Razor

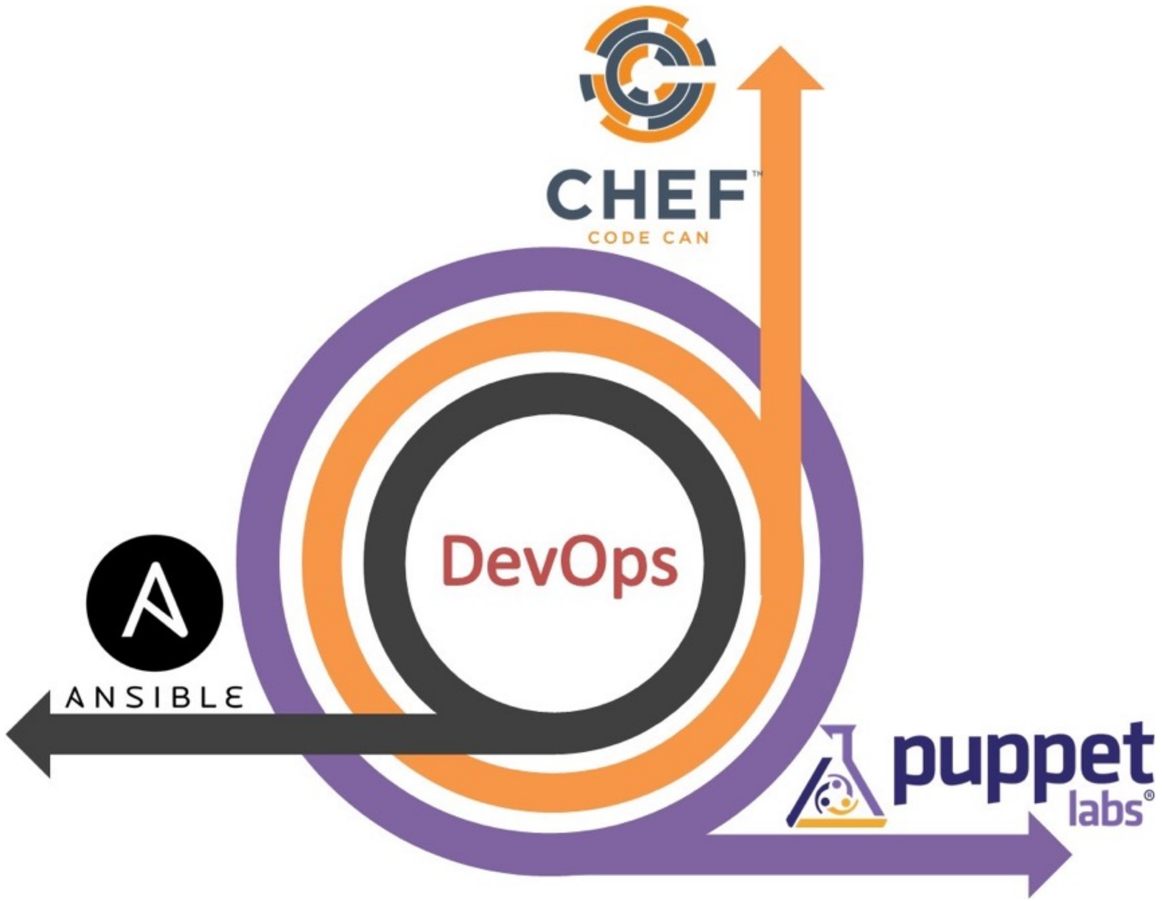
Michal Bryxí
Software Engineer | Puppet Labs
@MichalBryxi

