Server Operating Systems

Lecture 9
Text Editors

Text Editors

On KDE desktop, text editors are on:

Start Menu > Utilities > Editor

There are many GUI text editors available for Linux.

- Kate
- gEdit
- Kwrite

There are more sophisticated ones that can be used for coding:

- Bluefish
- Quanta plus
- Screem

These offer extra features such as syntax highlighting.

vi Editor

visual Editor

ASCII text editor, no formatting capabilities

Useful when working remotely on unix workstations, called the number one tool for the network administrator.

It may be the only text editor available on the system.

Does not require mouse or arrow keys to position cursor!

Available on all UNIX systems

Also used for remote login or telnet

First Exercise

Type: **`vi firstfile`**

To insert some text, type: **`i (for insert)`**

Type three lines of text

Press the esc key to stop inserting.

Use the arrow keys to go to the end of the first line.

Does the backspace key work?

Press the **`i`** key

Does the backspace key work now?

Press the esc key to stop inserting.

Type: **`:wq`**   to save the file and exit.

The vi Modes

Command Mode

– Mode entered when vi starts up.

– For moving the cursor around on the screen and performing advanced editing.

– Only way to access the other vi modes.

– The 'Esc' key will always get you back to Command Mode from any other mode.
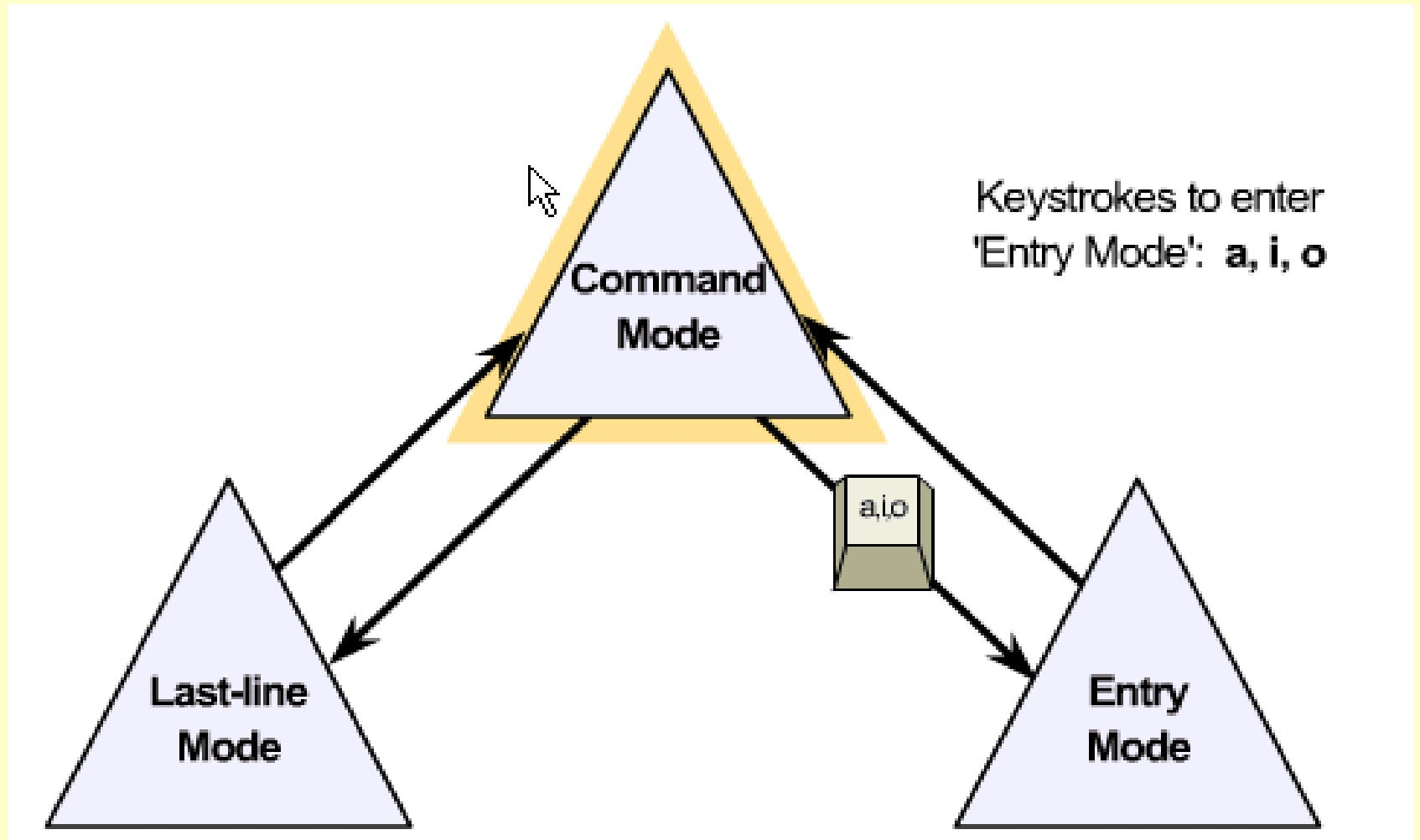
The vi Modes

