How to be a Vim Ninja

Robert Bittle
robert.bittle@dominionenterprises.com
github.com/guywithnose

December 12, 2014

Slides and demos at:
https://github.com/guywithnose/de-devcon-vim
http://bit.ly/11po3hk

Why is Vim hard to learn?

- Vim is extremely complex
- Vim is full of features that are not imediately discoverable
- Vim is designed for experienced developers not beginners.

Why is Vim hard to learn?

- Vim is extremely complex
- Vim is full of features that are not imediately discoverable
- Vim is designed for experienced developers not beginners.

Why is Vim hard to learn?

- Vim is extremely complex
- Vim is full of features that are not imediately discoverable
- Vim is designed for experienced developers not beginners.

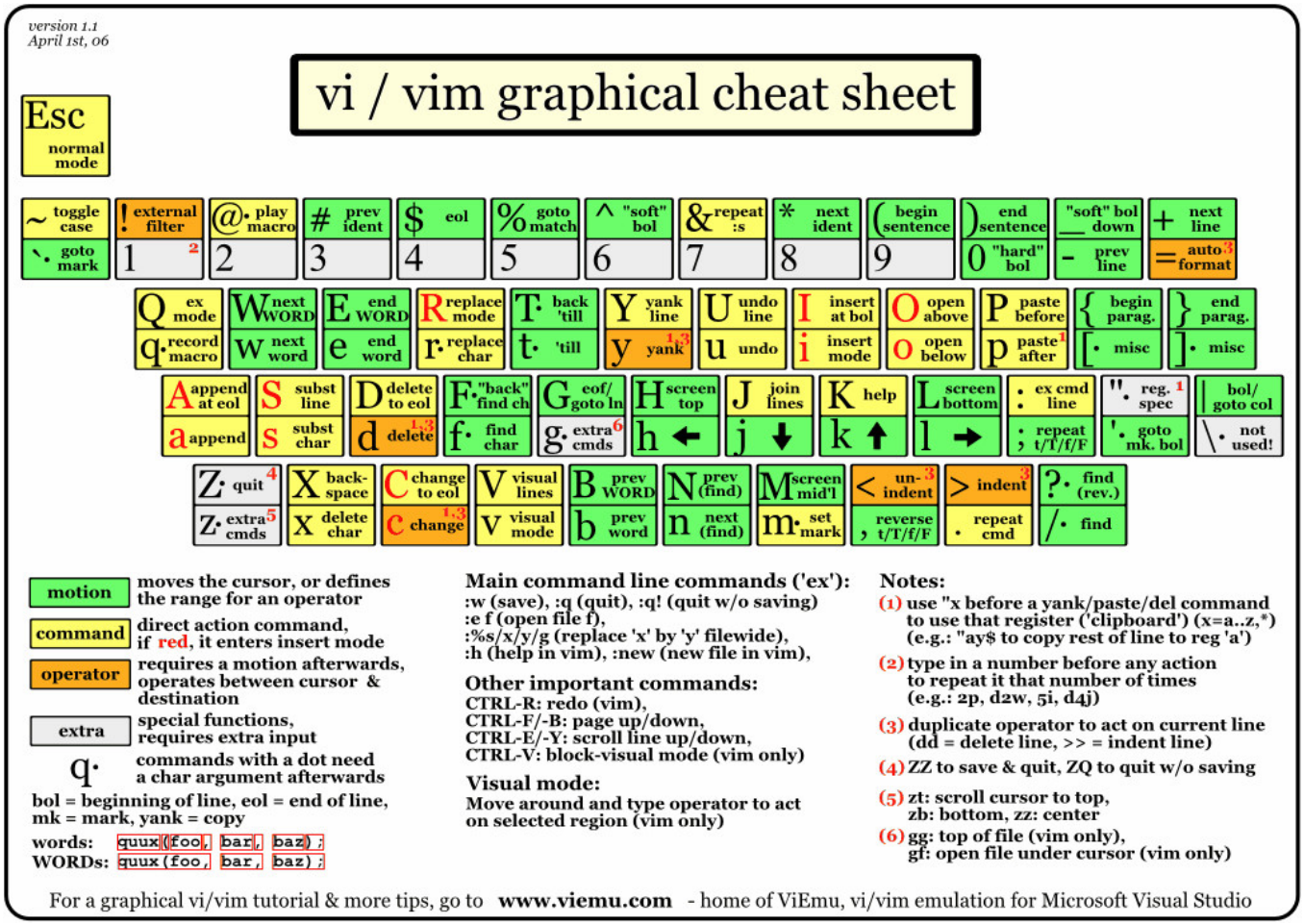
Why is Vim worth the steep learning curve?

