Using the text editor Joe.
JOE COOL

ANDREW HALLIWELL

Whenever the subject of text editors comes up in a group of Unix or Linux users, an almost religious debate will commonly arise, as people gravitate to one of two factions within the room. These two factions are the advocates of VI (and it's clones) and Emacs. The arguments on the side of VI will point out that VI is a standard editor used on every Unix system which is incredibly powerful, with syntax highlighting and the ability to very productive. The argument on the side of Emacs will say that it's easier to use and has a method of configuration and extension so powerful it can be used to create programs within Emacs (performed using the Emacs LISP interpreter). In fact, Emacs could be almost considered an operating system in its own right.

There are drawbacks to both of these editors however. VI has an extremely steep learning curve. Someone who just wants to enter an editor, type what they need and correct spelling without having to fiddle with different modes is out of luck with a moded editor. Emacs can do almost anything, including run mail and newsreaders – but at a cost. The cost in Emacs' case is size. While VIM (one of the more popular VI clones that stands for VI iMproved) only takes approximately 700k. Emacs takes up over 22MB of disk space. There are many other editors available however, and one of the most flexible alternatives to the two previously mentioned is Joe's own editor, commonly known simply as Joe.

Joe is a modeless editor with multiple personalities. Built into Joe are five different personalities. See Table 1. All of these personalities exist within the same executable that is less than 200K in size, and are activated by symbolic links to the main executable Joe.

Getting started

Joe comes with simple built in help. This is activated by pressing ^K H and occupies the top area of the editor screen. Help comes in six sections, which are accessed by using ESC. and ESC, to move forward or back through the screens. The different Help screens explain the following functions: Basic editing commands such as cursor movement, block highlighting and manipulation, load, save, insertion of text files, search and replace and spell checking and Window manipulation. The editor screen can be split into multiple windows to allow the viewing/editing of more than one file, or part of a file, at once. Miscellaneous commands such as scrolling, macros and book marking are also covered. Programming commands such as commands to parse errors, indent and search by code-block separator or tag file. Advanced Search and Replace Help clarifies regular Joe expressions. The final Help window is the e-mail address to which bug reports should be sent.

Basic editing commands

Joe uses three methods of command access. Ctrl, Ctrl+K and ESC. Basic text navigation can be performed using the cursor keys, Page up, Page down, Home and End. These basic navigation commands are duplicated with Control key combinations so that they will work on non-PC keyboards or terminals. With these commands, it's quite simple to use this editor for basic text manipulation. You also want to be able to load, save and insert text files into the body of the document. As well as inserting text, it is also desirable to be able to delete it. Deleting a larger block of text is covered in block manipulation next, but deleting words, lines and parts of lines is covered here. On a PC keyboard, Backspace and Delete will delete the character before the cursor and on the cursor respectively. But again, Delete is replicated as a Control key. When you have inserted into (or deleted from) an existing paragraph, the command Ctrl+K+J will reformat the text.

