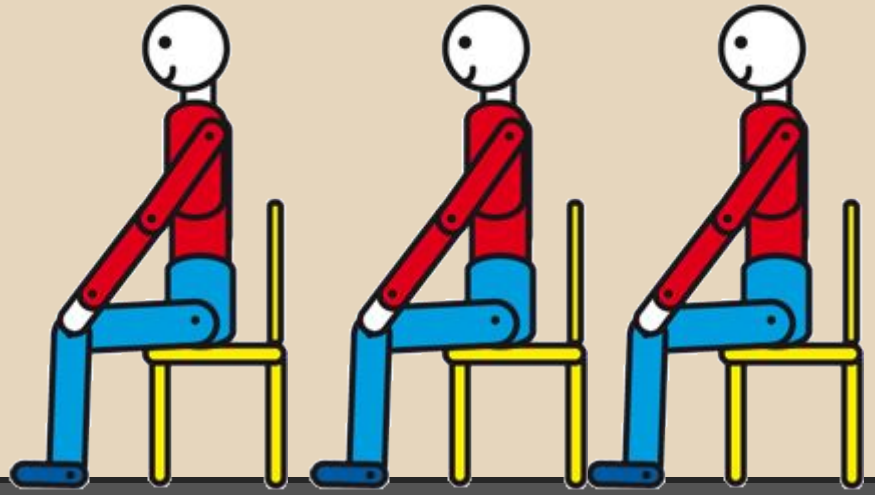


Welcome!

Write Once, Run Everywhere

Beaker & Puppet Labs

me

Alice Nodelman

alice@puppetlabs.com

anode (freenode #puppet-dev)

alicenode (twitter)

anodelman (github)


PROMISE MADE
PROMISE KEPT
imgflip.com

what's beaker?

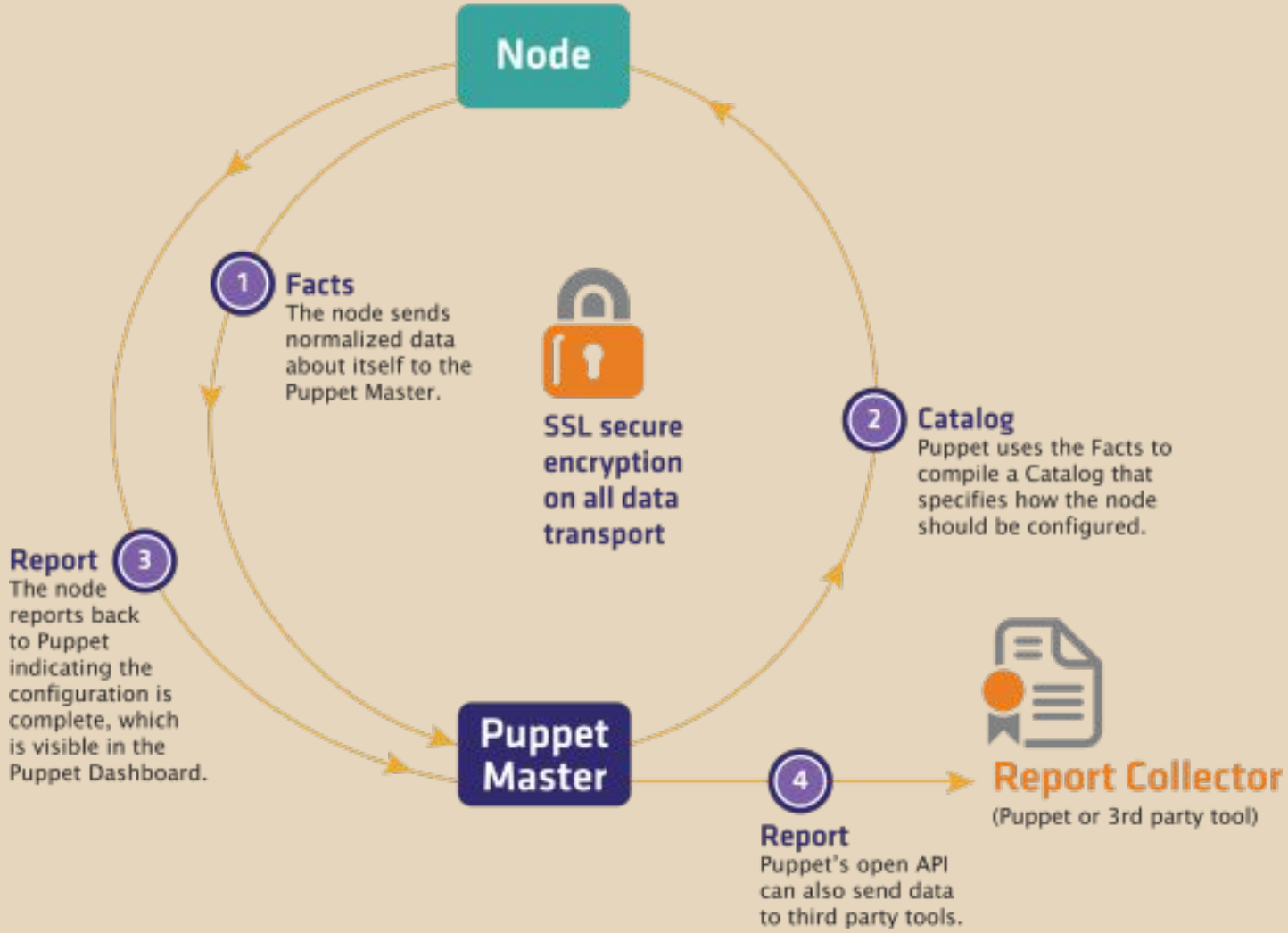
set things on fire so that users don't get burnt

