Yinlin Chen
University Libraries, Virginia Tech

Efficient development and deployment of Hydra projects using Vagrant

Outline

- Introduction
- The problem
- The concept
- Our solution
- The usage
- Future plan

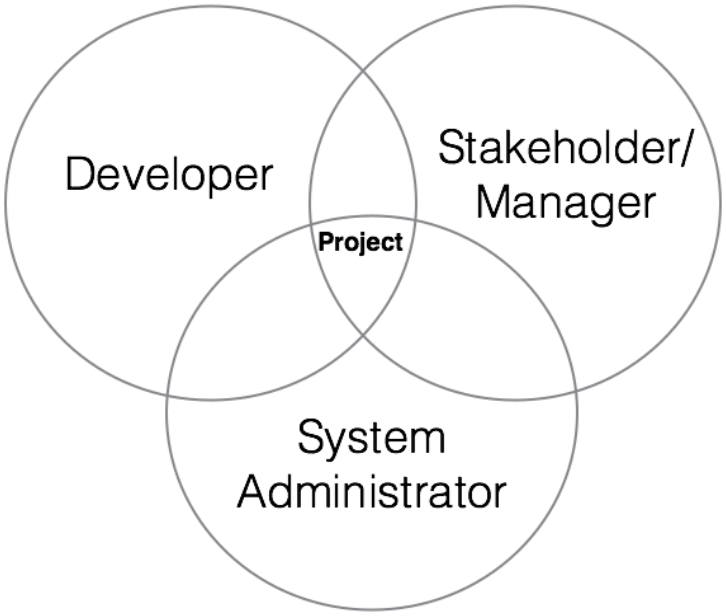
University Libraries Projects

- VTechData (https://data.lib.vt.edu): Sufia 6
- ETDPlus (http://etdplusdemo.educopia.org): Sufia 6
- GEOBlacklight (https://geodata.lib.vt.edu)
- IAWA (https://iawadev.lib.vt.edu): Sufia 7

Projects timeline

- VTechData: 1.5 year
- ETDPlus: 1 year
- GEOBlacklight: 6 months
- IAWA: 3 months