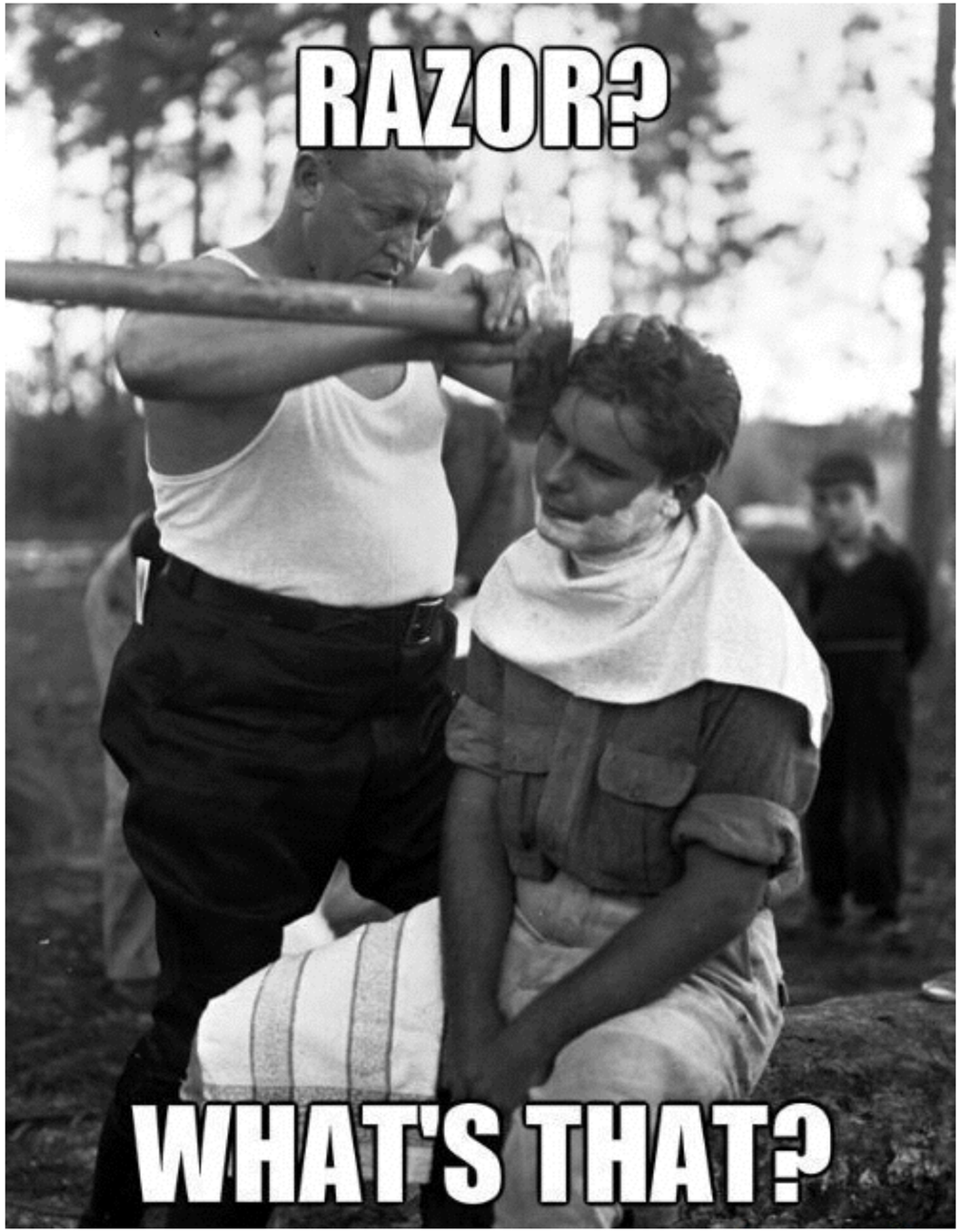
Slides based on work from:

