

Modern Puppet for the Jenkins project

A power point by R Tyler Croy

**Mr. Kanies 6th Period
GitHub Decentral High School 2014**

<http://tiny.cc/MrKaniesIsTheBest>

Who is Tyler!

- GitHub Decentral High Senior (GO CATS!)
- Member of the Ruby Pep Band
- Mathelete
- DevOps Lab Safety Marshal
- Open Source Debate Team Secretary

BOOKOUT

- Core Systems, Team Captain
- JRuby Club President
- AP Distributed Systems

Mr. Jenkins

- Free & open Source continuous integration server
- Built with Java but can do other things to
- Uses Puppet but didn't use to

*Modern Puppet
for the Jenkins project*

Project

Infrastructure

The Services

- Many web applications
- LDAP
- Atlassian services
- Jenkins
- IRC robots
- DNS
- MirrorBrain

The Wares

- 2 physical machines
- 3 KVM-based VMs
- Assortment of EC2/Rackspace instances
- 10+ mirrors

The People

- Zero "full-time" contributors
- Small group of Owners
- Sporadic drive-by-committers

Running the
infrastructure

At Sun/Oracle

- Scavenged machines
- Dubious locations within the company
- Questionable ownership of physical assets
- Manually managed, no audit trail/history
- Completely unreproducible

"What if we used 'PUPPET?'"

Masterless Puppet

- Needed something after leaving Sun/Oracle
- Centralized audit logging in Git repo
- Not everything is Puppetized because time
- Some parts reproducible, others special snowflakes
- Code structure is messy and bad, also awful

The Year 3000

- Puppet Enterprise
- Clearer definition of separation between Puppet modules and "our code"
- Git log of changes
- Some snowflakes killed, some new ones

PUPPET ENTERPRISE

Why Puppet Enterprise

- Higher visibility into changes being applied
 - Reporting (email, irc)
 - PE Console
- MCollective for basic ad-hoc orchestration
- Debian/Yum repo for agent bootstrap
- Easy breezy beautiful, cover girl

Special Thanks to Puppet Labs



*Modern Puppet
for the Jenkins project*

C:\ PuppetCode.bat

Masterless Puppet

- 80 .pp files
- 0 tests
- github.com/jenkinsci/infra-puppet
- librarian-puppet for dependency "management"
- All secrets managed out of band

```
infra-puppet/  
├── Gemfile  
├── Gemfile.lock  
├── Puppetfile  
├── Puppetfile.lock  
├── confluence.deb/  
│   └── build.sh  
├── features/  
├── local-modules/  
│   └── apache2  
├── manifests/  
│   ├── cabbage.pp  
│   ├── cucumber.pp  
│   ├── eggplant.pp  
│   ├── kale.pp  
│   ├── lettuce.pp  
│   ├── spinach.pp  
│   └── vagrant.pp  
└── run.sh
```

Code Layout

```
class base { A "module"
  # Unfortunately this module only supports Ubuntu
  if ($operatingsystem == 'Ubuntu') {
    include nagios::client
  }
}
```

