

# TEX NOTES

## Automatic Page Sizing

Picture a 50-page manual with 50 topics, each topic needing about a page. How could I automatically get a page-filling combination of margin size, magnification, openup, etc., so that *each designated topic will fill its page*? I'm picturing a camera's "program" mode, where the designer decided in advance how to balance the aperture-size/shutter-speed so that the light, whatever its quantity, would always "fill" the film. Just as each photo has different exposure parameters, each page of the manual could be slightly different. The only form of automatic *page* control for TEX seems to be through inter-word spaces and hyphenations, but not through line length, margins, and interline spaces. Of course, page-filling, whether through a program or through trial and error, can work only within certain quantity limits. As with the camera, it would be good (but not necessary) if there were a means of over-riding the automatically-chosen parameters, so that the user could, for instance, put priority on either magnification or margin. It would be nice if appropriate horizontal and vertical offsets were included in the program, and if the user could direct that the inner margins be so much larger or smaller than the outer.

I look forward to reading — and seeing — the reply.

Alan Auerbach  
Wilfrid Laurier University

**Editor's reply:** On page 59 of *The TEXbook*, it is decreed "You cannot apply two different magnifications to the same document." So `\magnification` is not a possible variable.

It should be possible to simulate the effect of magnification by defining suitable typesize groups in the manner of `\tenpoint`, etc. (*The TEXbook*, page 414), perhaps naming them `\tenpointmaghalf`, `\tenpointmagone`, etc., for larger sizes where specific magnified fonts have been loaded explicitly, as `\font\tenmaghalfcmr=\cmr10 scaled \magstephalf`, etc. (The user should be careful to set appropriate values for baselineskips and the dimensions of strutboxes.)

Modifications to margin width, line length, and interline spacing can all be applied in a relatively straightforward manner to the text, while manipulation of top, bottom and side margins is probably most easily implemented through parameters in the output routine. The biggest challenge seems to be in deciding how much adjustment is needed to each page (after test-setting it using "standard" values), and recycling the data using the revised parameter values.

However, just because implementing a format of this sort might be possible doesn't mean that it's always a good idea. (After all, TEX aims to be a tool which will permit typographers to create documents of the highest quality when judged according to traditional standards.) Noticeable variations in spacing, size and character style can distract from the main business of a document, which is to deliver a message clearly without making the reader consciously aware of the document's appearance. If the extreme cases diverge too greatly

from the norm, then probably some different kind of solution is wanted.

Barbara Beeton

## Where to Find TEX File Descriptions

The TUG office frequently receives calls asking in what issues of *TUGBOAT* various file descriptions can be found. Here is a list of the pertinent references; information applicable only to TEX78 has been omitted.

- DVI Device Independent file, produced by TEX, read by output device interfaces. *TUGBOAT* 3, no. 2: 14-19.
- GF Generic Font file, produced by METAFONT, read by output device interfaces. *TUGBOAT* 6, no. 1: 8-11.
- PK Packed font raster file, produced by GFtoPK, read by output device interfaces. *TUGBOAT* 6, no. 3: 115-120.
- PXL font raster file, produced by METAFONT78, read by output device interfaces (obsolete, but still widely used), *TUGBOAT* 2, no. 3: 8-12.
- TFM TEX Font Metric file, produced by METAFONT, read by TEX and output device interfaces. *TUGBOAT* 2, no. 1: 12-16.

Much of the information in the TEX Users Group articles has been extracted from WEB source files, so the programs listed below are also useful references.

- DVI TEX82, §583
- GF METAFONT84
- PK PKtoPX
- TFM TFtoPL and PLtoTF