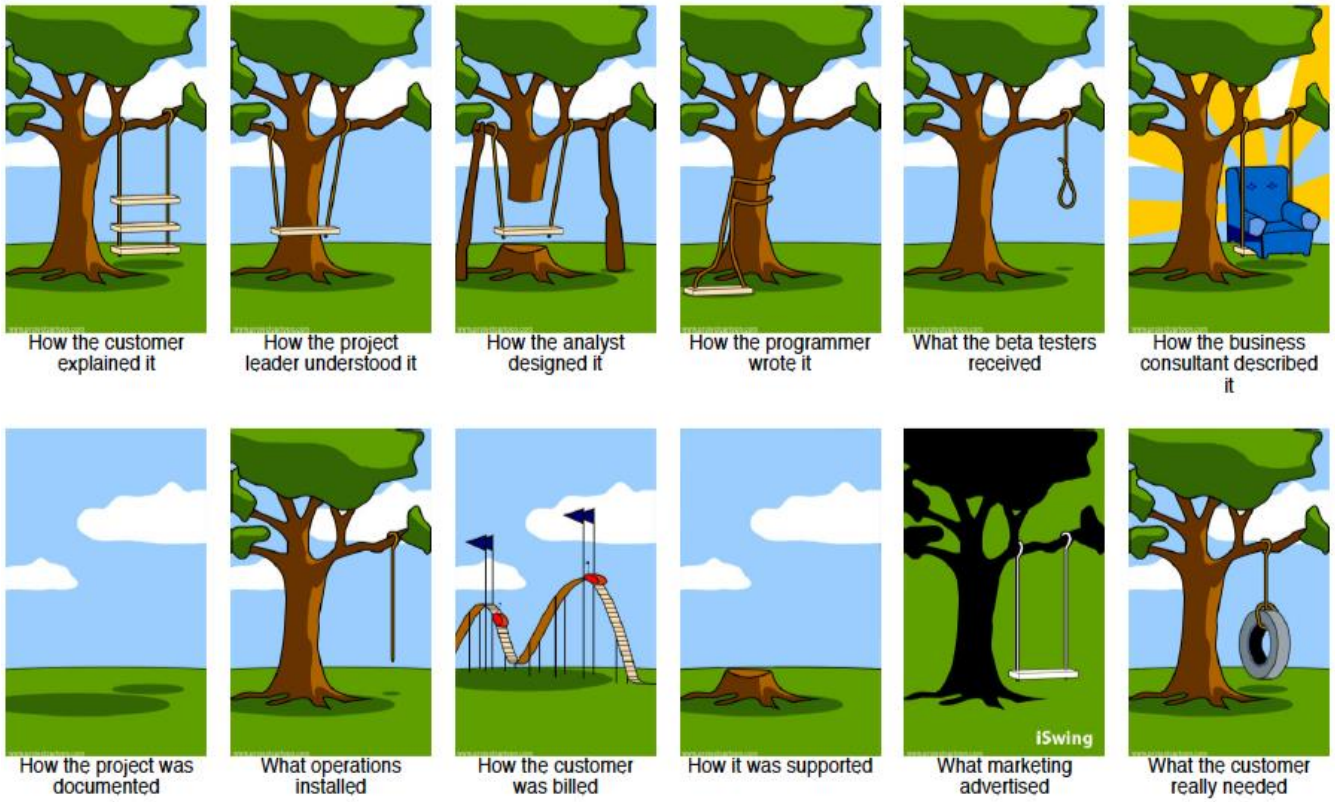
The problem



Time is Money

- Project requirement
- Development environment setup
- Deploy application into production environment

Project requirement



How to make it more efficiently?

- Repeat communication
- Show something during the development
- Early delivery

Help them to know what they really want!

Development Env setup

- Various OS
- Various software versions
- Various programming habits
- Installation is not “THAT” easy

GeoBlacklight installation

Installation

Creating a new GeoBlacklight application from the template

```
$ rails new app-name -m https://raw.githubusercontent.com/geoblacklight/geoblacklight/master/template.rb
```

To launch app:

```
$ cd app-name
$ rake geoblacklight:server
```

```
Gem files will remain installed in /var/folders/zr/hchc67ln4n19x3wbnw_0nnc40000gn/T/bundler20160922-4824-c3wsqfsqlite3-1.3.11/ge
ms/sqlite3-1.3.11 for inspection.
Results logged to /var/folders/zr/hchc67ln4n19x3wbnw_0nnc40000gn/T/bundler20160922-4824-c3wsqfsqlite3-1.3.11/gems/sqlite3-1.3.11
/ext/sqlite3/gem_make.out
Installing turbolinks-source 5.0.0
An error occurred while installing json (1.8.3), and Bundler cannot continue.
Make sure that `gem install json -v '1.8.3'` succeeds before bundling.
    generate    blacklight:install
Could not find gem 'sass-rails (~> 5.0)' in any of the gem sources listed in your Gemfile or available on this machine.
Run `bundle install` to install missing gems.
    generate    geoblacklight:install
Could not find gem 'sass-rails (~> 5.0)' in any of the gem sources listed in your Gemfile or available on this machine.
Run `bundle install` to install missing gems.
        rake    db:migrate
Could not find gem 'sass-rails (~> 5.0)' in any of the gem sources listed in your Gemfile or available on this machine.
Run `bundle install` to install missing gems.
         run  bundle install
Fetching gem metadata from https://rubygems.org/
Fetching version metadata from https://rubygems.org/
Fetching dependency metadata from https://rubygems.org/
Resolving dependencies.....
Rubygems 2.0.14.1 is not threadsafe, so your gems will be installed one at a time. Upgrade to Rubygems 2.1.0 or higher to enable
 parallel gem installation.
Using rake 11.3.0
Using i18n 0.7.0
Installing json 1.8.3 with native extensions
```

Sufia installation

Prerequisites

Sufia 7.x requires the following software to work:

1. Solr version >= 5.x (tested up to 6.2.0)
2. Fedora Commons digital repository version >= 4.5.1 (tested up to 4.6.0)
3. A SQL RDBMS (MySQL, PostgreSQL), though **note** that SQLite will be used by default if you're looking to get up and running quickly
4. Redis, a key-value store
5. ImageMagick with JPEG-2000 support
6. FITS version 0.8.x (0.8.5 is known to be good)
7. LibreOffice

Ruby installation

Homebrew (OS X)

On OS X El Capitan, Yosemite, and Mavericks, Ruby 2.0 is included. OS X Mountain Lion, Lion, and Snow Leopard ship with Ruby 1.8.7.

Many people on OS X use Homebrew as a package manager. It is really easy to get a newer version of Ruby using Homebrew:

```
$ brew install ruby
```

OS X Sierra

```
checking whether we are cross compiling... configure: error: in `/private/tmp/libyaml-20160922-5392-1ezide5/yaml-0.1.6':
configure: error: cannot run C compiled programs.
If you meant to cross compile, use `--host'.
See `config.log' for more details

READ THIS: https://git.io/brew-troubleshooting
If reporting this issue please do so at (not Homebrew/brew):
  https://github.com/Homebrew/homebrew-core/issues

Warning: You are using OS X 10.12.
We do not provide support for this pre-release version.
You may encounter build failures or other breakages.
```

How to make it more efficiently?

- As a programmer, coding is the top priority, not software installation
- Use only one programmer to prepare the development environment for all.
- Work with system administrator to prepare this development environment.

Deploy into Production Env

- Can’t just deploy application into the production environment

Developer	System Administrator
OSX	Ubuntu
Jetty (Tomcat / Solr)	Tomcat, Solr
Sqlite	RDBMS
Local network, port forwarding	NAT, Load balancer, etc

How to make it more efficiently?

- Make the development environment is almost the same with production environment
  - Be “minimal” the difference
- Work with software developer to prepare this development environment.

Not just a single project

- We always have multiple projects to work on
- One Vagrant box, many Git repositories (Projects)
  - With simply edit configuration files

The Concept

- Let developer focus on coding!
- Let system administrator focus on software installation, hardware config and network settings!
- Let stakeholder can see “Something” whenever they want to see!

The Concept

- All programmers use exactly the same development environment
- There is not much difference between development and production environment
- Switch between projects on demand
- Anyone can use it

InstallScripts

https://github.com/VTUL/InstallScripts

- For Hydra and Ruby on Rails applications
- Vagrant (https://www.vagrantup.com/)
- Ansible (https://www.ansible.com/)

InstallScripts Features

- Support both public/private GitHub repository
- Fully configurable
- Cloud Computing
  - Amazon Web Services (AWS) https://aws.amazon.com/
  - OpenStack in the Chameleon Cloud https://www.chameleoncloud.org/

Usage – Start Vagrant

- Local machine
  - Vagrant up
- Amazon Web Services (AWS)
  - vagrant plugin install vagrant-aws
  - vagrant up —provider aws
- OpenStack in the Chameleon Cloud
  - vagrant plugin install vagrant-openstack-provider
  - vagrant up —provider openstack

Services and ports

Local VM

Local	VM	Description
8983	8983	Solr services
8888	8080	Tomcat
8080	80	Application (HTTP)
4443	443	Application (HTTPS)

AWS/OpenStack/Production

- http(s)://$SERVER_HOSTNAME or http(s)://IP

Vagrant commands

- Start the machine
  - vagrant up
- Shuts down the running machine
  - vagrant halt
- Shuts down the machine and then start
  - vagrant reload
- Destroy the machine
  - vagrant destroy
- Connect to VM
  - vagrant ssh

Project configuration

- site_secrets.yml
  - project_name
  - project_git_identifier: 'master’
  - project_app_env: 'development’
  - project_db_name
  - project_solr_url
  - project_fedora_url
  - project_git_url

https://github.com/VTUL/InstallScripts/blob/master/ansible/example_site_secrets.yml