Entry Mode

– Allows the modification of text into the loaded file.

– Three ways to get into Entry Mode from Command Mode:
  1) Type '**i**' to *insert* text at the cursor.
  2) Type '**a**' to *append* text after the cursor's position.
  3) Type '**o**' to *open* a new line between the cursor's line and the line below it.

– Hit 'Esc' to get back to Command Mode.
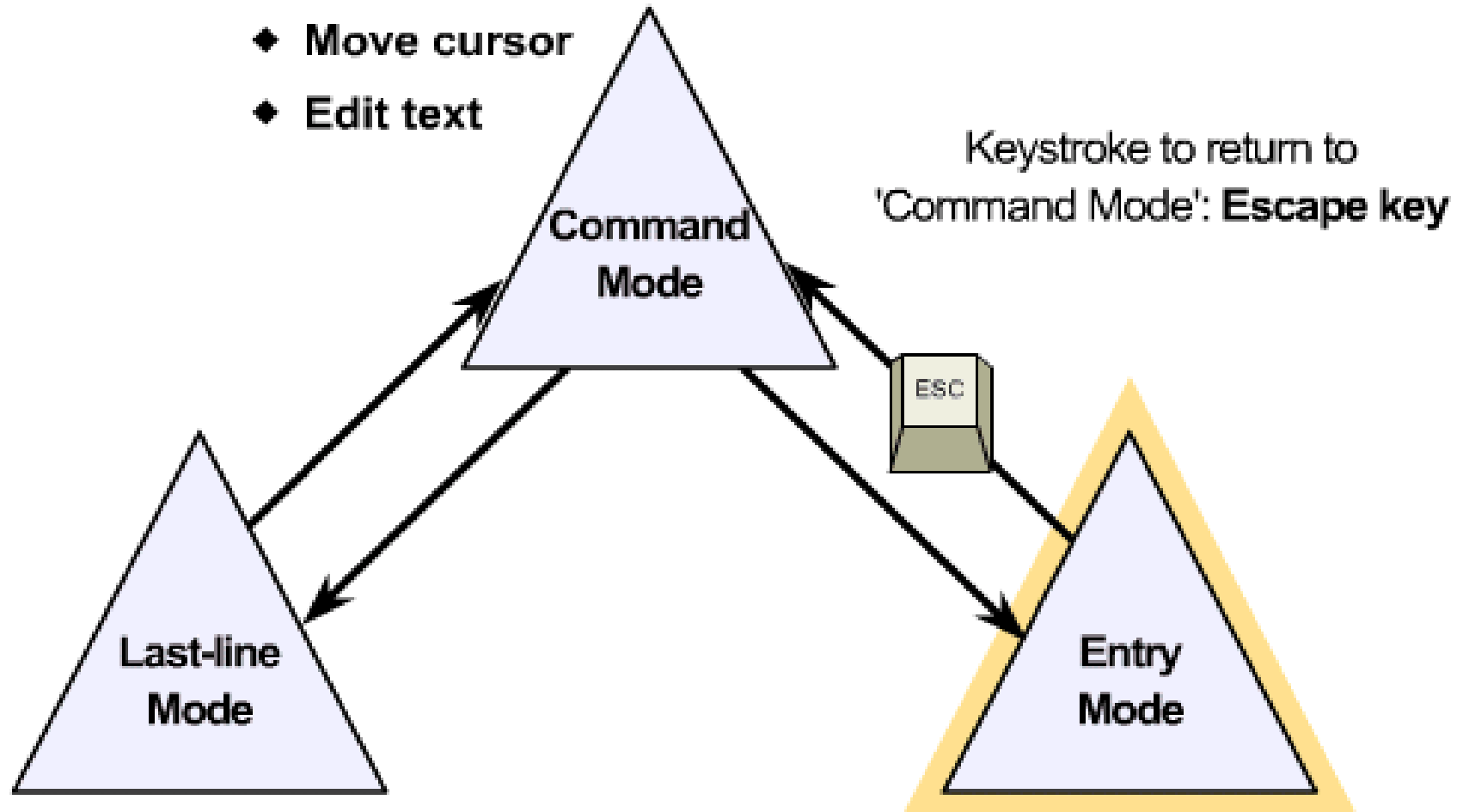
The vi Modes