The Problem Space

It’s Problematic.

Puppet in 30 seconds

Puppet is a configuration management system that allows you to define the state of your IT infrastructure, then automatically enforces the correct state.


Node
1
Facts
The node sends normalized data about itself to the Puppet Master.
SSL secure encryption on all data transport
2
Catalog
Puppet uses the Facts to compile a Catalog that specifies how the node should be configured.
3
Report
The node reports back to Puppet indicating the configuration is complete, which is visible in the Puppet Dashboard.
Puppet Master
4
Report
Puppet's open API can also send data to third party tools.
Report Collector
(Puppet or 3rd party tool)

the matrix

- Red Hat Enterprise Linux (RHEL) 4, 5, 6, 7
- Windows Server 2003, 2003 R2, 2008, 2008 R2, 2012, 2012 R2, 2012 R2 core, Vista
- Windows 7, 8-8.1
- Ubuntu 10.04, 12.04, 14.04
- Debian 6, 7
- Solaris 10, 11, 11.2
- SLES 10, 11, 12

more matrix!

- SLES 10, 11, 12
- Scientific Linux 5, 6, 7
- CentOS 4, 5, 6, 7
- Oracle Linux 5, 6, 7
- AIX 5.3, 6.1, 7.1
- Mac OSX 10.9 (Mavericks), 10.10 (Yosemite)
- Arista EOS 4.13.7+
- Cumulus 2.2 (x86_64)

Puppet Modules

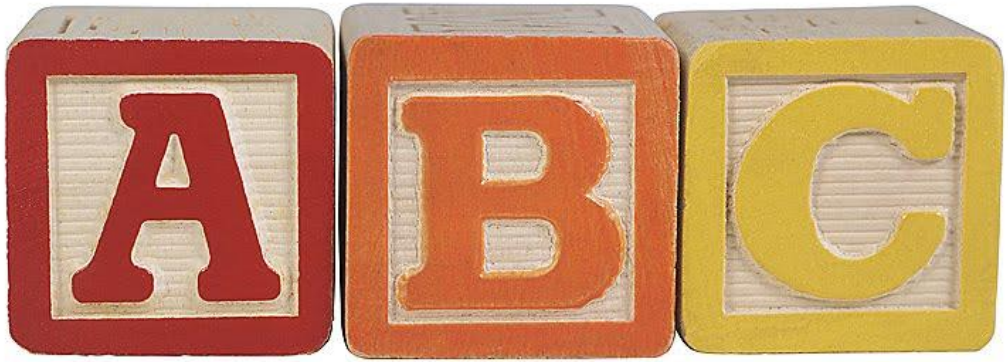
Modules are reusable, sharable units of *Puppet* code.