Security

- Default “vagrant” is sudoer
- Default “ubuntu” is sudoer in AWS
- Default “cc” is sudoer in OpenStack
- Use a custom user instead. e.g. hydra
- See:example_site_secrets.yml

VM Sync folder

- To use local sync folders, set the following parameters in ansible/site_secrets.yml to these values:

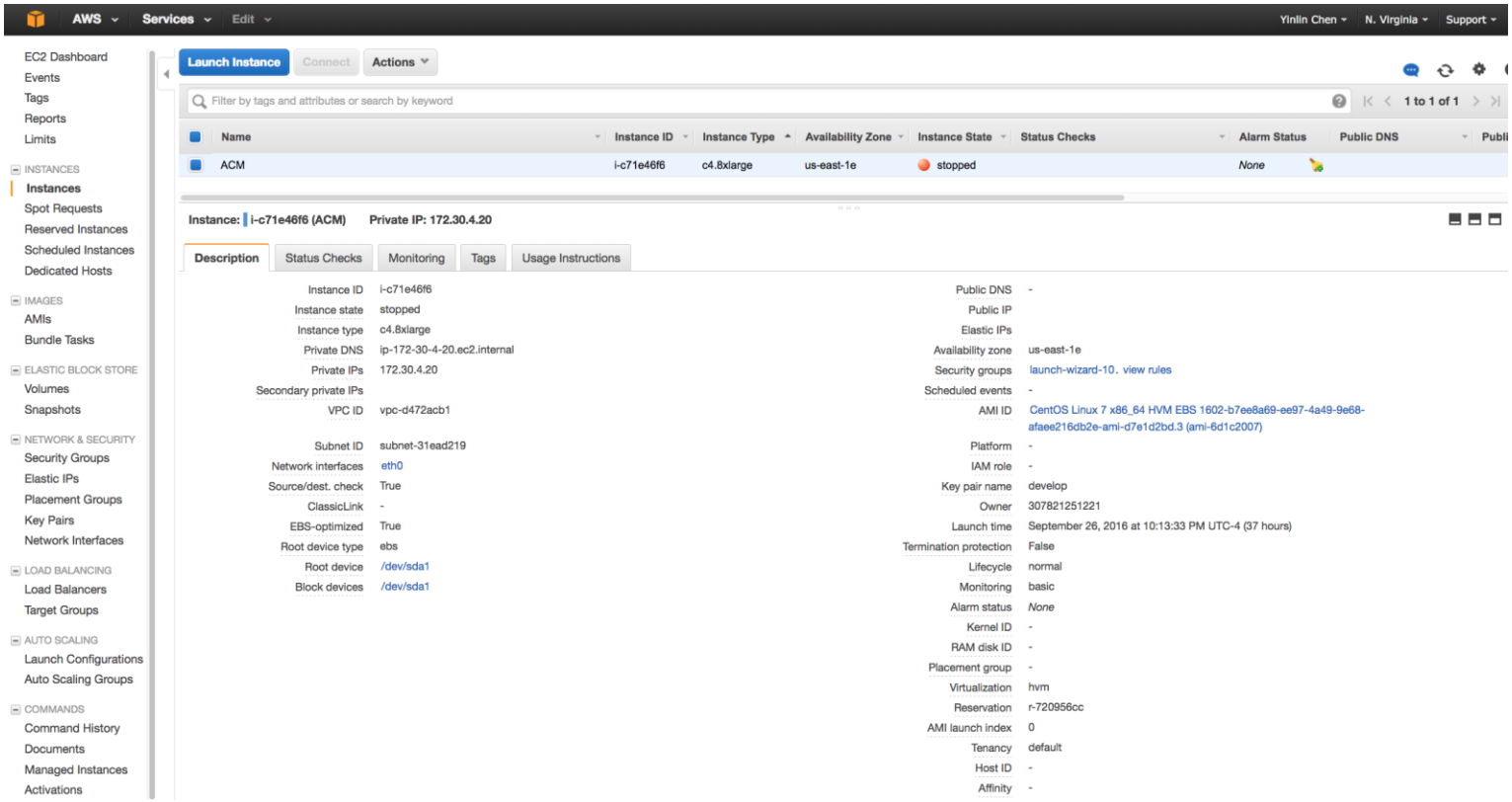
```
project_user: 'vagrant'
project_user_home: '/home/{{ project_user }}'
project_app_root: '/vagrant/src/{{ project_name }}'
```