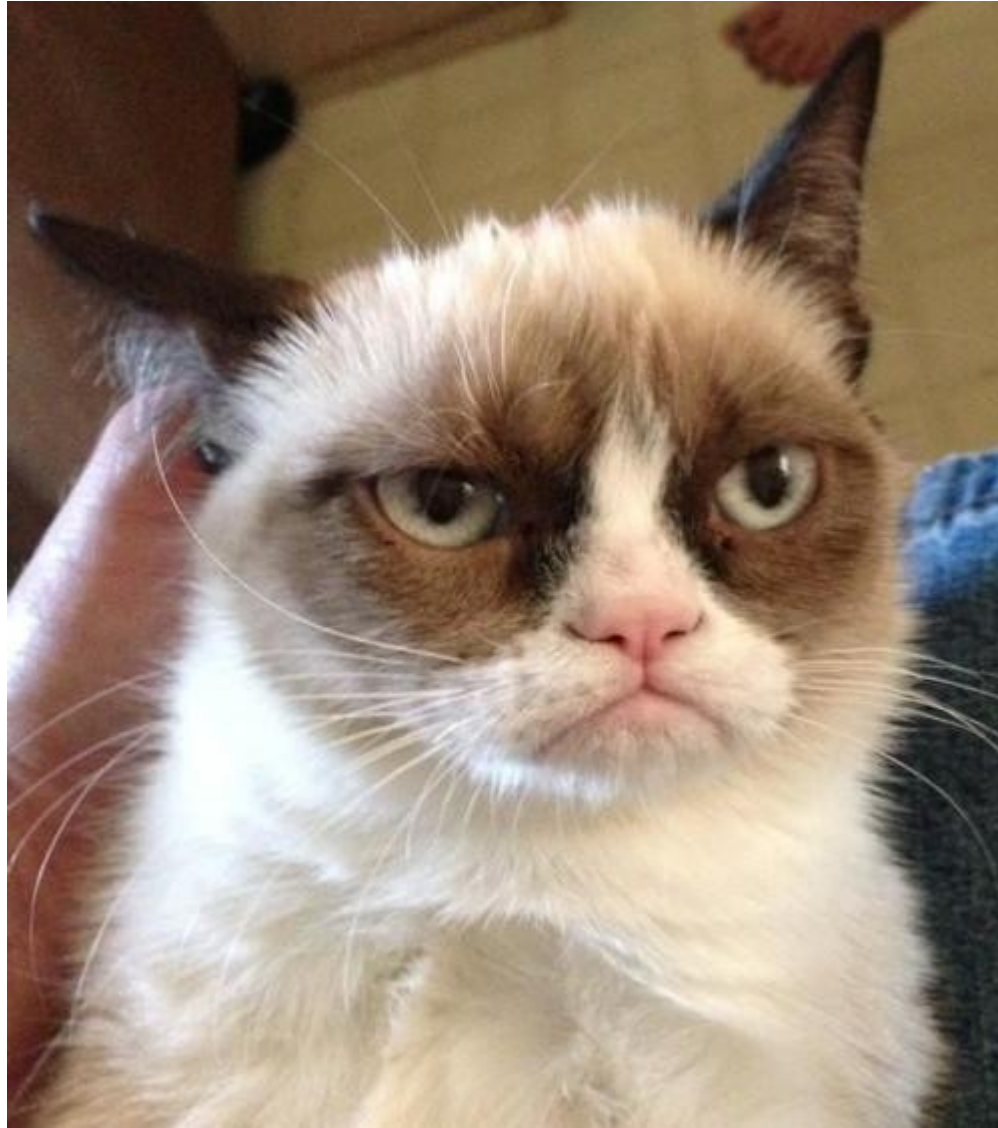
```
stage {
  'pre' :
    before => Stage['main'];
  'post' :
    require => Stage['main'];
}
```

```
class {
  'base::pre' :
    stage => 'pre';

  'base::post' :
    stage => 'post';
}
```

"deployment" aka "testing"

- `git pull --rebase`
- `librarian-puppet install`
- `puppet apply manifests/`hostname`.pp`
- manual guess and check



A BETTER WAY

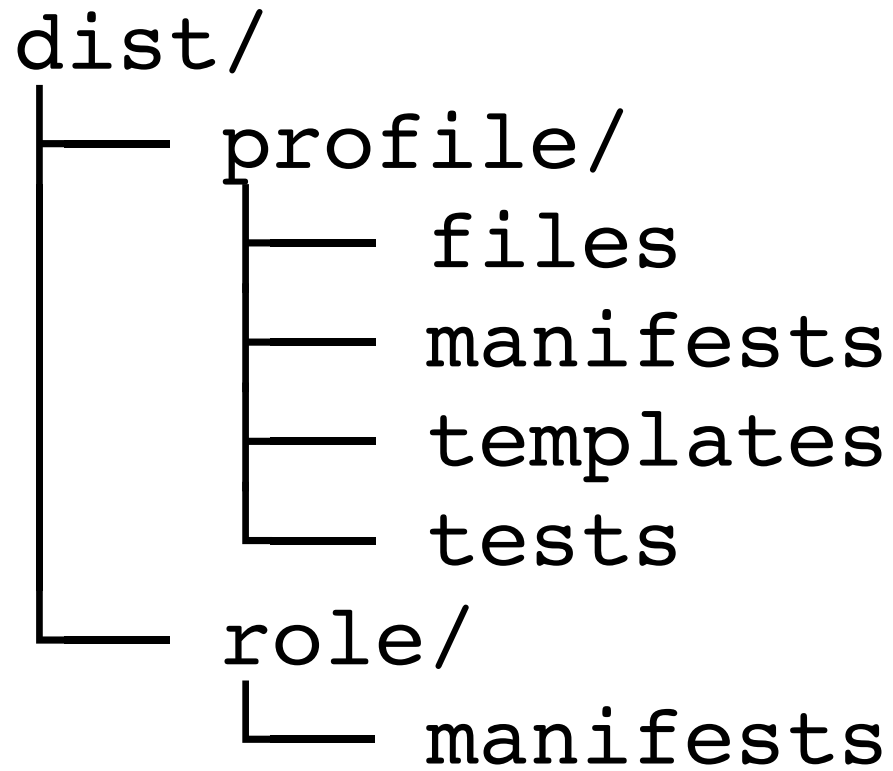
```
jenkins-infra/  
├── Gemfile  
├── Gemfile.lock  
├── Puppetfile  
├── README.md  
├── Rakefile  
├── Vagrantfile  
├── ci/  
│   └── 00_setupgems.sh  
├── dist/  
│   ├── role/  
│   └── profile/  
├── hieradata/  
│   └── common.yaml  
├── manifests/  
│   └── site.pp  
└── spec/  
    ├── classes/  
    └── server/
```

