

# PUPPET

## A QUICK INTRODUCTION

Dominique Broeglin / @dbroeglin

Cambridge Technology Partners

# DEVOPS: OLD PRACTICES ARE CONSTRAINTS...

- Biannual deployments (because it is so hard...)
- Manual provisioning

# ...AND OLD CONSTRAINTS ARE GONE.

- Upfront capacity planning
- Physical provisioning/access

Source: Luke Kanies

# WALL OF FAME



# PUPPET IN 3 BULLETS

- Version control
- Declarative configuration management
- Centralized automation

# VERSION CONTROL

- Version control is ubiquitous in development:
  - Sharing artifacts
  - Archiving successive versions
  - Documenting change
  - Allowing parallel work
- Lets version our infrastructure!  
*Good idea... but how ?!?*

# DECLARATIVE CONFIGURATION MANAGEMENT

```
class ssh {  
  package { ssh: ensure => installed }  
}
```

+

# DECLARATIVE CONFIGURATION MANAGEMENT

```
class ssh {  
  package { ssh: ensure => installed }  
  file { "/etc/ssh/sshd_config":  
    owner   => root, group => root,  
    source  => "puppet://sshd/sshd_config",  
    require => Package[ssh]  
  }  
}
```

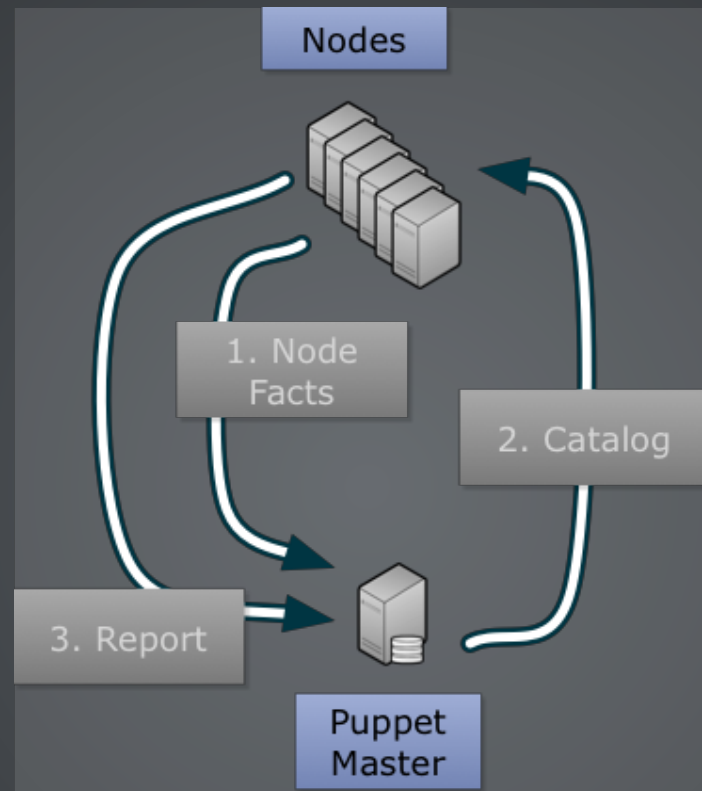


+

# DECLARATIVE CONFIGURATION MANAGEMENT

```
class ssh {  
  package { ssh: ensure => installed }  
  file { "/etc/ssh/sshd_config":  
    name      => "/etc/ssh/sshd_config",  
    owner     => root, group => root,  
    source    => "puppet://sshd/sshd_config",  
    require => Package[ssh]  
  }  
  service { sshd:  
    ensure => running,  
    subscribe => [Package[ssh], File[sshd_config]]  
  }  
}
```

# CENTRALIZED AUTOMATION



# NODE FACTS (1)

```
architecture => amd64
blockdevices => sda
domain => home
fqdn => vagrant-ubuntu-raring-64.home
ipaddress_eth0 => 10.0.2.15
is_virtual => true
kernel => Linux
kernelmajversion => 3.8
kernelrelease => 3.8.0-31-generic
kernelversion => 3.8.0
macaddress_eth0 => 08:00:27:49:4c:f1
...many more...
```

# CATALOG FACTS (2)

1. *Puppet Master* computes a catalog based on:
  - Node facts
  - Manifests
2. *Puppet Master* sends the catalog to the *Node*

# NODE REPORT (3)

1. The *Node* receives the catalog and applies changes as needed
2. The *Node* aggregates a report of what happened and sends it back

# DEMO

## I WANT TO SETUP MY DEVELOPMENT ENVIRONMENT:

- Vagrant VM setup (could be ESX & co.)
- Setup Ubuntu base box
- Setup Puppet
- Then puppet setups:
  - NTP, APT (Ubuntu package management)
  - Oracle Java 7 & Maven
  - Service MIX (OSGi container)
  - My application & configuration
  - ...

# THERE ARE MANY MORE THINGS IN THE PUPPET ECO-SYSTEM

- MCollective : Host orchestration
- Puppet Dashboard
- Puppet enterprise
- Puppet modules
- Librarian
- ...

# REFERENCES

- Reductive Labs Puppet Guides  
<http://docs.reductivelabs.com/guides>
- Infrastructure Deployment with Puppet at CTP ZH