David Lutterkort
Principal Engineer | Puppet Labs
@lutterkort

Presented by puppet labs

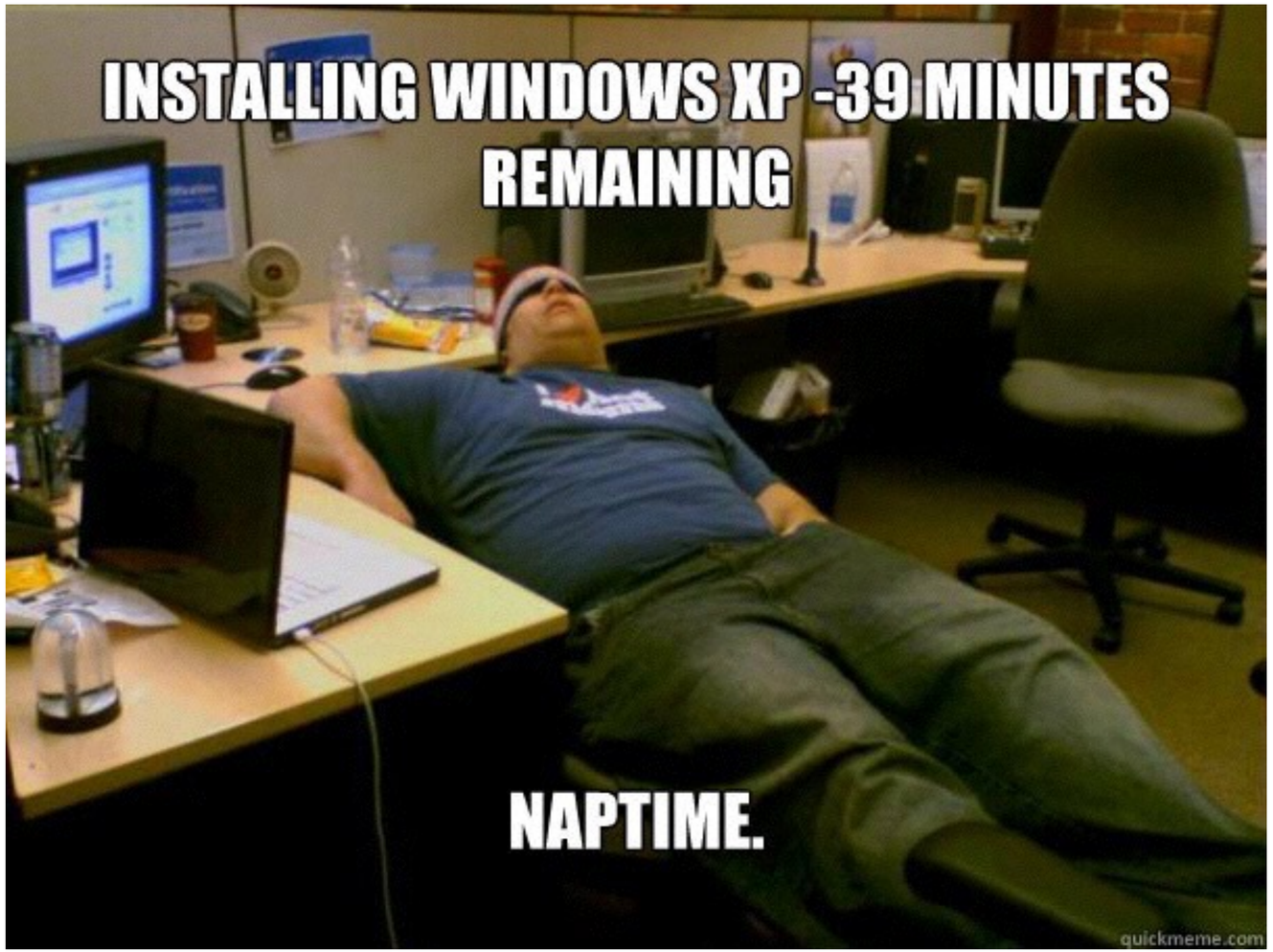
Configuration management? Anyone?




RAZOR?
WHAT'S THAT?

Razor in a nutshell

- Install **X** bare-metal servers
- Let **X** $\in$ <10, ∞)
- Let there be **Y** operating system versions (repos)
- Let there be **Z** fact-to-OS mappings



