Vim

Omaha Linux Users Group
Vim / Gvim Presentation
November 1, 2005

By: Jim Franco
Email: stream@cox.net

Vim Sky Image Credit - http://www.derwok.de/2d-3d/3d/vimlogo/

Window Commands

- **:split <filename>** - open file in window
- **:vsplit <filename>** - open in vertical
- **<Ctrl-w><Ctrl-w>** - switch windows
- **:wq** – write and quit a window
- **<Ctrl-w>j   <Ctrl-w>k** – move to upper/lower
- **<Ctrl-w>h <Ctrl-w>l** – move to left/right
- **<Ctrl-w>_   <Ctrl-w>=** - window resizing
- **<Ctrl-w>L   <Ctrl-w>J** – change orientation

Buffer Commands

- **:buffers** – list all of the current buffers
- **:buffer** <buffer_number> - switch to a buffer
- **:badd <filename>** - add a buffer to the list
- **:bdelete <buffer_num.>** - delete a buffer
- **:sbuffer <buffer_num.>** - open in a window
- **:hide** – hide a buffer from a window
- **:unhide** – open hidden buffers

Marks

- Up to 26 marks can be placed.
- Represented by letters a-z
- **ma** – place a mark and name it the letter “a”
- **`a** – to go to exact row and column of the mark
- **‘a** – to go to column one in ther row of the mark
- **:marks** – shows all marks set

Registers

- 26 user definable registers
- Referenced as letters a-z
- **“ay** – yank a selection and place it into register a (will replace existing contents)
- **“Ay** – yank a selection and append it to register a
- **“ap** – put from register a
- **:registers** – view all registers

Recording

- **qc** – begin recording in register c
- **q** – stop recording
- **@c** – use recording in register c
- **@@** - repeat last used recording

Regular Expression Ordinary Atoms

- **\<** - attach as beginning of a word
- **\>** - attach as the end of a word
- **^** - beginning of the line
- **$** - end of the line
- **.** - any single character
- **x** – any literal character x

Regular Expression Branching Atoms

- **\|** - match previous or next pattern (or)
- **\&** - match next pattern only if previous pattern matched (and)

Regular Expression Multi Atoms

- * - 0 or more times
- **\+** - 1 or more times
- **\{n,m}** – n to m times (as many as possible)
- **\{n}** – exactly n times
- **\{,m}** – 0 to m times (as many as possible)
- **\{}** - 0 or more times (as many as possible)

Regular Expression Grouping Symbols

- **\(\)** - group patterns for later use
- **\1** – reference group 1
- up to 9 groupings per expression

Regular Expression Collections

- **[]** - use brackets to specify
- **[a-z]** – hyphenate to specify a range
- **[^aeiou]** – circumflex specifies not

Regular Expression Character Classes

- **\s** – space and tab characters (\S - not)
- **\d** – decimal digits (\D - not)
- **\a** – an alphabetic character (\A - not)
- **\l** – a lowercase letter (\L – not)
- **\u** – an uppercase letter (\U – not)
- **\t** – a tab

Regular Expression Character Classes

- **[:blank:]** - space and tab characters
- **[:alpha:]** - an alphabetic character
- **[:lower:]** - a lowercase letter
- **[:upper:]** - an uppercase letter
- **[:tab:]** - a tab
- **[[:alpha:]]** - group with additional brackets
- **[^[:digit]]** – circumflex means not

Substitution

```
:[range]substitute/from/to/[flags]
```

Abbreviations

- **:abbreviate (left side) (right side)**
- **:abbreviate** – list abbreviations
- **:unabbreviate (left side)** – remove abbreviation
- **:abclear** – remove all abbreviations
- **:iab, :cab, :iuna, :cuna, :iabc, :cabc** (insert and command mode forms)

Mapping

- **:map (left side) (right side)** – any mode
- **:unmap (left side)** – remove a mapping
- **:map** – list mappings
- **:imap, :vmap** – insert or visual mode only
- **:noremap, :inoremap, :vnoremap** – use the default actions of any characters within the mapping.

Folding

- **zf** – create a fold
- **zc** – close a fold
- **zo** – open a fold
- **zd** – delete a fold