

Hackathon Spring 2018



DevOps: Coding Infrastructure

Chad Sutton
casutton@noctrl.edu





About Chad



- 15 Years as an IT Engineer at JPMorgan Chase
 - Desktop, file and print services Engineer for 169,000 desktops
 - Started first internal department web server in 1998
 - Lead architect and eventually web team manager for the IT website.
- Started at North Central College in 2009 managing desktops, servers, and advising on Drupal.
- Using Agile/Continuous Delivery philosophy, techniques, and process for almost 20 years.

What the heck is DevOps?



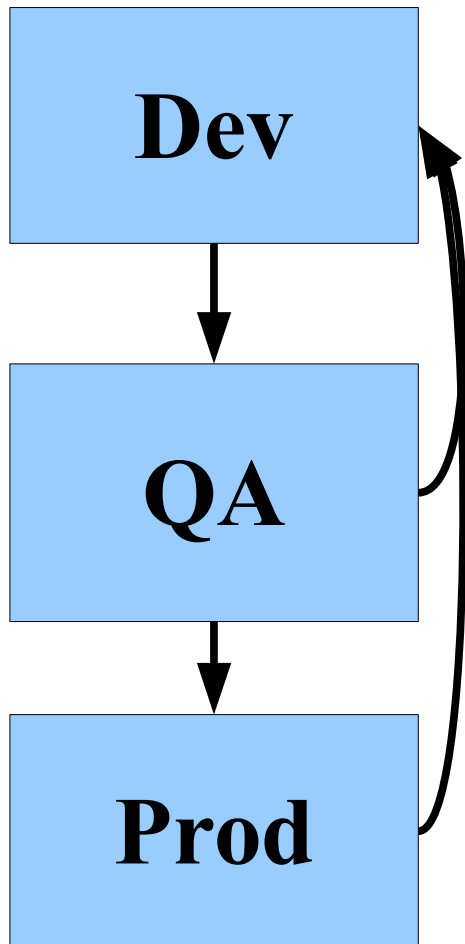
- Infrastructure as code
- Why? (All the 'ables')
 - Shareable
 - Repeatable
 - Fixable (Continuous Improvement)
 - “Version-able”
 - Testable
 - “Automate-able”
 - Scalable

Continuous Delivery



- Process not a tool (tools sure help though!)
 - Repeatable processes
 - Code versioning
 - Automation
 - Development environment
 - Development process
 - Be bold and stupid! (let the process save you)
 - **Continuous Delivery** - great book for philosophy not so much for tactical examples :(

Development Process



Development Environment

- Take lots of risks here
- Doesn't matter if it breaks
- Easy to restore good state from Production
- Typically this is on your own workstation but could also be a server hosted website

Quality Assurance Environment

- Test procedures documented in Dev to make sure it works
- If a change fails go back to Dev testing and fix it
- Typically this is on your own workstation but could also be a server hosted website
- I often call it Production +1

Production

- No changes here without going through Dev & QA process
- Has tested backup and backout procedures
- Implement procedures that passed QA testing
- If a change fails go back to Dev testing and fix it
- This is your production web server

Problem: Need Virtual Xubuntu

Solution: DevOps!



- Provide repeatable improvable (all the 'ables) Xubuntu LTS image to CSC faculty, labs, and Web Team.
- Install required software automatically (LAMP, Eclipse, NetBeans, etc...)
- LDAP logins using NCC User ID and password
- Map network drives
- Other functions as requested by CSC and WebTeam
- Use Eclipse, Git, VirtualBox, Vagrant, Xubuntu, LibreOffice to created automated build process.

NCC Xubuntu 16.04 LTS Virtual Appliance



- Install Git, VirtualBox and Vagrant (Win, Mac, or Linux)
- Github <https://github.com/chadarius/nccxubuntu>
 - `cd` to your workspace
 - `git clone https://github.com/Chadarius/nccxubuntu`
 - create/edit conf files
 - `vagrant up`
 - `vagrant ssh` (if needed)
 - Enjoy the NCC Xubuntu build!
 - enjoyment may vary depending on your conf files. Trust me. It probably won't work for you out of the box! Just use this as an example:)



Open Source

Short Version



- Richard Stallman – The father of Open Source
 - MIT sysadmin who just wanted to share printer drivers
- Copyleft instead of Copyright
 - <http://www.gnu.org/copyleft/gpl.html>
- OSI - <http://opensource.org>
 - Find their definition of Open Source
- Free as in beer vs. Free as in Freedom
- Common Open Source Licenses
 - GPL, LGPL, Apache, BSD



Some rights reserved by chrys



Tools



- Code Versioning (does anyone use anything other than Git?)
- Repos as a service - Github, Bitbucket, Gitlab
- Roll your own - **Gitlab community edition**
- Does it have a ticket/error tracking tool? Useless without it.



High compression ratio

Tools 7-Zip



- The best file archive utility out there!
- You need this to download and extract zip, tar, tar.gz, and 7z files.
- Mac and Linux can handle .zip and .tar.gz files natively, but do need a utility to handle 7z files.
- [Mac 7zip software here](#)

Secure Shell/SCP



- **Putty/SSH** – Remote command line utility. Used to manage your hosted website.
 - SSH is encrypted and very secure.
 - Replaces telnet, which is not encrypted and not secure. telnet is plain text over the network
- **Filezilla** - Graphical file management utility that uses the SSH protocol to manage files. Replaces FTP.
- SSH is built into Mac/Linux/Windows 10 (with Ubuntu bash shell)

Tools



- IDE (Integrated Development Environment)
 - [Eclipse](#) for development
 - [Notepad ++](#) for general use
- Graphics and Layout
 - Go free with
 - [GIMP](#) (Photoshop), [Inkscape](#) (Illustrator)
- Office Suite
 - LibreOffice - Doc, presentations, spreadsheets. Everything you need for free.

Virtual Tools



- VirtualBox
 - Run all kinds of operating systems and browser versions for testing. Run Windows on your Mac or Linux workstation or vice-versa!
- VMWare Free but not open source
- Vagrant - code your VMs!
- Manage your VMs - [Saltstack](#) or [Ansible](#) (or go old school with [Puppet](#) or [Chef](#))
 - Saltstack is my favorite. Well managed project and uses Python. Easy to add features.

What OS Should I Use?



- Use the OS you deploy on!
 - Almost no one deploys on Mac OS just stop it!
 - Minimum you need to install MacPorts and continually mess with your environment to keep things running. Yuck.
 - Unless you code for iOS, Macs are overpriced, underpowered, self entitled waste of everyone's time now. Save yourself \$1000 and buy a Windows laptop and slap Xubuntu on there OK?
 - Linux (and some Windows) are what most people deploy to.
 - More native tools and capabilities for updating and developing
 - Check out [Chocolatey](#) for Windows package management!
 - You have virtual capabilities! Use them!
 - I run Xubuntu locally and virtualize everything else

How to get started?

Stop mousing and start scripting!



- Example Documentation - NCC Xubuntu 16.04 LTS
 - If you can't document it you can't automate it
- Code counts as documentation! (as long as it is well documented)
 - start slamming out bash scripts, batch files, powershell scripts, and python!
 - Even if you just script bits and pieces, eventually you can improve your solution to being fully automated.
 - Create Git repos for everything