Code Layout

ci/

```
for f in ci/*.sh; do \  
    echo "> Running ci/$f"; \  
    bash $f; \  
done
```

dist/

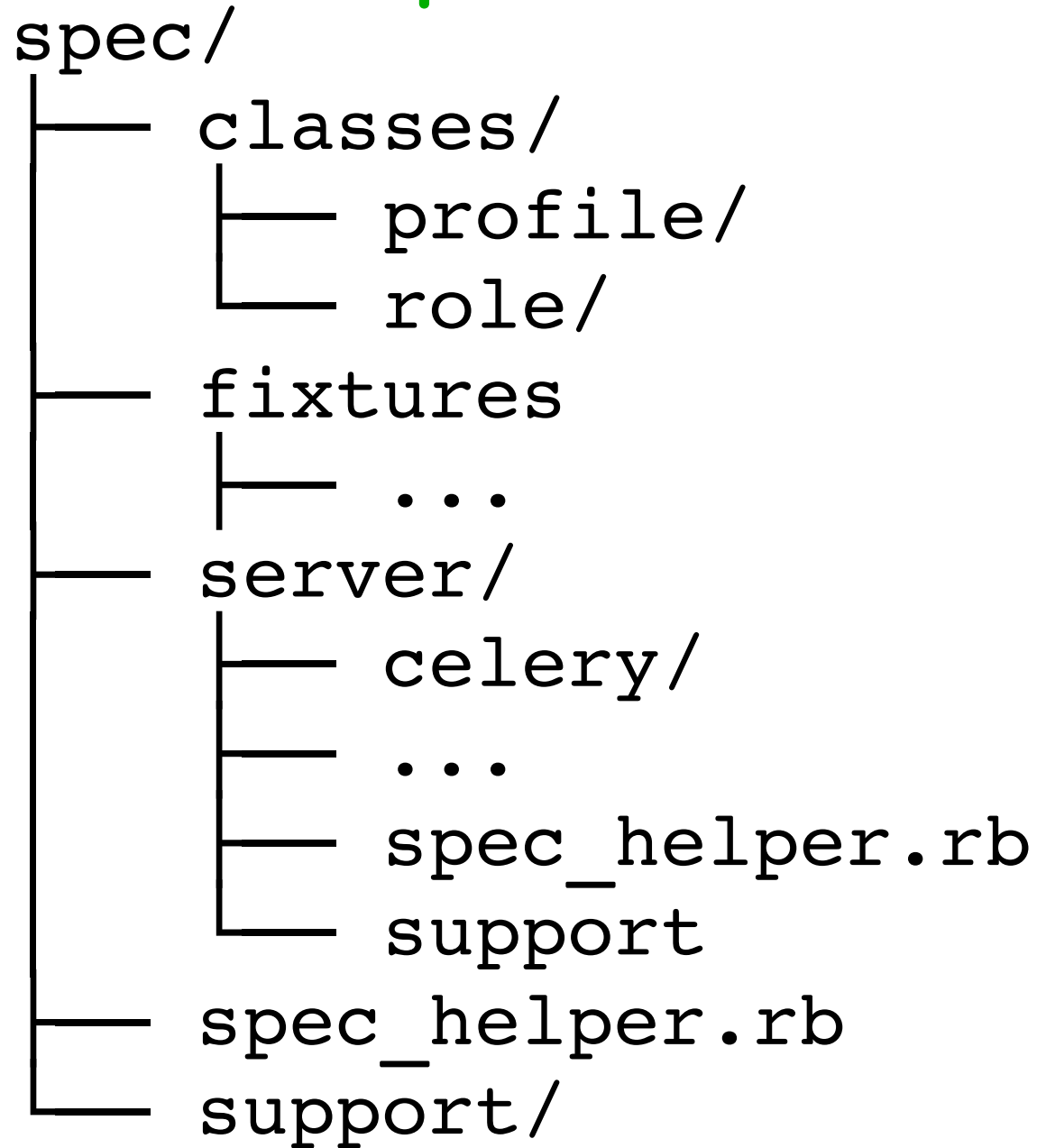


profile/manifests/

dist/profile/manifests/

- accounts.pp
- apache-misc.pp
- apt.pp
- archives.pp
- ...
- sudo.pp
- vagrant.pp

spec/



A "module"