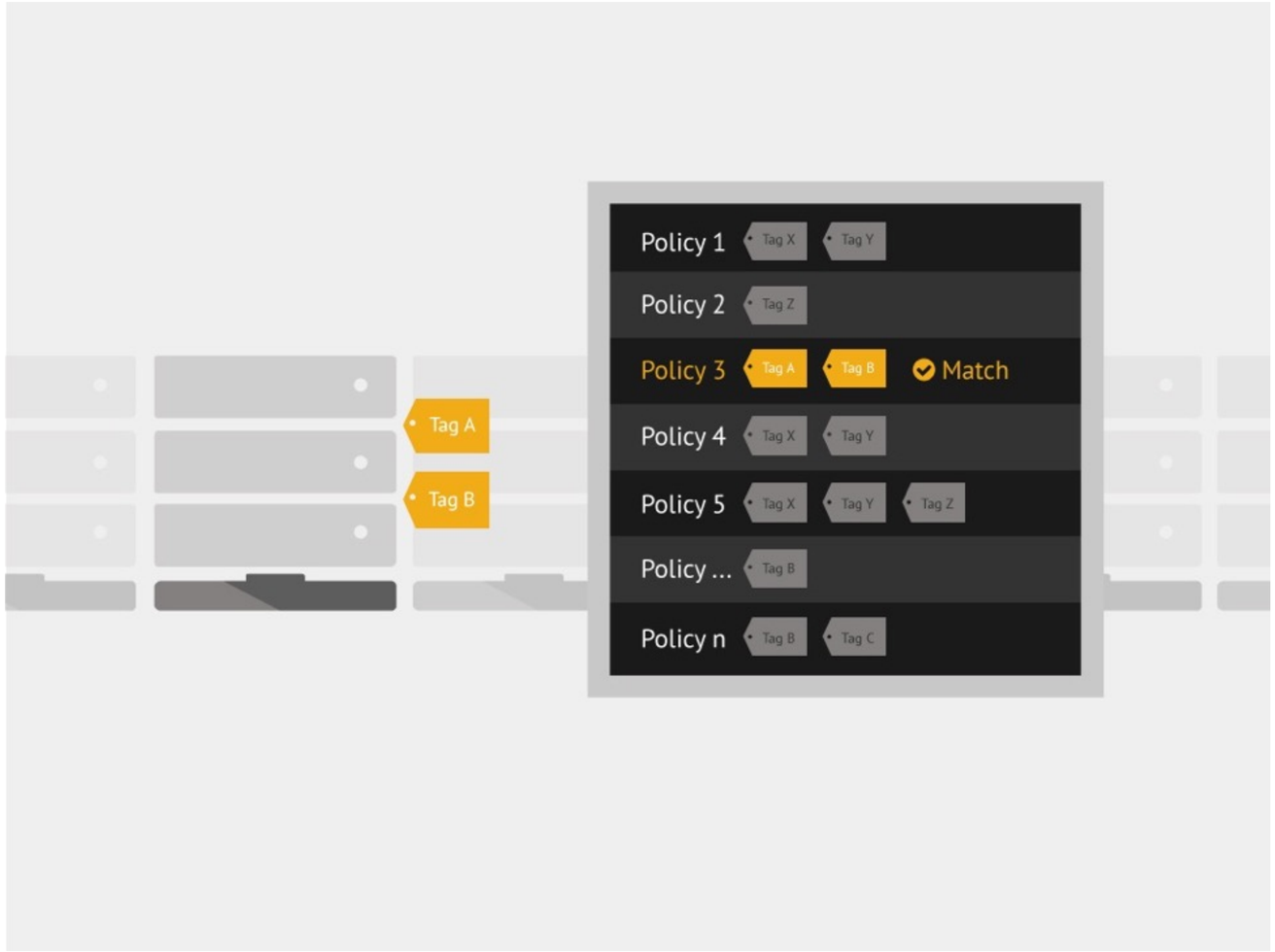
How it works

Razor identifies a new node

How it works

The node is tagged

How it works



The node tags match a Razor policy

How it works

The policy governs how the node is installed.

Repositories
The content to install on a node. To create a repository, either import an install ISO or point at an existing package repository.

Tasks
Installation scripts such as kickstart files, preseed files and additional shell scripts. Predefined tasks are shipped with Razor, and custom tasks can easily be added without additional coding.

Brokers
Post-installation scripts that install a config management agent on the node and enroll the node with the config management system. A predefined broker type for Puppet Enterprise is shipped with Razor.

Tags
Boolean expressions that use node facts and metadata. Tags are used to match nodes and policies.