Joe in the standard personality

Occasionally, you may want to delete, copy or move a large block of text. For example, when posting a reply to a newsgroup or e-mail, to delete large blocks of text which are not relevant to your reply. Joe can do this quite simply by using its block manipulation commands. If no block is selected with the indent/outdent commands, the paragraph the cursor is currently on will be marked as a block. A simple example of the filter command can be seen by making a block of the above key definitions and using the shell command wc. This replaces the block with the text below.
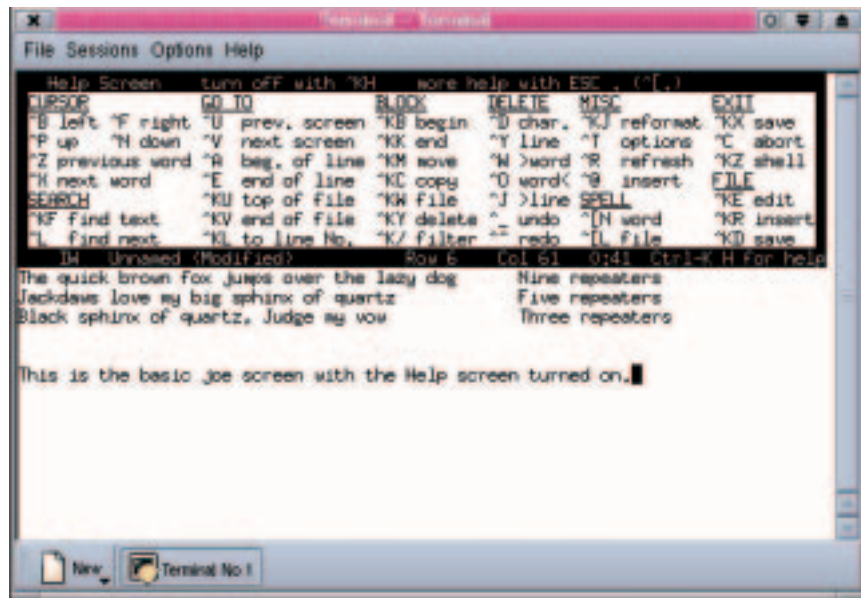
        10   74   418

These figures describe 10 lines, 74 words and 418 characters. This is by far the most powerful command in Joe, as it taps into the full power of Unix. With it, you can create shell or Perl scripts to do complex things and use those as filters for Joe.

Everyone makes mistakes once in a while, be it deleting a block of text by accident, or replacing all occurrences of a word with something nonsensical. Joe is quite capable in this area as it has multi-level undo and redo commands. Search and replace functionality is one of the most useful features in a text editor. In Joe, this is also powerful and is activated with the following commands:

    Ctrl+K+F   find text
    Ctrl+L   find next occurrence
    \^   match the beginning of a line
    \$ match the end of a line
    \<   match the beginning of a word
    \>   Match the end of a word

Spell checking

No editor in this day and age would be up to much if it had no spell checking capabilities. There are two spell check commands in Joe. Check a single word (if you are unsure how one is spelt) and check the entire document. Joe calls on *ispell* to perform all its spell-checking functions. This means that *ispell* can be replaced by other spell checkers as it suits the user, as long as a symbolic link is used to make the new spell checker assume the name *ispell*.

Joe configuration options

Pressing Ctrl+T brings up a horizontal menu at the bottom of the screen. This can be navigated by using the left and right cursor keys and activated by pressing Enter. The menu can be dismissed by pressing Esc.

In Rectangle mode, instead of selecting a continuous block of text, a rectangular area of text is selected. This is useful for editing tabulated data. When a block is moved or deleted in this mode (in conjunction with overtype mode) the area is replaced by spaces rather than deleting the text to maintain table integrity.

In insert mode, the block is removed and text to the right of the block falls back to fill the hole. In Anchor mode, the Ctrl+K+B marks the start of the block, and the end of the block follows the cursor.

This covers most of the editor's simpler commands. But there are many more that are beyond the scope of this article. These include commands that deal with code editing, editing multiple files, macros, shell commands and more.   ■

Table 1

jpico: Mimics the pico editor usually supplied as part of the PINE package. It is however much more powerful than pico, as it still holds all the features of Joe.
jstar: This imitates the old WordStar editor that was widely used in the DOS era.
jmacs: This copies Emacs. It doesn't support ELISP or any of Emacs' more unusual properties, but can be used comfortably by Emacs users when Emacs isn't available.
rjoe: This is restricted Joe. It can be used in environments where you wish to limit what the users of your system can do. rjoe can only edit the filename(s) supplied in the command line, which means in a menu-based shell, it can be used to prevent the user from editing configuration files in their home directory, but still be able to reply to e-mail/news.
Joe: This is the personality that is most widely used and the one that will be covered in more depth here.