements is fed to TEX, the latter produces a .DVI (device independent) file. One of the advantages of TEX is its ability to transcend hardware peculiarities. A consequence of this is that if authors set their own papers with TEX on an available proofing device and send all their input files to a publishing house which runs TEX, they can be assured that the publisher's typesetters will break lines and pages in exactly the same way. This could significantly reduce both editing time and exasperating communication. TUGboat has already taken several .TEX files from authors and reproduced them with a minimum of editorial fiddling.

In order to achieve "device independence", TEX describes the set page in an absolute way: characters are only boxes of a certain height, depth, and width, and distances between characters are given in terms of printer's points (72.27 to the inch). Given this invisible ink description of a page, it is left to each typesetter to fill in the details of the raster pattern or whatever else specifies a character. Thus, each site must construct, or otherwise obtain, a program which combines the instructions in the .DVI file with information about the local fonts to produce a "painted" page for the output device. Typically, this point in the sequence has involved the most work for implementers simply because a programmer must digest something about both TEX's output and the internal workings of the printing device in order to write the driver. It may also be desirable to drive the output device with a microprocessor separate from the machine which runs TEX. Needless to say, an effort at bringing TEX up will be greatly aided by finding a site where TEX is running with the same operating system and output device. The TUG membership list contains information on architectures and output devices currently supporting TEX.

Fonts are also an important consideration. Even if an output device comes with reams of beautiful characters, users will need to make sure that these fonts can be connected to TEX. In order to construct the .DVI file, TEX needs to know certain information about the characters it is setting. Specifically, it needs to know the width, height above the baseline, depth below the baseline and ligature and kerning information for each symbol. This is exactly the information in the .TFM (for TEX Font Metrics) files created by METAFONT. Currently the easiest way to get started is to use fonts generated by METAFONT since .TFM's are available and the fonts can be made, in some sense, device independent. Fonts for two devices of different resolution


.MF } Metafont character descriptions
Metafont
.TFM } TEX font metrics ★
.PXL } font for Varian, Versatec, etc.
.ANT } font for Alphatype ★
others } other font files—we are not slighting these, the list is just long
.EXE } executable program
Sail Compiler
.SAI } SAIL program ★


.TEX } main calling file
.MAC .HDR .FMT } files containing macros, dimensions, font definitions, output routines, etc.
not all files above need be present; nor, on the other hand, is there a restriction to only these—this scheme just shows a possible set of input files
---! .TFM } TeX font metrics file—one such for each font mentioned
TEX
!
.DVI } device independent file
---! .PXL .ANT others } font files: a file for each font, a set of fonts for each device
device driver
device dependent file
output device
final document
.EXE } executable program
Pascal Compiler
.PAS } Pascal program
.TEX } documentation ready to be run through TeX
WEAVE
.WEB } WEB description of program and documentation
★
TANGLE

require two separate runs of METAFONT, but, if the source files have been properly constructed, identical .TFM's are produced. TEX will view such a pair of fonts as identical because it looks only at the .TFM file, not the font files. The drawback so far with METAFONT fonts has been the lack of a wide range of styles and sizes. If non-METAFONT fonts are to be used, .TFM's will have to be constructed by hand. Access to the innards of the fonts will also have to be obtained to write the device driver and not all typesetting companies are excited about making such information available. As TEX becomes more widely available and typesetters see a growing market for their machines, it is anticipated that more of them will open their font libraries to TEX users.

The discussion so far has encompassed the TEX-METAFONT system before 1982. Some out-of-date points have not been discussed here (such as the difference between .TFX and .TFM files), but, for those curious about or in need of such things, we suggest reading back issues of TUGboat.

TEX82 resembles TEX in its basic box and glue approach to page make-up, but improves upon its sibling by adding new features to the language and more space to store information. Documentation on the changes and a full description of the new language can be obtained in TUGboat and the new TEX manual when it appears. In a certain way, however, the younger TEX is hardly first kin to the older: TEX82 is written in the new WEB language whereas the original TEX was an exercise in SAIL code. WEB is a system of structured documentation which combines the features of both a programming language (in this case, Pascal) and a document formatter (in this case, TEX). Writing a program in WEB, the programmer is able to display and expose the structures of his or her algorithms in ways which are not possible with commented Pascal code. A .WEB file generates both a running Pascal program (passing a .WEB file through the program TANGLE produces a .PAS file of properly written Pascal code) and a nicely formatted description of whatever structures the programmer thinks are essential to the understanding of that program (passing a .WEB file through the program WEAVE produces a properly written .TEX file which can be run through TEX). The "essential structures" need not be Pascal subroutines and the overall approach may be top-down, bottom-up, or some combination. Since the same source produces both program and documentation, there are fewer occasions when the two are mismatched, and reliable, clear programs are easier to write. Of course, the first major project written in WEB was TEX82, but the uses are much more general.

Finally, a note on reference where to seek help. *TEX and METAFONT* co Knuth's 1978 Gibbs Lecture as well as the TE METAFONT manuals. It can be obtained from

Digital Press
Dept. AMS
Educational Services
Digital Equipment Corporation
12A Esquire Road
North Billerica, MA 01862

The new TEX manual will contain a c plete description of the TEX language as well a the WOVEN and TEX'd output of the WEB sources the program. This manual will be published by Addison-Wesley and should be available early in 1983. Knuth's next project is to be the WEB version of METAFONT and this will also be accompanied by a new manual. The third volume of Knuth's series on typesetting and type design will contain a description of the work he has done on the Computer Modern family of fonts. Additional papers and research articles on the algorithms involved in TEX can be obtained from:

Computer Science Department
Stanford University
Palo Alto, CA 94305
attention: Dawn Yolton

To obtain information about WEB and a tape containing TANGLE, WEAVE and other related programs, write to Professor Arthur Samuel at Stanford University. In fact, Arthur is the man to contact for many of your Stanford needs, be they fonts or programs. Contact one of us here at the AMS for information about AMS-TEX and AMS fonts, and consult your TUGboats for more names and resources. Finally, please share your experiences and ideas about anything connected with TEX or METAFONT by writing to TUGboat. The community welcomes your comments and suggestions.