

# Continuously Integrating Puppet

---

SNAP Interactive, Inc.

Presented by:

Matthew Barr  
Technical Architect



# Company Overview

---



- SNAP Interactive, Inc.
- Makes AYI: on Facebook, iPhone, Android, AYI.com
- One of the largest social dating platforms on the web (based on monthly active users)
- More than 5 million monthly active users
- Over 1 billion total pieces of structured data from its users
- Synced to 20+ million Facebook profiles
- Receives over 35,000 real-time updates per minute from Facebook
- Subscription-based business model
- SNAP is publicly traded - Ticker: STVI

# About me

---



- Team lead for the operations group
- Sysadmin for over 15 years
- Dev/ops, WebOps, etc.
- [LOPSA](#) & USENIX member
- @MatthewBarr - twitter, github
- [mbarr@mbarr.net](mailto:mbarr@mbarr.net)

# Continuous Integration



- The practice of frequently integrating one's new or changed code with the existing code repository

- **Maintain a code repository**

- **Automate the build**

- **Make the build self-testing**

- **Keep the build fast**

- **Everyone can see the results of the latest build**



---

# Background on the basic tools

# Puppet

---



- Puppet Labs
- <http://puppetlabs.com/>
- Configuration Mgmt tool with a DSL

# Jenkins



- An extendable open source continuous integration server
- <http://jenkins-ci.org/>

The screenshot shows the Jenkins web interface in a browser window. The browser's address bar displays "localhost:8090". The Jenkins dashboard includes a search bar, a sidebar with navigation links like "New Job", "Manage Jenkins", "People", and "Build History", and a main content area. The main area features a table with columns for "S", "W", "Job", "Last Success", "Last Failure", and "Last Duration". Below the table, there are sections for "Build Queue" (showing "No builds in the queue.") and "Build Executor Status" (showing "1 Idle" for the "Master" node). The interface also includes an "ENABLE AUTO REFRESH" button and an "add description" link.

| S | W | Job ↓               | Last Success | Last Failure | Last Duration |
|---|---|---------------------|--------------|--------------|---------------|
|   |   | <a href="#">one</a> | N/A          | N/A          | N/A           |

# So, let's do this!

---



- Setup - Dynamic Env & Jenkins
- Static Analysis: Puppet Parsing & Linting
- Module Testing
- Catalog Testing
- Dynamic Analysis: Test deployment & Integrations test
- Nightly Rebuilds



# Actual Deployment!

---



- Canary servers
- Production rollout!

# Screeeetch!



- This is about CI, but...
- CI begins at ~~home~~ work.
- Vagrant! - <http://www.vagrantup.com/>
- “Create and configure lightweight, reproducible, and portable development environments.”

