



ISPF Edit and UNIX/Linux vi - a comparison

With z/OS UNIX Systems Services embedded in the z/OS operating system, and Linux on z support for Linux operating systems running native under z System LPARs, z/VM, or KVM, it seems likely that z/OS users will also need to be UNIX/Linux users. This is the first in a series of planned articles to address the similarities and differences in each operating system's user interface. We begin with a comparison of the primary file editors.

Watch this [QuickRef Academy](#) video to learn more

NOTES:

1. The quote (“) characters used in this text indicate the command strings and are not a part of the command. An apostrophe (') in the command string is actually part of the command.
2. The command “vi” is most likely invoking the vim (vi Improved) editor which is a more user-friendly editor that uses the arrow keys for cursor placement/scrolling, as well as Page Up/ Page Down keys for scrolling.

Profiles

ISPF Edit Profiles are stored in the ISPPROF data set and there are multiple profiles. Each ISPF Edit Profile is associated with a specific group of data sets by the data set file type. For example: the JCL profile would apply to all files in data sets whose names end with “.JCL”.

The vi editor has a single profile named “.exrc” which is stored in the user's home directory.

Both profiles have a setting which effects the line numbers displayed in the edit session. In ISPF, it's the NUM option (NUM ON, NUM OFF) and in vi it's the “number” option (set number, set nonumber). To achieve similar displays, set the profiles for ISPF using **NUM OFF**, and vi .exrc using **set number**.

ISPF automatically follows indentation in the text files you edit. To have vi work the same use **set autoindent** in the vi .exrc profile.

Both editors have a profile option for parenthesis matching. In ISPF use the command **HI PAREN** to enable and disable this feature. In vi, the options are **set showmatch** and **set noshowmatch**. In both cases the option is off by default.

ISPF and vi have two command modes. ISPF uses “**line commands**”, issued by overtyping the line numbers to the left of the text, and “**primary commands**”, issued on the Command ==> line input field. The vi editor has “**browse mode**” commands which cause the editor to take actions based on the keys pressed on the keyboard, and “**ex mode**” commands which are issued on the editor's command line.



ISPF Line and vi Browse Commands

ISPF has seven basic line commands for editing. They are: “l” – insert, “d” – delete, “r” – repeat, “c” – copy, “m” – move, “a” – after, and “b” – before.

In ISPF the letter “l” inserts a line below the line where it is entered and places the editor in input mode. Input mode is terminated by pressing ENTER without entering data on the input line.

In vi typing “o” in browse mode inserts a line below the line where the cursor is and places the editor in insert mode. Insert mode is terminated when the ESC key is pressed.

The ISPF line command “d” deletes a line, and “d3” deletes 3 consecutive lines. In vi, “dd” deletes a line, and “3dd” deletes 3 consecutive lines.

In ISPF, “r” repeats a line. Using “rr” to mark both the beginning and end of a block of text repeats the entire block of text. You may also use the “c” command to copy, the “m” command to move and “a” for after, or “b” for before, to copy and move lines in the file. For vi, the “yy” command copies lines and “dd” cuts lines. The placement for lines that are copied or cut is determined by “p” (put after), and “P” (put before). The vi equivalent to the ISPF “r” (repeat) command is “yy”, then “p”, all on the same line.

ISPF and vi command lines

The ISPF command line is indicated by “**Command ==>**” and may appear at the top or bottom of the display. The vi command line is at the bottom of the edit display. Cursor placement in ISPF is done via the HOME key (top), F24 (top or bottom command line), or moving the cursor with the mouse or arrow keys. The **vi command line is accessed using ESC + “:”** in most cases, and browse mode is resumed when the command completes.

In ISPF you can issue **TSO and ISPF commands** from the command line and return to the edit session using ENTER or F3. In vi pressing **ESC+“:”** and entering “**sh**” invokes the current shell where you can issue UNIX/Linux commands. Entering “exit” in the shell returns to the edit session.

ISPF can copy additional text from another data set using the command “**copy ‘dataset(member)’**”. In vi use **ESC+“:”** followed by “**r /filepath/filename**” to accomplish the same thing.

The ISPF CHANGE command format is “**c old-text new-text**” while vi uses “**%s/old-text/new-text/**”. To change all occurrences in an ISPF edit session use “**c old new ALL**”. In vi it would be “**%s/old/new/g**” where “%” is all lines and “g” is all occurrences on each line.

continued next page

The FIND command for ISPF uses “**f text**” for a forward search and “**f text prev**” for a backward search. The **F5 key repeats** the search in the current direction. In vi the “/” initiates a forward search and “?” initiates a backward search. In either case the command “n” repeats the search in the current direction. The vi search commands are actually a function of browse mode. Typing any character other than “/”, “?”, “n”, or “N” will take the browse mode action assigned to that key. For example, typing “o” while using a search in vi will insert a new line below the last line located by the search and place you in insert mode.

Both ISPF 7.1 for z/OS 2.1 and vi support **regular expression strings** used in find and change/substitution actions. ISPF also supports “picture strings” which are similar to regular expressions. This more advanced form is too complex to address in this article, but may be addressed in a future post.

In ISPF you may use the “**save**” command at any time to save changes and continue editing, the “**create**” and “**replace**” commands to save under another name or dataset, and **F3 (END or EXIT)** to save and exit. In vi the command “**w**” saves without exiting, “**w /filepath/filename**” allows saving under a different name, and “**wq**” saves and exits. The vi “**q!**” command is equivalent to ISPF’s “**cancel**” command. Remember that vi requires you to press ESC+“:” to access the command line. You also press ESC to return to browse mode.