Our modules

- puppet-groovy (*fork*)
- puppet-irc (*fork*)
- puppet-apache-log-compressor
- puppet-account (*fork*)
- GarethR-docker (*fork*)

TOP SECRETS

- All secrets in Hieradata
- Heavy use of hieradata-eyaml
- Private repository for sharing keys

Testing

- `rspec-puppet` for catalogue logic
- `serverspec` + `vagrant-serverspec` for masterless integration testing
- Manual testing with **`puppet agent -t`** for final sanity check

```
describe 'profile::archives' do
  let(:facts) {
    {:operatingsystem => 'Ubuntu',
     :osfamily => 'Debian' }
  }

  it { should contain_class 'profile::apache-misc' }
  it { should contain_class 'lvm' }
  it { should contain_class 'apache' }

  # ...

  it { should contain_apache__mod 'bw' }
end
```

Vagrant + ServerSpec

- Relies Vagrant's ability to use multiple provisioners
 - Shell
 - Puppet
 - Serverspec
- Dynamically creates a machine per role

```
require_relative '../spec_helper'

describe 'spinach' do
  it_behaves_like "a standard Linux machine"
  it_behaves_like "a DNS server"

  context 'groovy support' do
    describe file('/etc/profile.d/groovy.sh') do
      it { should be_file }
    end

    describe file('/opt/groovy-2.3.1/bin/groovy') do
      it { should be_file }
    end
  end
end
end
```

```
% ./vagrant-aws up spinach
==> spinach: Running provisioner: shell...
<SNIP>
==> spinach: Running provisioner: puppet...
<SNIP>
==> spinach: Running provisioner: serverspec...
```

spinach

behaves like a standard Linux machine

File "/etc/ssh/sshd_config"

should contain "PasswordAuthentication no"

User "kohsuke"

should exist []

should have home directory "/home/kohsuke"

Port "22"

should be listening

Cron

should have entry "20 2 * * * apt-get update"

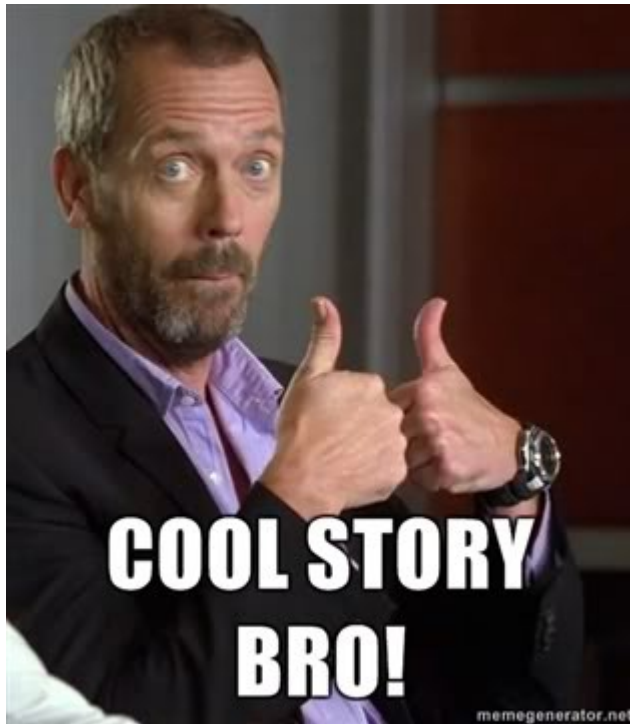
groovy support

File "/etc/profile.d/groovy.sh"

should be file

deployment

1. Create pull request
2. Merge changes into **staging** branch
3. Create pull request
4. Merge changes into **production** branch
5. R10k webhook updates master
6. Agents consume ~15 minutes
7. Report back to PE Console



OTHER COOL THINGS

Docker'd

- 3 dockered services
 - bind
 - supybot
 - custom java bot

bind/Dockerfile

```
FROM ubuntu:trusty
RUN apt-get install -y bind9 dnstools
RUN ln -f -s /etc/bind/local/named.conf.local /etc/k
RUN mkdir /var/run/named

EXPOSE 53 53/udp

# config files
VOLUME ["/etc/bind/local"]

CMD /usr/sbin/named -g
```