# Vagrant File



```
Vagrant::Config.run do |config|

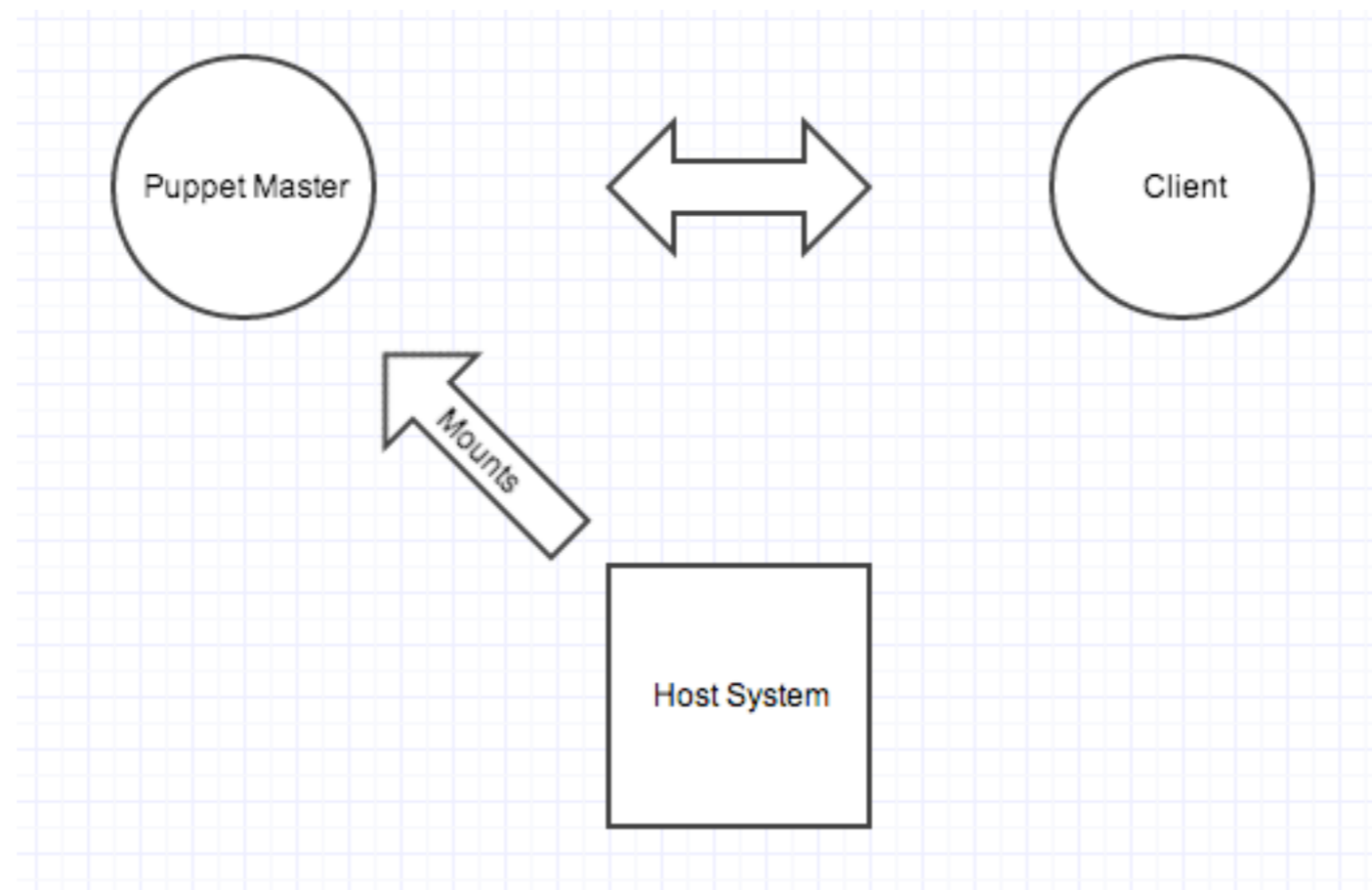
  config.vm.box = "centos"

  config.vm.define :master do |master_config|
    master_config.vm.host_name = "puppet.local"
    master_config.vm.network :hostonly, "10.100.1.10"
    master_config.vm.share_folder("puppet-modules", "/etc/puppet/modules", "/Users/mbarr/puppet/modules/")
    master_config.vm.share_folder("puppet-manifests", "/etc/puppet/manifests", "/Users/mbarr/puppet/manifests/")
    master_config.vm.share_folder("puppet-hiera", "/etc/puppet/hiera", "/Users/mbarr/puppet/hiera/")
    master_config.vm.share_folder("repos", "/var/www/html", "repos")
    master_config.vm.share_folder("centos", "/var/www/html/centos", "/Volumes/CentOS_6.3_Final/")

    master_config.vm.provision :puppet do |p|
      p.manifests_path = "manifests"
      p.manifest_file = "init.pp"
    end
  end

  config.vm.define :client do |client_config|
    client_config.vm.host_name = "client.local"
    client_config.vm.network :hostonly, "10.100.1.11"
    config.vm.provision :shell, :inline => "cp /vagrant/files/hosts.client /etc/hosts"
  end

end
```



# Pre-commit Hooks

---



- Almost all the static analysis tools make good pre-commit hooks
- in SVN, git, Hg, your favorite SCR tool



---

**Back to our regularly  
scheduled show..**

# Setup

---



- Puppet - Dynamic Environments
- Jenkins
- Github / Github Pull Request Builder

# Puppet Dynamic Env's

---



- google: puppet dynamic environments
- Git based
- Each branch is it's own environment
- Commit hook / web
- <https://puppetlabs.com/blog/git-workflow-and-puppet-environments/>
- by Adrien Thebo



# Jenkins install

---



- puppet module install rtyler-jenkins
- Puppet forge module: rtyler/puppet-jenkins
- Most up to date is in our github repo:
  - <https://github.com/jenkinsci/puppet-jenkins>
- Allows for base & plugin installs, as well as automatic slaves

# Jenkins



## Jenkins



Jenkins > Puppet CI

[DISABLE AUTO REFRESH](#)

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Project](#)

[Configure](#)