- Sync folder located in the InstallScripts/src directory

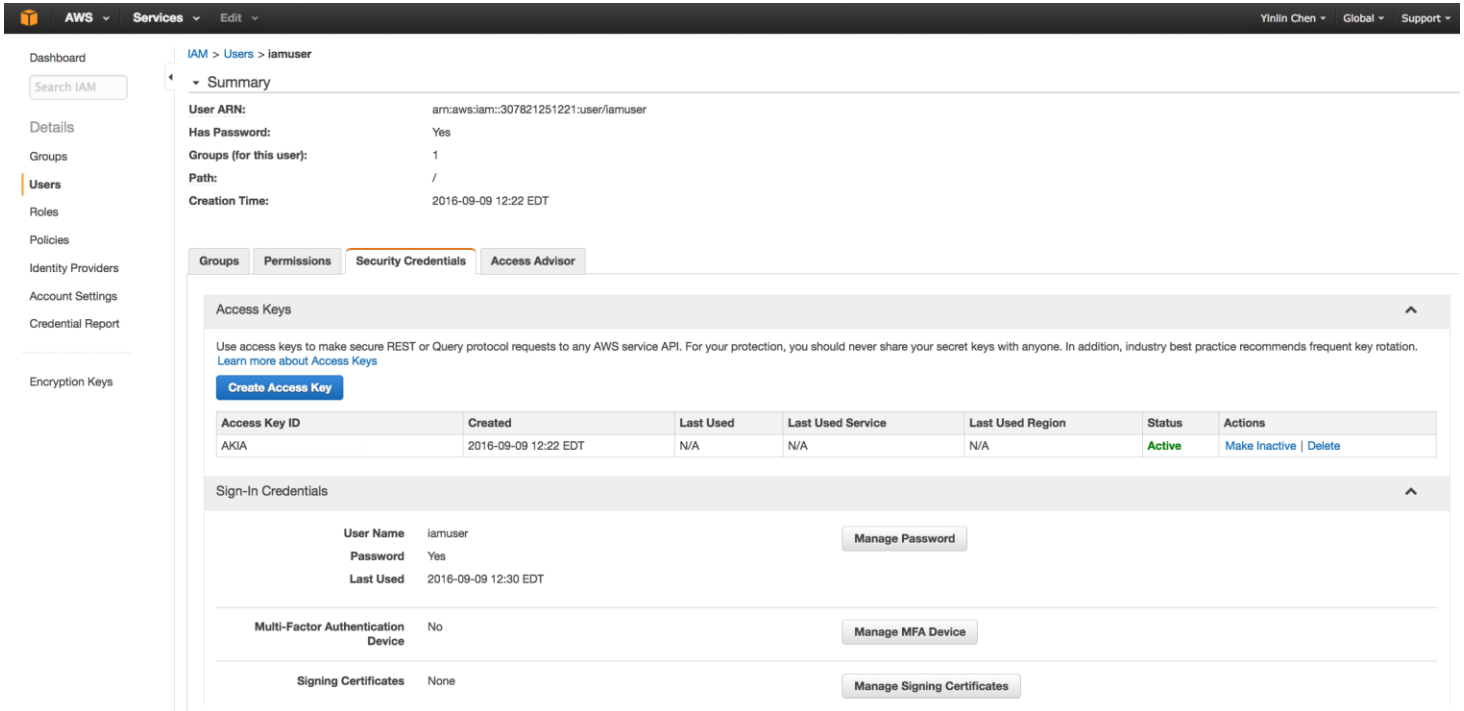
AWS launch configuration

- **KEYPAIR_NAME**: the name of the AWS keypair that will be used to log in to the instance. This keypair should already exist within your AWS account and its private key file should reside on the local system.
- **KEYPAIR_FILE**: the pathname of the private key on the local system corresponding to the aforementioned keypair.
- **AWS_ACCESS_KEY**: the AWS IAM access key to the account under which the EC2 instance will be created.
- **AWS_SECRET_KEY**: the AWS IAM secret key to the account under which the EC2 instance will be created.
- **AWS_SECURITY_GROUPS**: a space-separated list of existing AWS security groups to apply to this instance. (If AWS_SECURITY_GROUPS is not set then a default security group is used.)