Quality

- 20+ spec files
- < 5% Puppet runs have failed
- All pull requests tested before merge
- > 250 successful builds in CI

SOME THINGS STILL

SUCK

Testing with Puppet Enterprise

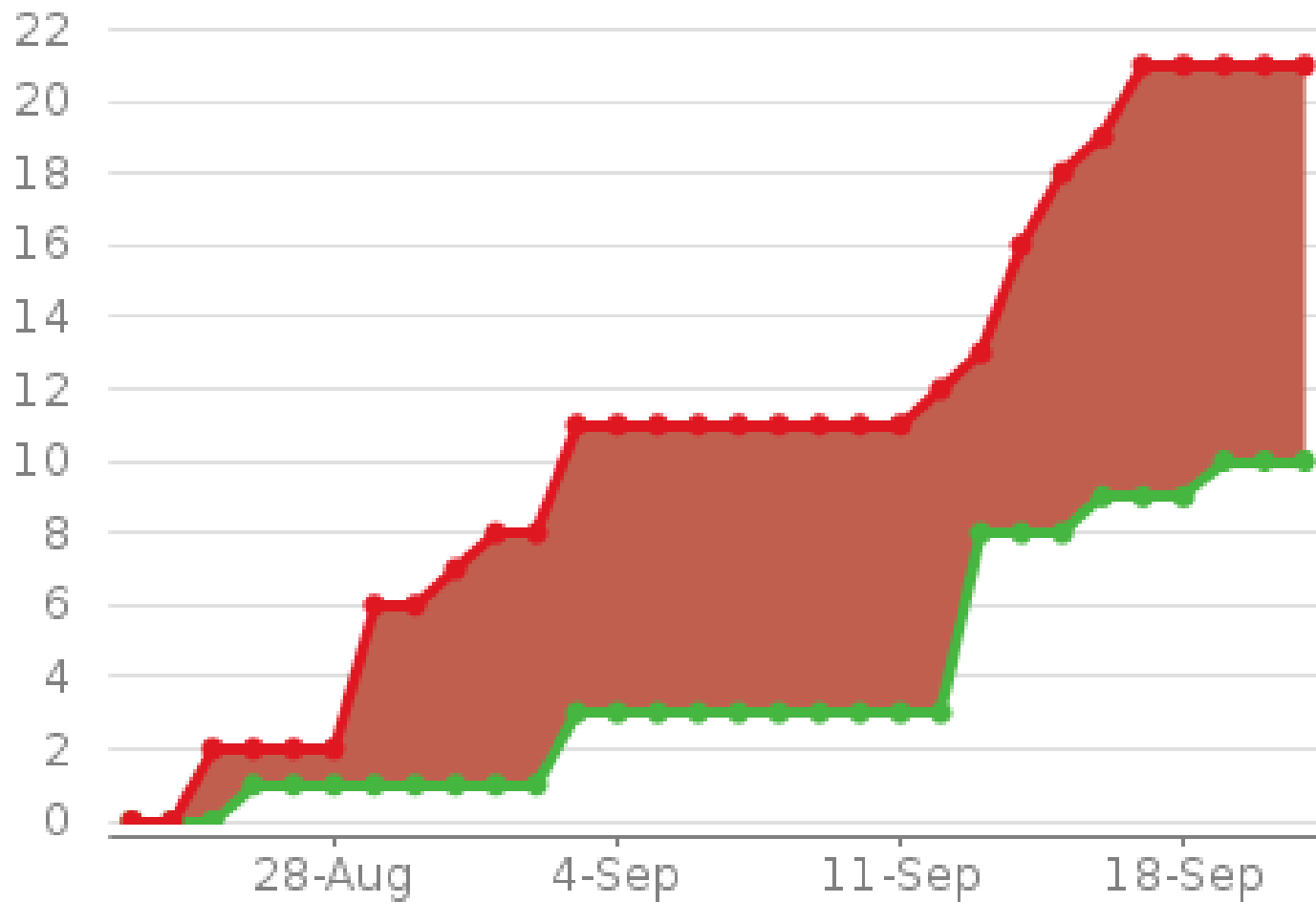
Automating Confluence and JIRA





OS UPGRADES

keeping up



you can help!

- find us on IRC: #jenkins-infra
- infra@lists.jenkins-ci.org
- github.com/jenkins-infra

THanks!



©jimbenton.com

@agentdero
github.com/rtyler
tyler@linux.com

<http://tiny.cc/MrKaniesIsTheBest>