[GitHub](#)

[Project Health](#)

[Copy project](#)

[Build History](#) [\(trend\)](#)

#14 [Apr 23, 2013 4:08:23 PM](#)

[Pull request #120](#)

#13 [Apr 23, 2013 4:04:23 PM](#)

[Pull request #120](#)

## Project Puppet CI

[add description](#)

| Job                               | S | W | Last Success | Last Failure | Last Duration | Console |
|-----------------------------------|---|---|--------------|--------------|---------------|---------|
| <a href="#">Puppet CI</a>         |   |   | 5 min 7 sec  | 9 min 7 sec  | 1 min 21 sec  |         |
| <a href="#">Static Analysis</a>   |   |   |              |              |               |         |
| <a href="#">Static Analysis</a>   |   |   | 5 min 1 sec  | 9 min 0 sec  | 15 sec        |         |
| <a href="#">Compile Tests</a>     |   |   |              |              |               |         |
| <a href="#">Compile</a>           |   |   | 4 min 38 sec | N/A          | 21 sec        |         |
| <a href="#">Dynamic Analysis</a>  |   |   |              |              |               |         |
| <a href="#">Test Hosts</a>        |   |   | 4 min 10 sec | N/A          | 22 sec        |         |
| <a href="#">Integration Tests</a> |   |   | 3 min 48 sec | N/A          | 30 ms         |         |

Icon: [S](#) [M](#) [L](#)

[Legend](#) [RSS for all](#) [RSS for failures](#) [RSS for just latest builds](#)

# Github

---



- Private repos for organizations
- Pull Requests!

# GH PR Builder


---





- Monitors github for a pull request
- Triggers a Jenkins build
- Updates merge status
- (it's a Jenkins plugin. **ghprb**)

# Github Pull Request




 matthewbarr opened this pull request 3 minutes ago Edit


## Updated github pull request builder config file



No one is assigned  No milestone 

to remove the user/ pass for a api token instead.. Also, moved to v 1.7.


✓ Good to merge — Merged build finished. ([Details](#))

1 participant 


 matthewbarr added a commit 3 minutes ago

 matthewbarr Updated github pull request builder config file to remove the user/ p...  ✓ [0920ca4](#)

You can add more commits to this pull request by pushing to the `puppet_ci` branch on `SnapInteractive/Operations`

 Good to merge — Merged build finished. ([Details](#))

**This pull request can be automatically merged.**  
You can also merge branches on the [command line](#).



# Static Analysis

---



- Puppet Parser / Syntax check
  - `puppet parser validate $file`
- ERB Syntax check
  - `erb -P -x -T '-' mytempl.erb | ruby -c`
- Puppet Lint
  - <http://puppet-lint.com/>
- YAML - Hiera checks
  - Still looking for a good tool, actually.

# Static Analysis



- Catches errors in syntax & styling

```
class parser {  
  file {  
    'name':  
    ensure => file # this needs a ,  
    content => template("${module_name}/parsed.erb),  
    source => 'puppet:///modules/parser/file.txt';  
  }  
}
```

# Static Analysis



- Catches errors in syntax & styling

```
class parser {  
  file {  
    'name':  
    ensure => file # this needs a ,  
    content => template("${module_name}/parsed.erb),  
    source => 'puppet:///modules/parser/file.txt';  
  }  
}
```

```
Wintermute:ci-demo mbarr$ puppet parser validate init.pp  
Error: Could not parse for environment production: Syntax error at 'content'; expected '}' at /private/tmp/ci-demo/i  
nit.pp:8  
Error: Try 'puppet help parser validate' for usage
```



# Static Analysis



- Catches errors in syntax & styling

```
class parser {  
  file {  
    'name':  
    ensure => file # this needs a ,  
    content => template("${module_name}/parsed.erb),  
    source => 'puppet:///modules/parser/file.txt';  
  }  
}
```

```
Wintermute:ci-demo mbarr$ puppet parser validate init.pp  
Error: Could not parse for environment production: Syntax error at 'content'; expected '}' at /private/tmp/ci-demo/i  
nit.pp:8  
Error: Try 'puppet help parser validate' for usage
```

```
Wintermute:ci-demo mbarr$ puppet-lint init.pp  
ERROR: parser not in autoload module layout on line 4  
ERROR: tab character found on line 5  
ERROR: tab character found on line 6  
ERROR: tab character found on line 7  
ERROR: tab character found on line 9  
ERROR: tab character found on line 10  
ERROR: trailing whitespace found on line 5  
ERROR: two-space soft tabs not used on line 5  
ERROR: two-space soft tabs not used on line 10  
WARNING: indentation of => is not properly aligned on line 8  
Wintermute:ci-demo mbarr$
```