AWS



AWS



OpenStack configuration

- **KEYPAIR_NAME**: the name of the OpenStack keypair that will be used to log in to the instance. This keypair should already exist within your OpenStack account and its private key file should reside on the local system.
- **KEYPAIR_FILE**: the pathname of the private key on the local system corresponding to the aforementioned keypair.
- **OS_FLOATING_IP**: the floating IP address (as a "dotted quad", i.e., x.x.x.x) to be assigned to this instance. This floating IP must already be available to the OpenStack project under which the instance is being deployed.
- **OS_SECURITY_GROUPS**: a space-separated list of existing OpenStack security groups to apply to this instance. (If OS_SECURITY_GROUPS is not set then a default security group is used.)
- **OS_USERNAME**: your OpenStack user name
- **OS_PASSWORD**: your OpenStack login password
- **OS_AUTH_URL**: the URL of the OpenStack endpoint
- **OS_TENANT_NAME**: the ID of your OpenStack Chameleon Cloud project (tenant)
- **OS_REGION_NAME**: the OpenStack region in which you wish to deploy the instance

OpenStack RC file

Chameleon | CH-816963 | ylchen

Project

Compute

- Overview
- Instances
- Volumes
- Images
- Access & Security

Network

Identity

Access & Security

Security Groups | Key Pairs | Floating IPs | API Access

Download OpenStack RC File | Download EC2 Credentials | + View Credentials

Service	Service Endpoint
Compute	https://openstack.tacc.chameleoncloud.org:8774/v2/CH-816963
Network	https://openstack.tacc.chameleoncloud.org:9696
Volumev2	https://openstack.tacc.chameleoncloud.org:8776/v2/CH-816963
Computev3	https://openstack.tacc.chameleoncloud.org:8774/v3
Image	https://openstack.tacc.chameleoncloud.org:9292
Volume	https://openstack.tacc.chameleoncloud.org:8776/v1/CH-816963
EC2	https://openstack.tacc.chameleoncloud.org:8773/services/Cloud
Identity	https://openstack.tacc.chameleoncloud.org:5000/v2.0