There are a lot of them!

motivations

1. Make it easy to interact with Puppet
2. Don’t lock in on one means of virtualization
3. Avoid duplication
4. Available to both employees and community members

Your First Beaker Test



```
test_name "say hello to all my hosts!"

hellos = 0
@hosts.each do |host|
  step "i'm going to stay hello to #{host}"
  on host, 'echo hello'
  hellos += 1
end

#make sure that i've been polite!
assert_equal(hellos, hosts.length, \
             "#{hellos} <> #{hosts.length}")
```

```
HOSTS:
  pe-ubuntu-lucid:
    roles:
      - agent
      - dashboard
      - database
      - master
    vmname : pe-ubuntu-lucid
    platform: ubuntu-10.04-i386
    snapshot : clean-w-keys
    hypervisor : fusion
  pe-centos6:
    roles:
      - agent
    vmname : pe-centos6
    platform: el-6-i386
    hypervisor : fusion
    snapshot: clean-w-keys
```

what you see in the console

```
Begin tests/confirm.rb

say hello to all my hosts!

  * i'm going to stay hello to pe-ubuntu-lucid

pe-ubuntu-lucid 11:09:10$ echo hello
Created ssh connection to pe-ubuntu-lucid, user: root, opts: {:config=>false, :paranoid=>false, :timeout=>300, :auth
_methods=>["publickey"], :port=>22, :forward_agent=>true, :keys=>["/Users/anode/.ssh/id_rsa"], :user_known_hosts_fil
e=>"/Users/anode/.ssh/known_hosts", :user=>"root"}
hello

pe-ubuntu-lucid executed in 0.69 seconds

  * i'm going to stay hello to pe-centos6

pe-centos6 11:09:11$ echo hello
Created ssh connection to pe-centos6, user: root, opts: {:config=>false, :paranoid=>false, :timeout=>300, :auth_meth
ods=>["publickey"], :port=>22, :forward_agent=>true, :keys=>["/Users/anode/.ssh/id_rsa"], :user_known_hosts_file=>"/
Users/anode/.ssh/known_hosts", :user=>"root"}
hello
```

console summary

```
Test Suite: tests @ 2015-01-05 11:09:10 -0800

- Host Configuration Summary -


        - Test Case Summary for suite 'tests' -
 Total Suite Time: 0.83 seconds
Average Test Time: 0.83 seconds
        Attempted: 1
           Passed: 1
           Failed: 0
          Errored: 0
          Skipped: 0
          Pending: 0
            Total: 1
```