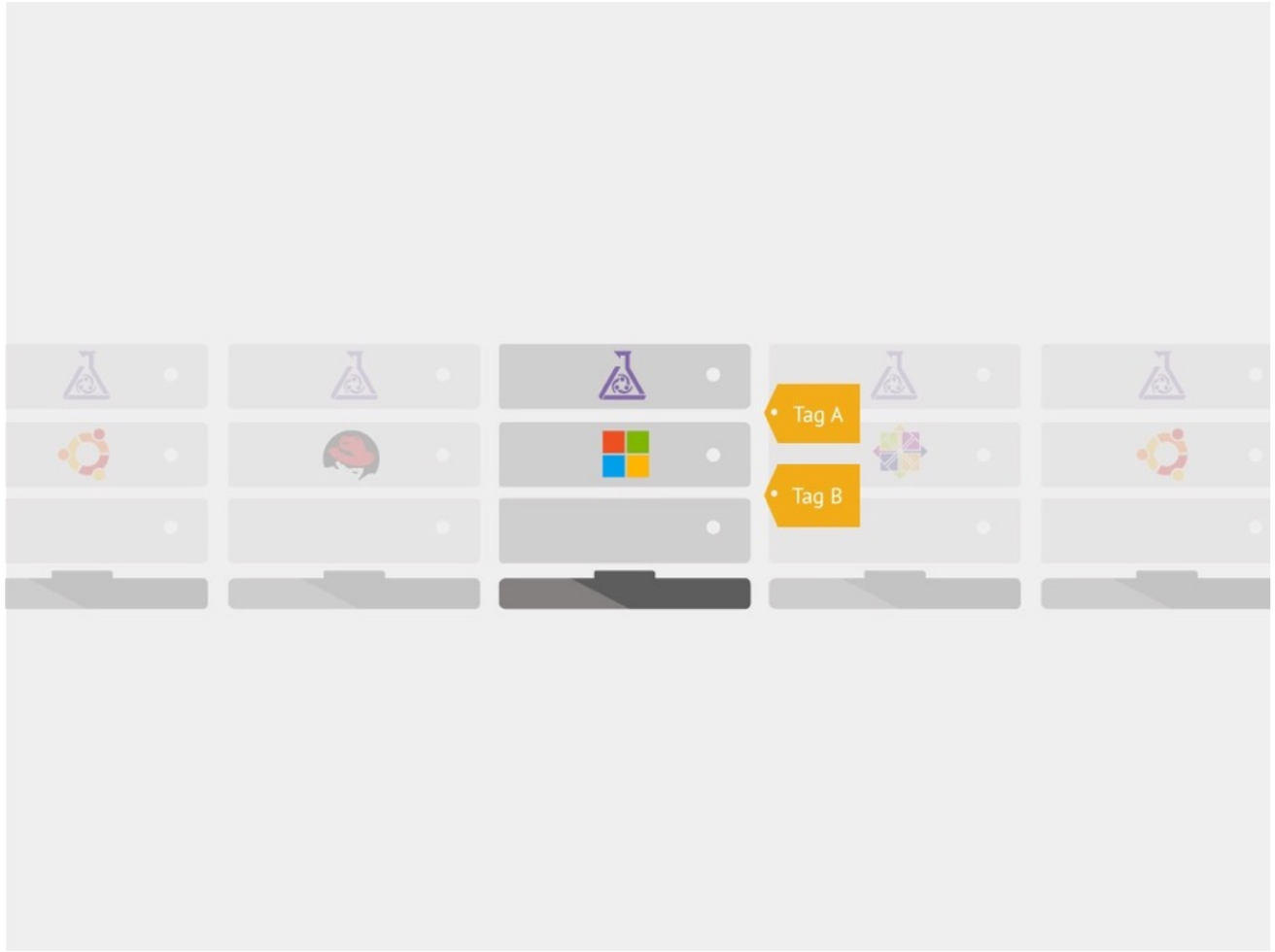
Misc. Data
Makes it possible to generate hostnames for nodes, limit the number of nodes that can match that policy, and initialize a node's metadata.

Policies pull together all the provisioning elements

Moving pieces

Repo	What to install	ISO contents
Task	How to install	Installer scripts
Broker	How to manage	PE agent install
Tag	Where to install	Named match rule
Policy	Combine it all	Ordered table

How it works



The node is provisioned with the designated OS and managed with PE/Puppet/ Chef/Ansible/ 

Demo time!

https://github.com/npwalker/pe-razor-vagrant-stack

Minimal setup

```
> razor create-repo --name trusty \
                    --iso-url http://foo.cz/bar.iso \
                    --task trusty
```

Minimal setup

```
> razor create-broker --name pe \
                      --broker-type puppet-pe \
      --configuration server=puppet-master.example.org
```

Minimal setup

```
> razor create-tag --name test --rule '["=", 1, 1]’

> razor create-tag --name small \
    --rule ‘["<", ["num", ["fact", "memorysize_mb"]], 1024]’

> razor create-tag --name mymac \
    --rule '["in", ["fact", "macaddress"],
                   "00:00:de:ad:be:ef",
                   "00:00:8b:ad:f0:0d"]'
```

Tag matching language

```
["=", arg1, arg2]
["and", arg1, ..., argn]
["fact", arg1]
["tag", arg]
["in", arg1, arg2, ..., argn]
["num", arg1]
[">", arg1, arg2]
...
```

Minimal setup

```
> razor create-policy --name demo \
                      --repo trusty \
                      --hostname '${id}' \
                      --root-password 'secret' \
                      --broker pe
                      --tag test
```

Max count in policy

```
> razor create-policy --name demo \
                      --repo trusty \
                      --hostname '${id}' \
                      --root-password 'secret' \
                      --broker pe
                      --tag test
                      --max-count 20
```