# Module Testing

---



- Rspec-Puppet
  - <http://rspec-puppet.com/>
- Works well for advanced testing of a single module.
- Especially useful for verifying complex internal logic.



```
require 'spec_helper'

# Note, rspec-puppet determines the define name from the top level describe
# string.
describe 'jenkins::plugin' do
  let(:title) { 'git' }

  describe "on RedHat" do
    let(:facts) do
      { :osfamily => 'RedHat' }
    end
    it { should contain_user('jenkins') }
    it { should contain_group('jenkins') }
    it { should contain_file('/var/lib/jenkins') }
    it { should contain_file('/var/lib/jenkins/plugins') }
    it { should contain_exec('download-git') }
  end
  describe "on Debian" do
    let(:facts) do
      { :osfamily => 'Debian' }
    end
    it { should contain_user('jenkins') }
    it { should contain_group('jenkins') }
    it { should contain_file('/var/lib/jenkins') }
    it { should contain_file('/var/lib/jenkins/plugins') }
    it { should contain_exec('download-git') }
  end
end
end
```

# Catalog Testing

---



- Execute a catalog compile (on the puppet master) with a specific set of facts and an environment:
- `puppet master --compile HOST --environment ENV > /dev/null`
- Another option is Ohad Levy's manitest
- <https://github.com/ohadlevy/manitest>

# Catalog Testing



```
test.sh — tmp
test.sh
1  #!/bin/bash~
2  ~
3  EXITRETURN=0~
4  ~
5  REPO="git@github.com:Example/puppet.git"~
6  BRANCH_DIR="/etc/puppet/environments/pullreq_env"~
7  ~
8  # branch is being updated~
9  echo "Testing branch $BRANCH_DIR"~
10 ~
11 { cd $BRANCH_DIR && git fetch && git checkout $1 && git submodule foreach git submodule update --init
   --recursive; } \~
12 || EXITRETURN=1~
13 ~
14 echo web01~
15 puppet master --compile web01.example.com --environment pullreq_env > /dev/null || EXITRETURN=1~
16 ~
17 # example mysql server~
18 echo mysql01~
19 puppet master --compile mysql01.example.com --environment pullreq_env > /dev/null || EXITRETURN=1~
20 ~
21 exit $EXITRETURN~
22
```

Line: 19:40 | Shell Script (Bash) | Soft Tabs: 2 | Symbols

# Catalog Testing

---



- Catches dependency loops, unknown modules, variables or facts.

# Catalog Testing



- Catches dependency loops, unknown modules, variables or facts.

```
[root@ops02 modules]# vi common/manifests/init.pp
[root@ops02 modules]# puppet master --compile web01.ayisnap.com --environment demo
Error: Could not find class abc for web01.ayisnap.com on node web01.ayisnap.com
Error: Could not find class abc for web01.ayisnap.com on node web01.ayisnap.com
Could not find class abc for web01.ayisnap.com on node web01.ayisnap.com
[root@ops02 modules]# █
```



# Dynamic Analysis

---



- Test deployments to all server types
  - See the changes to each type of server
  - catches: RPM isn't in a repo, etc
- Integration tests
  - catches: Config files that break a service



# Test Deployments



- `ssh root@testweb1 puppet agent -t --environment pullreq_env --color=false`
- `ssh root@testmysql1 puppet agent -t --environment pullreq_env --color=false`

## Parsed Console Output

 [Error \(12\)](#)

 [Warning \(0\)](#)