Beaker Puppet Labs Automated Acceptance Testing System

Elapsed Time: 3278.00423 sec

Total: 268 **Failed: 0** **Skipped: 48** **Pendi**

pre_suite — Elapsed Time: 86.39030 sec

Total: 4 Failed: 0 Skipped: 0 Pending: 0

tests — Elapsed Time: 3191.61392 sec

Total: 264 Failed: 0 Skipped: 48 Pending: 1

Properties

agent_disable_lockfile.rb

Path: tests/agent/agent_disable_lockfile.rb Elapsed Time: 35.45660 sec

what beaker can’t do

nobody’s perfect

- GUI
- threading
- orchestration

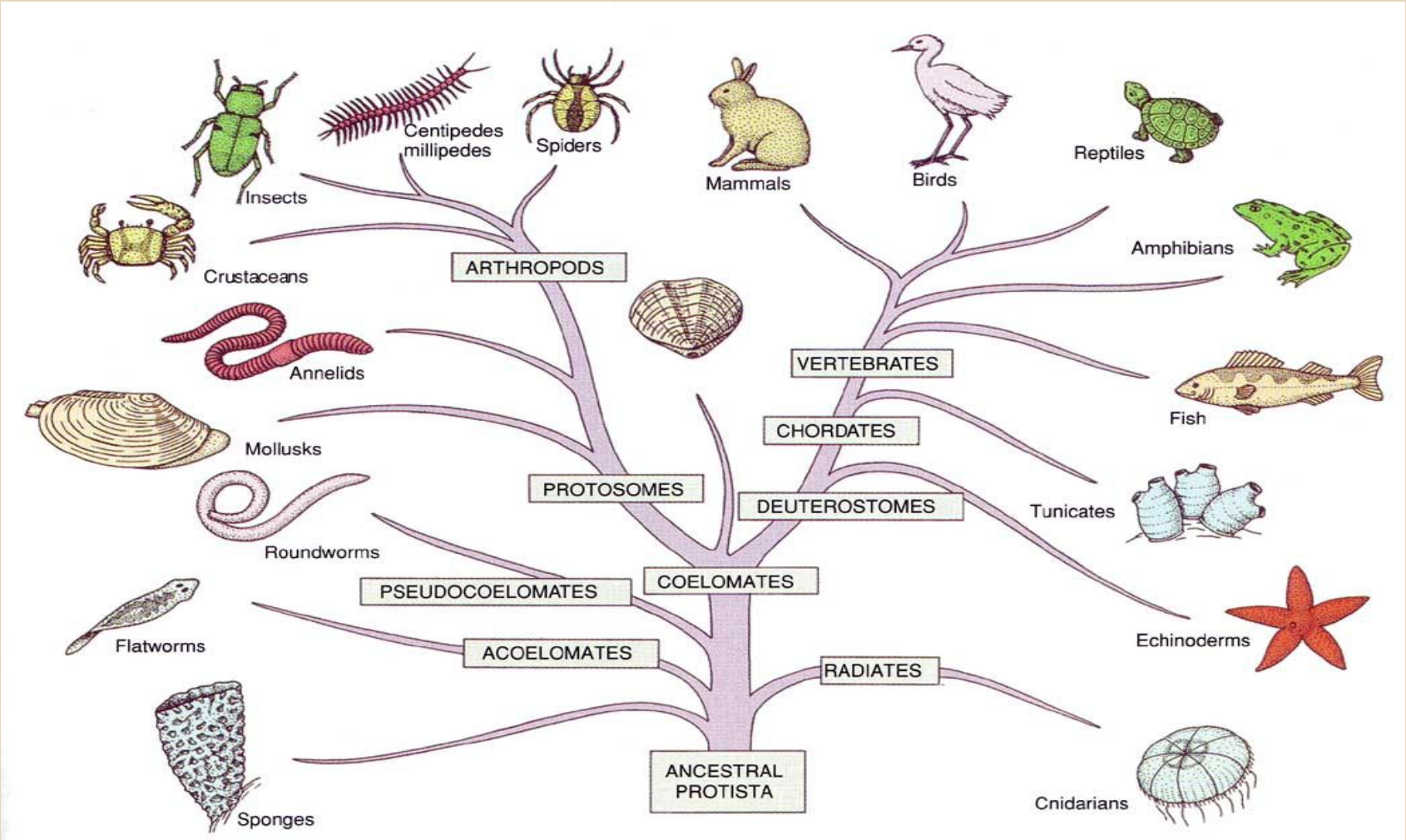
what beaker can do

write once, run everywhere

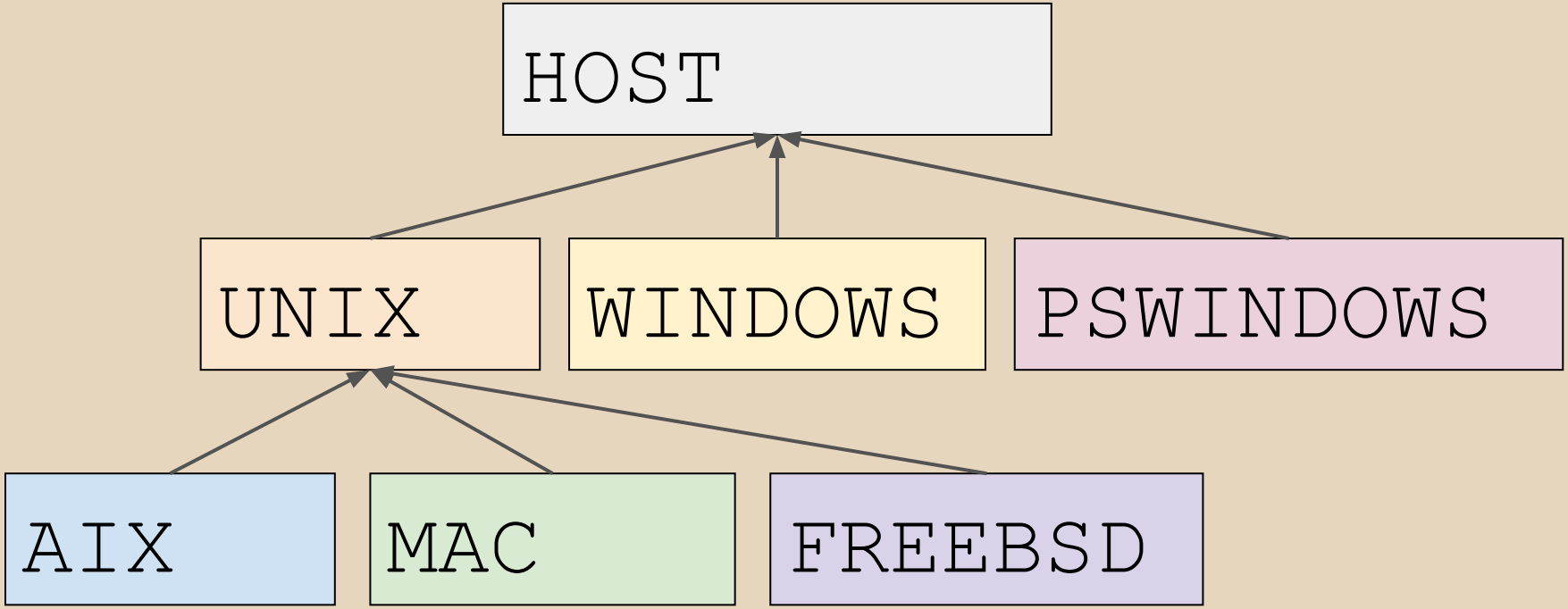
- multi-node
- multi-platform
- multi-cloud