Using node metadata

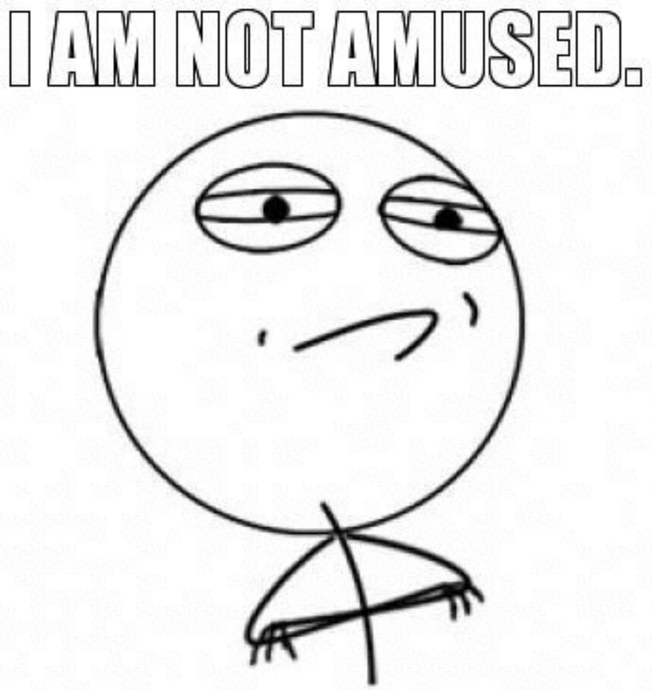
```
> razor update-tag-rule --name test \
                        --force \
          --rule '["=", ["metadata", "os"] "centos"]'
```

Brownfield deployments

- `protect_new_nodes` setting in `config.yaml`
- `register_node` command

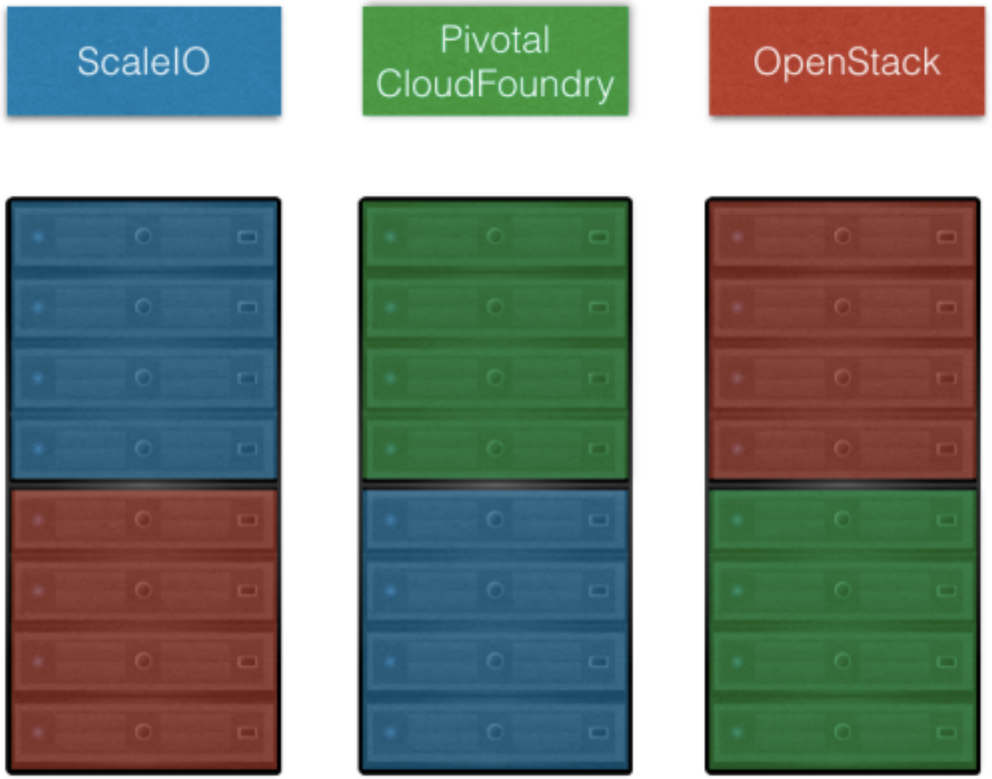
Windows installation

- Additional setup steps:
  1. Build WinPE
  2. Create Windows repo
  3. Samba server



- Supported: Windows 8, Windows 2012R2, 2008R2 WIP

Server locality (@virtualswede)



- server role based on position
- talk to network equipment
- extend MK (micro kernel) with LLDP (Link Layer Discovery Protocol) facts