Last Line Mode

– Allows saving the edited file to disk, exiting vi, setting vi preferences, and searching for character strings.

– Usually entered by typing ':'
  • The cursor is put on the last line of the screen where the commands can be entered.

– Hit 'Esc' to get back to Command Mode.

Moving Between vi Modes

Command Mode

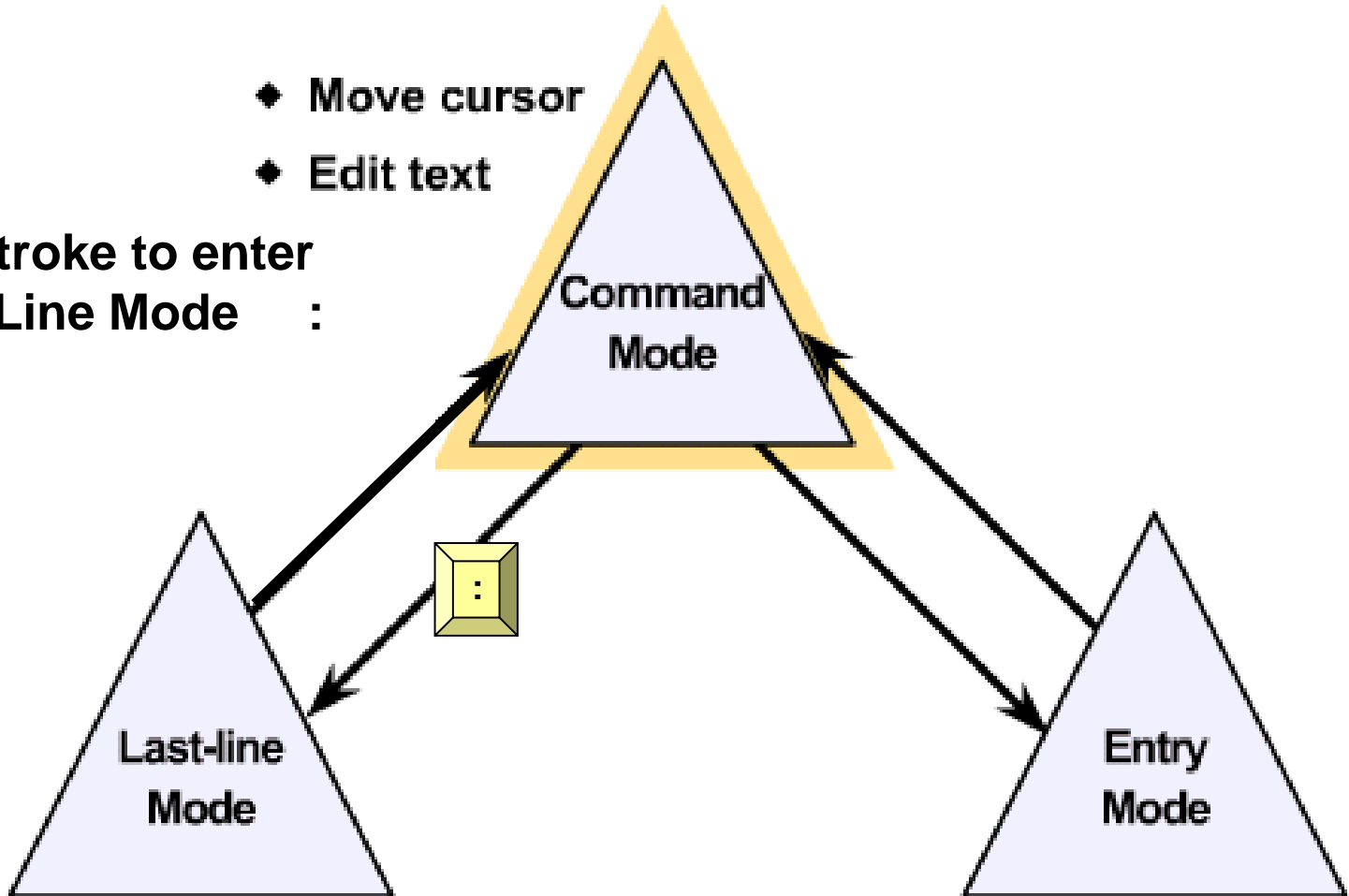Last-line Mode