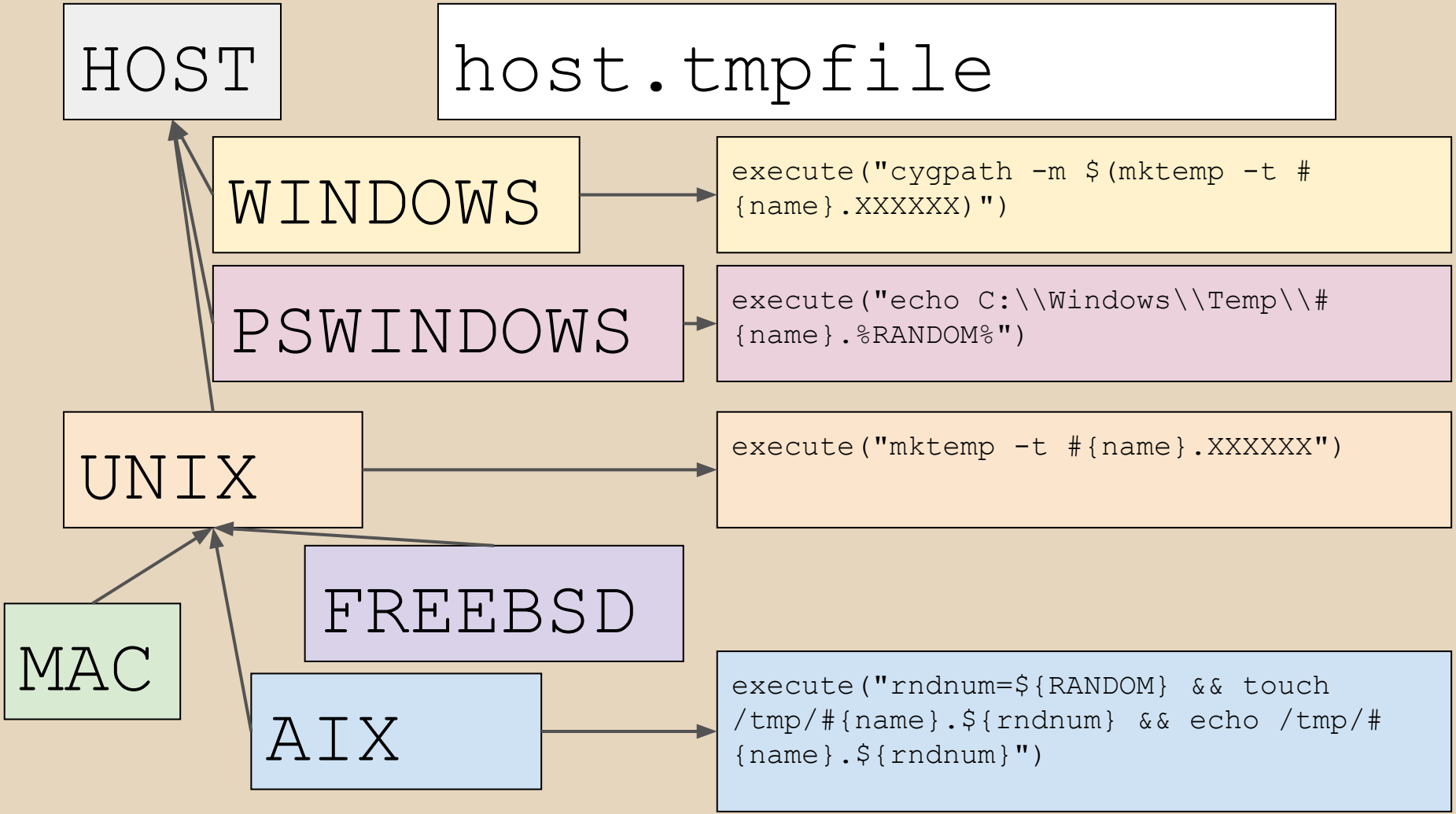
dive in!