- Vim is designed to maximize your productivity potential.
- Your vim skills can grow slowly but consistently over time

Why is Vim worth the steep learning curve?

- Vim is designed to maximize your productivity potential.
- Your vim skills can grow slowly but consistently over time

The power of modes



vimtutor

- vimtutor is a primer for the base functionality, but it is only enough to get you started.
- The most important thing you learn in vimtutor is where to find the manual.

vimtutor

- vimtutor is a primer for the base functionality, but it is only enough to get you started.
- The most important thing you learn in vimtutor is where to find the manual.

Vim documentation

- Vim has very exhaustive documentation.
- It is not intended to be read completely.
- Learn to use the search functionality (:helpgrep).

Vim documentation

- Vim has very exhaustive documentation.
- It is not intended to be read completely.
- Learn to use the search functionality (:helpgrep).

Vim documentation

- Vim has very exhaustive documentation.
- It is not intended to be read completely.
- Learn to use the search functionality (:helpgrep).

Maintain a cheatsheet

- Decide on a few (no more than 10) things you want to learn.
- Write them down where you can see them while you are working in vim.
- Force yourself to use them in real world scenarios.

Maintain a cheatsheet

- Decide on a few (no more than 10) things you want to learn.
- Write them down where you can see them while you are working in vim.
- Force yourself to use them in real world scenarios.

Maintain a cheatsheet

- Decide on a few (no more than 10) things you want to learn.
- Write them down where you can see them while you are working in vim.
- Force yourself to use them in real world scenarios.

Recognize mechanical operations

- How often do I do this?
- How long did it take me?
- Can it be scripted?

Basic commands

- q[0-9a-z] - start recording
- Do something that is repeatable
- q - stop recording
- @[0-9a-z]
- :help complex-repeat

Tips

- Try to start the macro with an operation that will fail after the last iteration.
- If the macro is applied linewise to a block of consecutive lines use visual block mode.
- Try to avoid absolute motions.
- Use search motions, word motions, mark motions, etc.

What is a vimrc file?

- Your vimrc file is where you store your custom vim configuration.
- Vim loads this information every time that it starts up.
- As you become a more mature vim user, your vimrc will grow to include all the special sauce that makes vim work the way you like it.

Mappings

- Mappings are an easy way to bind macros to a key or combination of keys.
- Over time as you find ways to automate common operations with macros you can save them as mappings in you vimrc file.

Motions and Actions (Vim grammar)

Actions

- Delete - d
- Change - c
- Yank (Copy) - y

Prepositions

- in - i
- around - a

Motions

- Word motions – w, W, b, B, e, ge, E, gE
- Search motions – t, T, f, F, /
- Text object motions – p, s, },],), >, t, ', "

Dot Command

- Think of this like an automatic mini macro.
- Combine simple actions using the change(c) action.

My Favorite Plugins

- Editing tasks
  - Surround
  - Ultisnips
- Folding (Usually language specific)
- Interface
  - fugitive
  - CtrlP

How to install plugins

Pathogen

- Add pathogen.vim to ~/.vim/autoload
- Add this line to your .vimrc

  execute pathogen#infect()
- Plugins get loaded from ~/.vim/bundle

How to install plugins

Git

Once you have Pathogen installed

- Copy plugins to your bundle directory.
- If the plugin is in git you can check it out to a directory in bundle.
- Take advantage of git submodules.

Directory structure

- plugin
- ftplugin
- autoload
- doc