Entry Mode

Keystrokes to enter 'Entry Mode': **a, i, o**

a,i,o

Moving Between vi Modes

- Move cursor
- Edit text

Command Mode

Keystroke to return to 'Command Mode': **Escape key**

ESC

Last-line Mode

Entry Mode

Moving Between vi Modes

Move cursor

Edit text

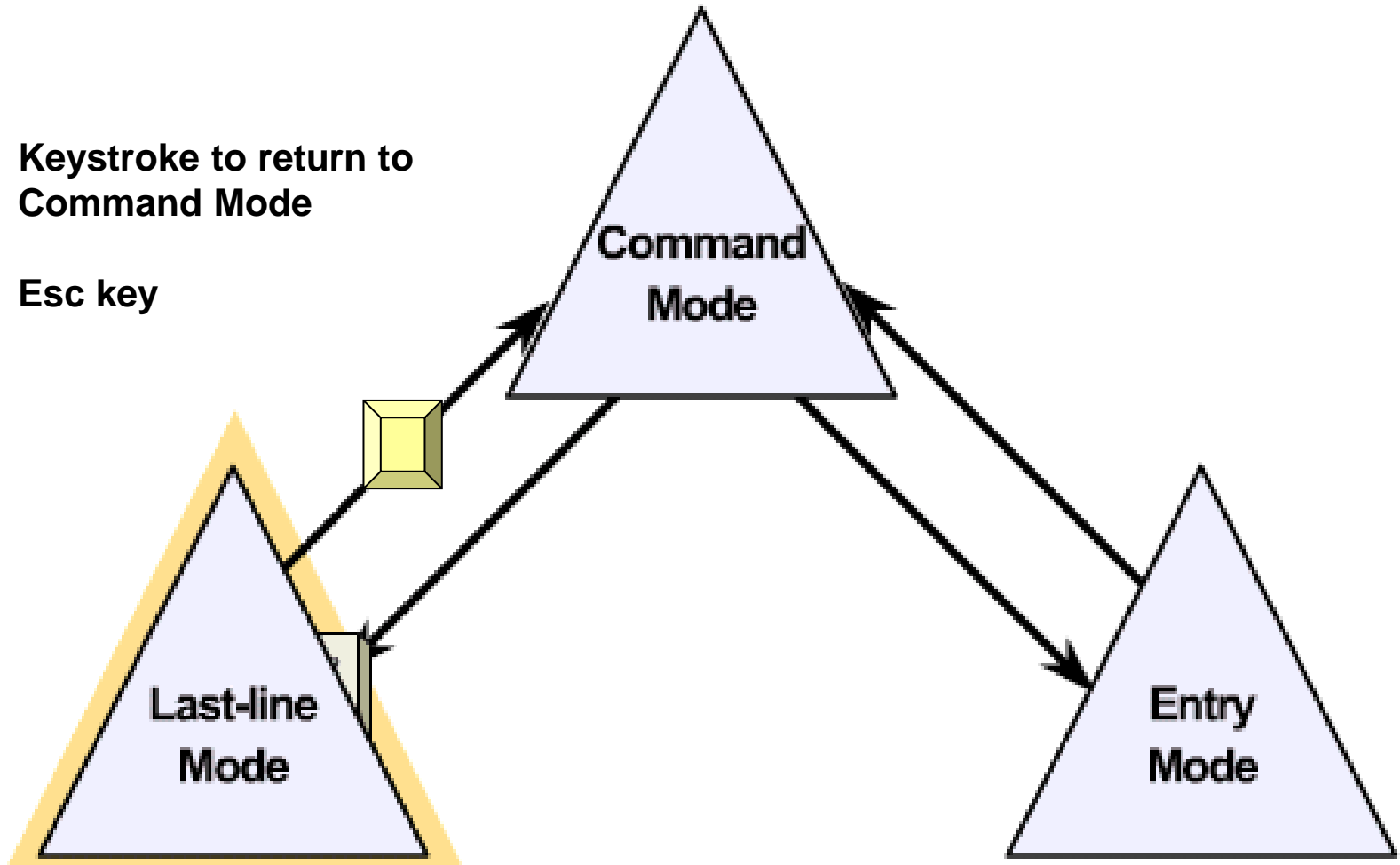**Keystroke to enter
Last Line Mode   :**

