

# System Software: Vim

Param Singh

National Institute of Technology, Hamirpur

*14MI517*

February 22, 2016

# Overview

- 1 What is Vim
- 2 Why Vim?
- 3 Installing Vim
- 4 Vim Philosophy
- 5 Using Vim
- 6 How to get good at Vim
- 7 Queries

# What is Vim?

- A Programmer's text editor
- Designed expressly for efficient text editing
- IDE-like features not built-in, but 1000's of plugins
- Highly customizable and extensible
- Incredibly powerful - will double or triple your productivity
- Kinda steep learning curve

# Why Vim?

Learning Curve? Wait, I never missed anything using ...

- Gedit
- Nano
- Sublime Text
- Atom
- Notepad
- that editor whose name we don't mention
- ...whatever

# Reasons to use Vim

- Vim is part of the standard.
- Meaning you can find it everywhere including BSD, Mac OS X, Solaris, your refridgerator (if it's POSIX compliant), etc.
- Really powerful (much more than other editors like gedit, nano etc)
- Makes you as fast as the Flash.
- For all of us lazy people, never have to raise hands from the keyboard.
- Increases productivity.

# Installing Vim

- Linux / Unix: Present in the repositories of practically all distributions
- OSX: Ships with vim preinstalled, but use MacVim for a GUI
- Windows: Use gVim (or install Linux!)

# Vim Philosophy

Vim philosophy can be easily described by the following points:

- Modal Editing: Text editing is a complex task and can be divided into modes.
- Operators and Motions: Commands that can be run directly from the keyboard.

# The Modal Editor

- Observation: We spend more time reading and editing than typing!
- Vim helps by providing separate modes for reading and editing.
- Normal mode: All keystrokes are interpreted as commands.
- Insert mode: Regular typing as you know it.
- Visual mode: Highlighting
- ... and more!



# Surviving in Vim

- `:wq` - Saves and quits vim (First thing you should know!)
- `i` - Gets you into Insert Mode, although you don't want to stay there for long.
- `ESC` - Gets you back to normal mode, where you want to spend most of your time.



# Advanced Movement

- Vim is all about saving keystrokes, so there are many many motion commands.
- `^` - beginning of line
- `b` - beginning of word
- `e` - end of word
- `w` - next word
- `$` - end of line

# Operator-Motion Model

- Use the operator motion model to concatenate commands
- d for deletion (operator)
- 2 for two (count)
- w for word (motion)
- d2w - deletes the next two words
- by providing an operator, a count and a motion, we create a new command.
- similarly, d15w will delete the next 15 words.

# Operators

Numerous operators!

- d - delete
- c - Change (delete and go into insertion mode)
- y - Yank! (copy)
- v - Visual / Highlight

Combine with great effect! Typing an operator twice makes it act upon the entire line.

# Other Common Commands

- v - visual line highlighting
- o - create new line below and go into insertion mode
- x - Delete character under cursor
- R - Go into replacement mode (Really awesome!)
- u - Undo (helpful :))
- ctrl + r - Redo
- A - Append to end of line (goes into insertion mode)

# Advanced features

Many more features I can't go into for lack of time.

- Regular expression find and replace - `:s/exp/replace`
- Scripting: Vim script - This is really awesome, an entire complete programming language
- Mapping directive line commands to certain keys (`:map key cmd`)
- Configuration in `.vimrc` for Unix users
- Syntax highlight (syntax on)
- show cursor position (set ruler)

# How to get good at Vim

- Keep using it.
- Don't give up.
- Keep using it.

# Remember...

- Being good at modal editing means being able to think up complex commands fast. Learn with practice!
- Use vintutor!
- Don't be surprised when you find yourself reaching for the ESC button even when you're not using vim.
- Use vim when web browsing (vimium / pentadactyl).
- Vim is love, vim is life.
- Emacs was sent by the devil to tempt you from the path of the righteous.



# Queries

Questions? `/**` and maybe answers! `*/`