 [Info \(7\)](#)

```
Started by upstream project "Puppet CI" build number 4
originally caused by:
Github pull request #113 of commit cc9c0bac29a9f29e3b9ca2d794c92fa7c3636459 automatically merged.
[EnvInject] - Loading node environment variables.
Building in workspace /var/lib/jenkins/jobs/Test Hosts/workspace
[workspace] $ /bin/sh -xe /tmp/hudson2979395280641206678.sh
+ ssh root@testweb101 puppet agent -t --environment pullreq_env --color=false
Info: Retrieving plugin
Info: Loading facts in /var/lib/puppet/lib/facter/facter_dot_d.rb
Info: Loading facts in /var/lib/puppet/lib/facter/root_home.rb
Info: Loading facts in /var/lib/puppet/lib/facter/puppet_var_dir.rb
Info: Loading facts in /var/lib/puppet/lib/facter/pe_version.rb
Info: Caching catalog for testweb101.ayisnap.com
Info: Applying configuration version '1366578395'
Error: Execution of '/usr/bin/yum -d 0 -e 0 -y install php-pecl-mongo' returned 1:

Error Downloading Packages:
php-pecl-mongo-1.3.4-1.el6.x86_64: failure: php-pecl-mongo-1.3.4-1.el6.x86_64.rpm from epel: [Errno 256] No more mirrors to try.

Error: /Stage[main]/Php/Package[php-pecl-mongo]/ensure: change from absent to present failed: Execution of '/usr/bin/yum -d 0 -e 0 -y
install php-pecl-mongo' returned 1:
```

# Jenkins Log Parser

---



- Log Parser plugin needs a parsing rules file:

```
Wintermute:puppet mbarr$ cat puppet.parse
error /^Error:/
ok /^Info:/
info /^Notice: Finished/
warning /^Warning:/
warning /^Notice:/
```

# Integration Testing

---



- Covers critical functionality
- Ensures that systems still serve their function after new code has been applied
- Could be as simple as:
  - `curl http://testweb1/`
- But probably isn't.

# Nightly Rebuilds

---



- What happens when the repos / packages are updated, or a kickstart file is changed?
- Or a manifest has something deleted
- Rebuild your servers from scratch. Often.
- Keeps you confident that you can rebuild your DC & systems
- Only you can prevent bitrot!

# Nightly rebuilds



- Create a new VM & get it's MAC
- Provision it in cobbler & IPA
- Turn it on

```
require 'rubygems'~
require 'fog'~
~
new_name=ARGV[0]~
ip=ARGV[1]~
f=Fog::Compute.new(:provider => "Vsphere" )~
new_vm=f.vm_clone( 'name' => new_name, 'datacenter' => 'Cogent', 'datastore' => 'Main','template_path' =>
'canary', 'power_on'=>false)~
mac=new_vm["new_vm"]["mac_addresses"]["Network adapter 1"]~
puts mac~
~
command="ssh cobbler sudo /root/newcobbler.sh VM #{new_name} #{mac} #{ip}"~
print command +" :"-
output=system(command)~
puts output~
~
|
new_vm["new_vm"]["instance_uuid"] = new_vm["new_vm"]["id"]~
f.vm_power_on(new_vm['new_vm'])~
```



# Nightly Rebuild



- Clean up your toys!
- Destroy the VM
- Remove from Cobbler
- Remove from IPA

```
require 'rubygems'↵
require 'fog'↵
↵
host_name=ARGV[0]↵
↵
f=Fog::Compute.new(:provider => "Vsphere" )↵
vm=f.get_virtual_machine(host_name)↵
vm["instance_uuid"] = vm["id"]↵
vm["force"] = true↵
f.vm_power_off(vm)↵
f.vm_destroy(vm)↵
↵
command="ssh cobbler sudo cobbler system remove --name #{host_name}"↵
puts "Cobbler:"↵
output=system(command)↵
if output
  puts "Removed #{host_name} from Cobbler"↵
end↵
↵
command="ssh ipa001 ipa host-del #{host_name}"↵
puts "IPA:"↵
output=system(command)↵
↵
command="ssh puppet sudo puppet cert clean #{host_name}.ayisnap.com"↵
puts "Puppet:"↵
output=system(command)↵
```

# Canaries



# Going Live!

---



- On canary servers:
  - `puppet apply -t --environment TEST --noop`
  - `puppet apply -t --environment TEST`
- Merge the pull request :)
- You do have functional checks for the app?
- Your monitoring tools here :)



# Code is available:

---



- Much of this code is available at:
- <https://github.com/matthewbarr/puppet-ci>
- It's out of sync w/ our prod systems, sorry:(
- Redhat / Centos for now...
- I'll try to get it updated soon...

# LOPSA-East Panel

---



- Configuration Management Workflow panel
- list of tools
- <https://github.com/matthewbarr/config-mgmt-utilities>
- (needs cleaning up..)

# Contact Info

---



- **SNAP Interactive, Inc.**  
[www.snap-interactive.com](http://www.snap-interactive.com)
- **Matthew Barr**  
[mbarr@mbarr.net](mailto:mbarr@mbarr.net)
- Twitter & Github:  
@MatthewBarr