LLDP facts

```
> razor nodes node1 facts
From http://localhost:8080/api/collections/nodes/node1:
…
                ipaddress_ens2f0: 192.168.1.107
               macaddress_ens2f0: 00:1e:67:4d:c2:06
                  netmask_ens2f0: 255.255.255.0
…
        lldp_neighbor_portid_ens2f0: Ethernet17
      lldp_neighbor_sysname_ens2f0: razor-switch1
 lldp_neighbor_mngaddr_ipv4_ens2f0: 192.168.1.254
```

Microkernel extensions

```
> tree extension

extension/
├── bin/
├── lib/
│   └── ruby/
│       └── facter/
└── facts.d/


> cd extension && zip -r ../mk-extension.zip *
```

Hook scripts

- Called at predefined points in a node's lifecycle
  - `node-registered`
  - `node-bound`
  - `node-uninstalled`
  - `node-deleted`

Questions ?

https://github.com/puppetlabs/razor-server

http://virtualswede.files.wordpress.com/2012/07/puppet-plus-razor-is-good.png

Bonus!

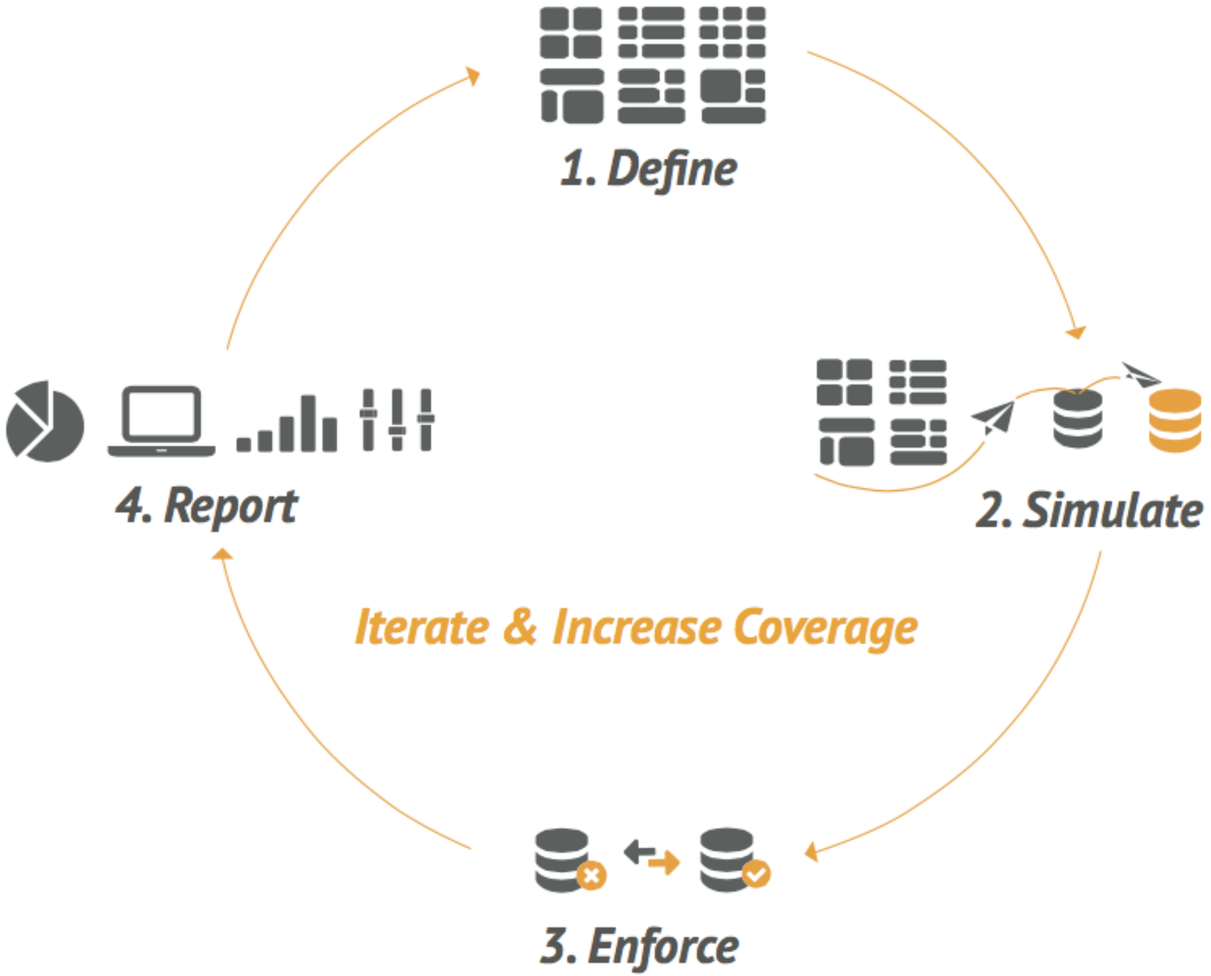
Puppet intro

Presented by puppet labs

What is puppet?

