

Point: Lots of windows but little else

Galen Gruman, Assistant Editor

Point, Lemma Systems' mouse-based text editor for IBM PCs, offers windows galore: 20 of them, limited in size only by your system's memory. I've never edited 20 files simultaneously (or even five!)—and I can't imagine ever wanting to. After all, considering the size of a PC screen, how could you really work with all those snippets of text?

However, someone might want to edit pieces from several files, perhaps to combine them. (I still think it would be easier to merge the files and delete what you don't want.) Point will let you do this.

Otherwise, it's a low-end text editor. Commands are invoked through function keys, pull-down menus activated by the mouse, Ctrl-key sequences, and Alt-key sequences. The manual, displays, menus, and prompts are serviceable, but hardly user-friendly. My impression is that the windows got all the programming effort while the user interface was little more than an afterthought.

I found the structure confusing and frustrating. While I like the idea of having several ways available to achieve a function (after all, why limit the software to one preference, especially in a multiuser environment?), I don't like the way Point set it up.

It wasn't so much that methods were hard to figure out; it was more that they were not easy to figure out—and some were downright annoying. For example,

for function-key commands, the Alt and Ctrl keys were interchangeable (Alt-F6 equals Ctrl-F6 equals search backwards), but, for keyboard-key commands, the Ctrl key was inactive. A minor point, perhaps, but it would have been less confusing if the Ctrl key were simply deactivated when used with function keys.

However, I was more bothered that some mouse commands (move and copy) required simultaneous use of the Shift (copy) or Ctrl (move) keys with the left mouse key. Wouldn't it have been better to use the right mouse key or use key-click sequences? As it stands, Point mixes keys from all over: the keyboard, keypad, function keys, and mouse. I never knew where to put my hands (and I am definitely not a touch-typist!).

Point does, however, have some nice features. You can use the thumbing feature to move rapidly within a text file, rather than having to scroll down page by page. (Thumbing is like a rubber band: you use the side of the window as if it were a ruler and click the mouse on a spot proportionately equal to how far you want to go in the file.)

The autoindent feature was nice, as was the insert-ASCII character feature for accessing control characters and the extended IBM character set.

When you save, your file is in ASCII, so it is ready for immediate printing or

transfer by modem. However, Point has no formatting ability for things like bold and underlining, nor does it have a print-from-file feature.

I thought I might achieve bold text and underlines by inserting the backspace character (Ctrl-H) and either repeating the character (for bold) or striking the underscore (for underlines). Unfortunately, Ctrl-H is not an insertable ASCII character in Point.

I also tried inserting only a carriage return (Ctrl-M)—rather than the traditional return and linefeed (Ctrl-M plus Ctrl-J)—at the end of the line so the printer head would return to the beginning of the line but not move down to the next line. Foiled again! Point automatically attaches a linefeed when you insert a return, even in the insert-ASCII mode.

Looking at the Point manual, I realized their underlines were simply rows of hyphens on the line below whatever was "underlined." I still wondered how they got the bold copy in their manual. (You can't insert the escape character either, so I doubt they were able to toggle their printer's bold through an escape-key sequence.) "Maybe they used a text formatter!" I thought. It turned out that they did.

In any event, the \$120 Point offers nothing you can't get elsewhere. While it's not bad, it isn't good either. There are better editor programs out there.

Reader Service Number 22