Command
Mode

Last-line
Mode

Entry
Mode

:

Moving Between vi Modes

Keystroke to return to Command Mode

Esc key

vi Command Syntax

$ vi [*option(s)*] [*filename*]

vi – by itself, starts the vi editor and opens a new file (not yet named)

vi filename – if file exists it is opened for editing - if not it's created

vi -r filename - recover a crashed file

vi Editing Commands

In Command Mode:

delete

Command	Meaning
x (lowercase)	Delete character at the cursor
dw	Delete word (or part of word to right of cursor)
3dw	Delete three words
dd	Delete line containing the cursor
3dd	Delete three lines

undo, change

Command	Meaning
cw	Change word (or part of word) at the cursor location to the end of the word. This command is a multi step operation: 1. position the cursor at the beginning of the word 2. press cw 3. type new word 4. press ESC
3cw	Change three words
r	Replace character at cursor with one other character
u	Undo previous command

copy, paste

Command	Meaning
yy	Yank a copy of line to the clipboard
p	Put yanked or deleted line below current line
P (upper case)	Put yanked or deleted line above current line

Positioning Commands

In Command Mode:

> Arrow keys can usually be used, but may not always work. The navigation commands will always work.

Command	Meaning
j (or down arrow)	Move down one line
k (or up arrow)	Move up one line
Space bar	Move right (forward) one character
w	Move forward one word (including punctuation)
b	Move back one word (including punctuation)
$	Move to end of line
0 (zero) or^	Move to beginning of line
Return	Move down to beginning of next line
h (or left arrow)	Moves back one character
l (or right arrow)	Move forward one character

vi Entry Mode

Get your cursor to the right place, then go into Entry mode, so that you can enter text

Command	Meaning
a	Append text after the cursor
A (Upper case)	Append text at the end of the line
i	Insert text before the cursor
o	Open a new line below the cursor

Saving Files and Quitting vi

Get into Last Line Mode (:) and then use w or q

Command	Meaning
:w	Write buffer (Save changes and continue working in vi)
:w new_filename	Write buffer to new_filename
:wq	Write buffer (save changes) and quit vi
ZZ (Upper case)	Save changes and quit vi. Alternative to :wq
:q!	Quit without saving changes
:wq!	Write buffer (save changes) and quit vi (The ! will override read only permissions if you are the owner of the file.)

The only vi commands I remember

Normally you can get by with just the following commands:

i   to get into 'enter' mode
esc   to get back to command mode
:   go to last line mode

arrow keys (in command and enter mode)
  or
j (down), k (up), h (left), l (right) (only in command mode)

x   delete character (command mode)
backspace delete character (enter mode)

:wq   save file and exit

Searching and Customisation

In Last Line Mode:

search

Command	Meaning
G (upper case)	Go to last line of file
:21	Go to line 21
/string	Search forward for string
?string	Search backward for string
n	Find next occurrence of string

session

customisation

Command	Meaning
:set nu	Show line numbers
:set nonu	Hide line numbers
:set showmode	Display current mode of operation (turned on automatically if the vedit command is used)
:set noshowmode	Turn off mode display
:set	Display all vi variables set
:set all	Display all possible vi variables and their current settings

Alternatives to vi

Non-GUI alternatives to vi are:

- Emacs,
- Pico, and others
- cat
  - The cat command can *only* be used to create text files.

Advantages of Emacs

There are many advantages to using Emacs. Some of the more important ones are listed below:

- Free and open source.

- Customizable and programmable.

- Modes (not like vi modes).

- Multiple files and instances simultaneously.

- Integrated email client, news reader, and web browser.

- Some have called it a text editor with a built-in operating system!

Perceived Disadvantages of Emacs

"Hard to use at first."

"Far too many control characters."

"Makes learning vi look easy."

Emacs is much larger than vi.

Emacs is not as available as vi.

vi vs Emacs was (still is) one the great flamewars of computing.