Puppet is a configuration management system that allows you to define the state of **your IT infrastructure**, then automatically **enforces the correct state**. Whether you're managing just a few servers or thousands of physical and virtual machines, Puppet automates tasks that sysadmins often do manually, freeing up time and mental space so sysadmins can work on the projects that deliver greater business value.

Node lifecycle



```
node 'foo.example.com' {
    case $operatingsystem {
      centos, redhat: { $service_name = 'ntpd' }
      debian, ubuntu: { $service_name = 'ntp' }
    }

    package { 'ntp':
      ensure => installed,
    }

    service { 'ntp':
      name       => $service_name,
      ensure     => running,
      enable     => true,
      subscribe => File['ntp.conf'],
    }

    file { 'ntp.conf':
      path    => '/etc/ntp.conf',
      ensure  => file,
      require => Package['ntp'],
      source  => "puppet:///modules/ntp/ntp.conf",
   }
}
```

```
node ‘server1’ {
  …
  @@nagios_host { $hostname:
    ensure          => present,
    check_command   => 'check-host-alive_4',
    use             => 'generic-host',
    contact_groups => 'admins',
  }
  …
}

node ‘nagios’ {
  …
  Nagios_host    <<| |>> { notify => Service['nagios'] }
  …
}
```

Puppet supported

- Red Hat Enterprise Linux (RHEL) 4*, 5, 6, 7
- Windows Server 2003*, 2003 R2*, 2008*, 2008 R2* & 2012*, 7 Ultimate SP1*, 8-8.1* Pro* • CentOS 4*, 5, 6, 7
- Ubuntu LTS 10.04, 12.04, 14.04
- Debian 6, 7
- Scientific Linux 4*, 5, 6
- Oracle Linux 4*, 5, 6, 7
- SLES 10 SP4* & 11 SP1+
- Solaris 10 Update 9+* & 11* • AIX 5.3*, 6.1*, 7.1*
- Mac OS X Mavericks 10.9*

Native Puppet types

augeas
computer
cron
exec
file
filebucket
group
host
interface
k5login
macauthorization
mailalias
maillist
mcx
mount
nagios_command
nagios_contact
nagios_contactgroup
nagios_host
nagios_hostdependency
nagios_hostescalation
nagios_hostextinfo
nagios_hostgroup
nagios_service
nagios_servicedependency
nagios_serviceescalation
nagios_serviceextinfo
nagios_servicegroup
nagios_timeperiod
notify
package
resources
router
schedule
scheduled_task
selboolean
selmodule
service
ssh_authorized_key
sshkey
stage
tidy
user
vlan
yumrepo
zfs
zone
zpool

Forge modules

- AWS EC2
- VMware vSphere
- Openstack
- Eucalyptus
- RightScale
- Zenoss
- Tomcat
- Apache
- nginx
- Cloudera Hadoop
- Red Hat KVM
- Splunk
- Wordpress
- NetApp
- Cisco IOS
- Juniper
- F5
- MySQL
- …

Found 66 modules matching '*apache*'

Relevancy | Latest release | Most Downloads

Filters clear

Operating System
- Any -

Puppet Version
- Any -

Puppet Enterprise Version
- Any -

Supported or Approved
- Any -

Include deleted modules

Apply Filters

puppetlabs/apache

SUPPORTED

Installs, configures, and manages Apache virtual hosts, web services, and modules.

Version 1.4.1 • Apr 28, 2015 • 10,175 downloads

616,896 | 4.5

example42/apache

Puppet module for apache

Version 2.1.9 • Apr 26, 2015 • 402 downloads

495,808 | 4.1

evenup/apache

Manages apache including ajp proxy, thin proxy, and mod_security.

Version 4.0.0 • Nov 11, 2014 • 943 downloads

2,376 | 4.1

theforeman/apache

Puppet Supported

puppetlabs/concat

puppetlabs/catalog_pr...

puppetlabs/apt

puppetlabs/apache

puppetlabs/docker_pla...

Learn more | **View all**

Puppet Approved

herculesteam/augeasp...

zack/r10k

Thank you…