Displaying 8 items

Scripts vs Ansible

```
==> default: Installing rack-test 0.6.3
==> default: Installing warden 1.2.6
==> default: Installing mime-types 3.1
==> default: Installing autoprefixer-rails 6.4.0.1
==> default: Installing uglifier 3.0.1
==> default: Installing binding_of_caller 0.7.2 with native extensions
==> default: Installing sprockets 3.7.0
==> default: Installing coffee-script 2.4.1
==> default: Installing faraday 0.9.2
==> default: Installing turbolinks 5.0.1
==> default: Installing sdoc 0.4.1
==> default: Installing activesupport 4.2.6
==> default: Installing loofah 2.0.3
==> default: Installing mail 2.6.4
==> default: Installing bootstrap-sass 3.3.7
==> default: Installing solr_wrapper 0.15.0
==> default: Installing rails-deprecated_sanitizer 1.0.3
==> default: Installing globalid 0.3.7
==> default: Installing activemodel 4.2.6
==> default: Installing deprecation 1.0.0
==> default: Installing config 1.2.1
==> default: Installing jbuilder 2.6.0
==> default: Installing rails-html-sanitizer 1.0.3
==> default: Installing rails-dom-testing 1.0.7
==> default: Installing activejob 4.2.6
==> default: Installing activerecord 4.2.6
==> default: Installing actionview 4.2.6
==> default: Installing actionpack 4.2.6
==> default: Installing actionmailer 4.2.6
==> default: Installing kaminari 0.17.0
==> default: Installing railties 4.2.6
==> default: Installing sprockets-rails 3.1.1
==> default: Installing jquery-rails 4.1.1
==> default: Installing coffee-rails 4.1.1
==> default: Installing responders 2.2.0
==> default: Installing font-awesome-rails 4.6.3.1
==> default: Installing rails 4.2.6
==> default: Installing sass-rails 5.0.6
==> default: Installing web-console 2.3.0
==> default: Installing twitter-typeahead-rails 0.11.1.pre.corejavascript
==> default: Installing devise 4.1.1
==> default: Installing blacklight 6.4.1
==> default: Installing devise-guests 0.5.0
==> default: Using geoblacklight 1.1.2 from https://github.com/geoblacklight/geoblacklight.git (at 69144fb@69144fb)
==> default: Bundle complete! 18 Gemfile dependencies, 82 gems now installed.
==> default: Use `bundle show [gemname]` to see where a bundled gem is installed.
==> default: Post-install message from rdoc:
==> default: Depending on your version of ruby, you may need to install ruby rdoc/ri data:
==> default:
==> default: <= 1.8.6 : unsupported
==> default:  = 1.8.7 : gem install rdoc-data; rdoc-data --install
==> default:  = 1.9.1 : gem install rdoc-data; rdoc-data --install
==> default: >= 1.9.2 : nothing to do! Yay!
==> default: + '[' '!' -f config/secrets.yml ']'
==> default: + cat
==> default: ++ openssl rand -hex 64
==> default: + sudo -H -u vagrant RAILS_ENV=development bundle exec rake db:setup
```

```
TASK [geoblacklight : geoblacklight requires a newer version of RubyGems] ******
changed: [hydravm]

TASK [geoblacklight : copy data repo site configuration file] ******************
changed: [hydravm]

TASK [geoblacklight : enable passenger site] ***********************************
changed: [hydravm]

TASK [geoblacklight : make sure project user owns application] *****************
changed: [hydravm]

TASK [geoblacklight : clone the geoblacklight repo] ****************************
changed: [hydravm]

TASK [geoblacklight : copy the secrets file] ***********************************
changed: [hydravm]

TASK [geoblacklight : install production gems] *********************************
skipping: [hydravm]

TASK [geoblacklight : install development gems] ********************************
changed: [hydravm]

TASK [geoblacklight : load db schema] ******************************************
changed: [hydravm]

TASK [geoblacklight : precompile] **********************************************
skipping: [hydravm]

TASK [geoblacklight : stop solr] ***********************************************
changed: [hydravm]

TASK [geoblacklight : create blacklight solr core] *****************************
changed: [hydravm]

TASK [geoblacklight : ensure core name in core.properties] *********************
changed: [hydravm]

TASK [geoblacklight : copy solr config files] **********************************
changed: [hydravm]

TASK [geoblacklight : ensure solr owns the conf files] *************************
changed: [hydravm]

TASK [geoblacklight : start solr] **********************************************
changed: [hydravm]

TASK [geoblacklight : configure cron to ingest content periodically via rake task] ***
changed: [hydravm]
```

Ansible introduction

- Roles
  - Defaults: define default value for variable used by role.
  - Files: static files
  - Handlers: Ansible commands (notify. Stop/start/restart)
  - Meta: define dependency on other roles
  - Tasks: actions the role performs
  - Templates: files with variable substitution

Inside InstallScripts

- Sufia – 6 and 7
- GeoBlacklight
- Solr
- Fedora
- PostgreSql
- Prerequisites software
- Custom software

For your project

- Create a base git repo
- Update project_git_url in site_secrets.yml
- Customize tasks/{{project}}.yml
  - E.g.: ansible/roles/sufia/tasks/main.yml#L102
- Vagrant up

Future Plan

- Use Docker to manage applications
- Automate Docker with Ansible

Thank you!

- University Libraries DLD Team
- Hydra Connect Program Committee
- Contact: ylchen@vt.edu

Q & A

Demo

- AWS demo
- OpenStack demo