

## Comment on “ $\TeX$ Does Windows”

Jim Fox

The emphasis of the “window paragraphs” article in the March TUGboat seems to be somewhat misdirected. It is not so big a deal that  $\TeX$  can be made to format such paragraphs, indeed the 14th and 15th chapters of the *TeXbook* describe completely and in detail the means both to specify paragraph shape and to split off parts of said paragraphs—thereby rendering these *holed* paragraphs essentially trivial. Nor should the emphasis be placed on the paragraphs themselves (the hole being distracting and making the paragraphs hard to read, especially when the hole produces not only a gap between words but a hyphenated word across the gap as well. The emphasis instead belongs on the  $\TeX$  itself, and the very fact that it can format such idiosyncratic paragraphs with ease. And one should note that the holes can be of arbitrary shape, and there can be several of them in the paragraph.

## Queries

Editor’s note: When answering a query, please send a copy of your answer to the TUGboat editor as well as to the author of the query. Answers will be published in the next issue following their receipt.

The following items, which appear elsewhere in this issue, are in response to, or otherwise relevant to, previous queries.

- First-line special handling (James Alexander, Vol. 7, No. 2, page 110), see page 193.
- Indexing with  $\LaTeX$  (Jim Ludden, Vol. 7, No. 2, page 111), see page 201.
- Setting parallel texts (John Stovall, Vol. 2, No. 2, page 57), see page 190.

### Time Line Macro

This query elicited no response when it was published in  $\TeX$ hax, so I will try the TUGboat audience. In addition to being quite useful for its (admittedly specialized) purpose, it would seem to be a challenging exercise for an expert—something along the lines of some of the esoteric exercises in the *TeXbook* or the tree-making macro of last year’s TUGboat. I offer it as such a challenge.

I would like a macro which makes a “time line”. It would read a file which consists of entries of the form

*(date) (event)*

(presorted if necessary) and produce a vertical line of some preassigned length with tick marks so that the top of the line represents the first date (or #1 in the macro call) and the bottom represents the last date (or #2). Down the line, with vertical spacing mimicking (and that is the key point) time intervals, the dates and events are printed horizontally out to the right. One problem is to do something intelligent when two or more of the dates cluster too closely (e.g. two events on the same date). One can see the general idea, but also many  $\TeX$ nical details. Alternately (perhaps less interestingly), one could write a preprocessor in C or Pascal.

Sometimes the time scale is linear (e.g. for the history of the USA); sometimes a logarithmic scale is appropriate (e.g. cosmological events since the beginning of the universe—as much happened in the first second or so as since—or, compressing in the opposite direction, the chronology of life on earth). Such time lines are a useful semi-pictorial way of presenting chronologies, but are somewhat awkward to create with conventional typesetting. Any takers?

James Alexander  
University of Maryland

### Reply: Printing Out Selected Pages

In TUGboat Vol. 7, No. 3, Helen Horstman asked, "Is there some way by which one can select only a page (or pages) of printout?"

I recently put some new lines, shown below, into MANMAC (the macros of Appendix E that generated Volumes A and E), so that I could put only selected pages into the DVI file. The method should work if you use it at the end of almost any macro file. (Or, if necessary, at the front of a source document.)

The idea is to make T<sub>E</sub>X look for a file called `pages.tex`. If such a file doesn't exist, everything works as before. Otherwise the file should contain a list of page numbers, one per line, in the order they will be generated. After the last page number has been matched, all further pages will be printed. Thus, if you want to print page 123 and all pages from 300 onwards, your file `pages.tex` should say

```
123
300
```

but if you want to print pages 123 and 300 only the, file should say, e.g.,

```
123
300
-9999999999 % impossible number
```

so that the end of file will never occur.

You should rename the `pages.tex` file after you're done with it; otherwise it will continue to affect the output.

The macros cause T<sub>E</sub>X to announce that fact that it's doing something special.

Donald Knuth  
Stanford University

### Using the Windows Environment

We currently run T<sub>E</sub>X on IBM PC/XT and AT's and have recently adopted Microsoft's Windows environment to provide us with a Mac-like interface. At present MicroT<sub>E</sub>X will run without modification under Windows but without pull-down menus and the like. I would be very interested to hear from anyone who either has a .DVI file previewer that will work under Windows or who is interested in developing such a previewer (or any T<sub>E</sub>X product that runs under Windows).

As Windows is about to be upgraded and will form the presentation manager of OS/2 for the new range of IBM Personal System computers, this would seem to be where the future is for those of us who live in the world of IBM compatibility.

Mike Black  
Kingsdown Publishing Ltd.  
London N16 7DP

### Macro for printing out selected pages

```
\let\Shipout=\shipout
\newread\pages \newcount\nextpage \openin\pages=pages
\def\getnextpage{\ifeof\pages\else
  {\endlinechar=-1\read\pages to\next
   \ifx\next\empty % in this case we should have eof now
   \else\global\nextpage=\next\fi}\fi}
\ifeof\pages\else\message{OK, I'll ship only the requested pages!}
\getnextpage\fi
\def\shipout{\ifeof\pages\let\next=\Shipout
  \else\ifnum\pageno=\nextpage\getnextpage\let\next=\Shipout
  \else\let\next=\Tosspage\fi\fi \next}
\newbox\garbage \def\Tosspage{\deadcycles=0\setbox\garbage=}
```