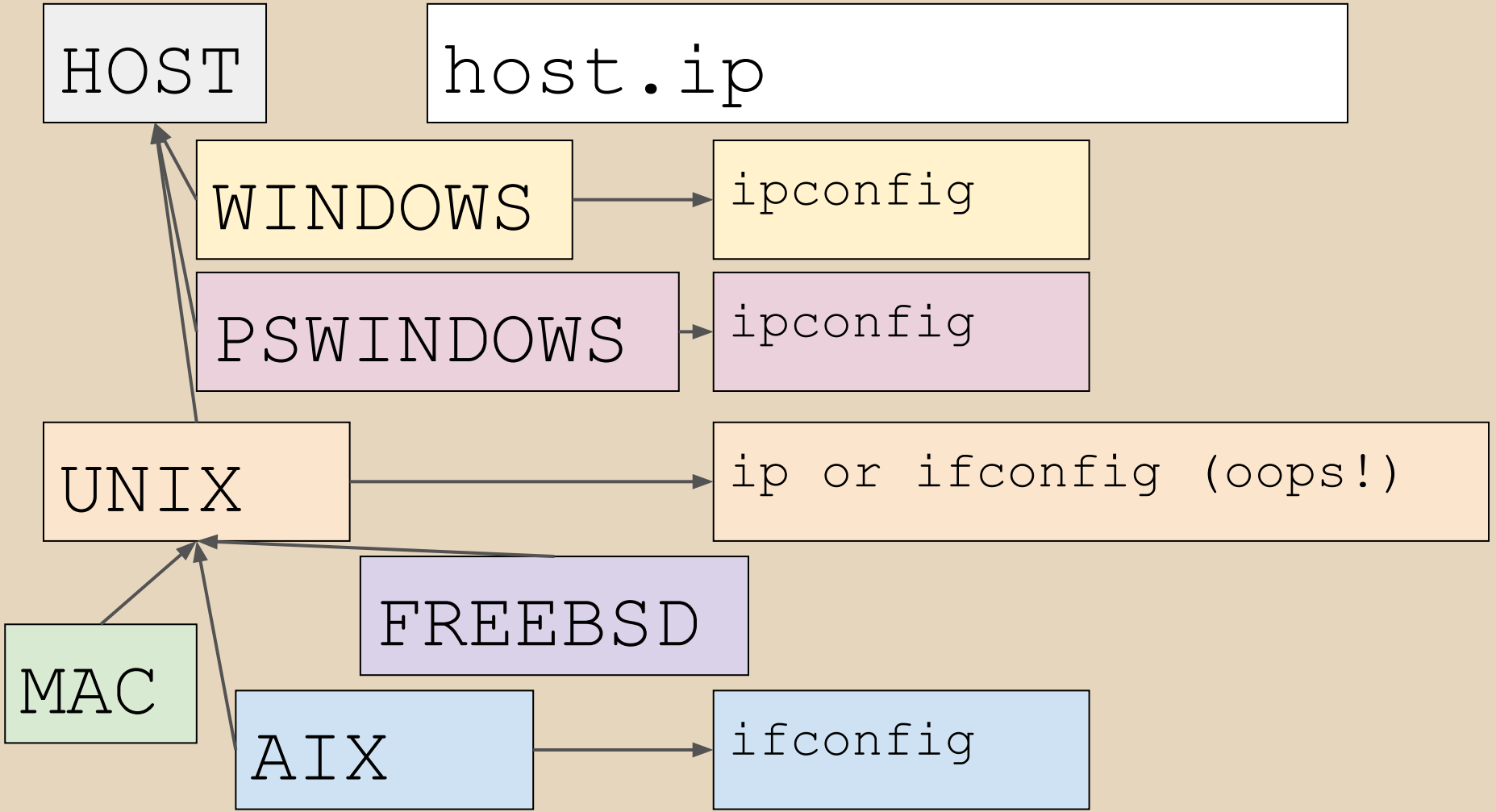
```
test_name "PNSQC Example Test"

step "Cycling through all test hosts"
hosts.each do |host|
  step "creating tmp file on #{host}"
  my_tmp = host.tmpfile('my_tmp')
  if not host.file_exist?(my_tmp)
    put "Failed to create temporary file on #{host}"
  end
  step "discovering ip on #{host}"
  my_ip = host.ip
  puts "#{host} (#{host[:platform]}) has ip #{my_ip}"
end
step "Completed"
```

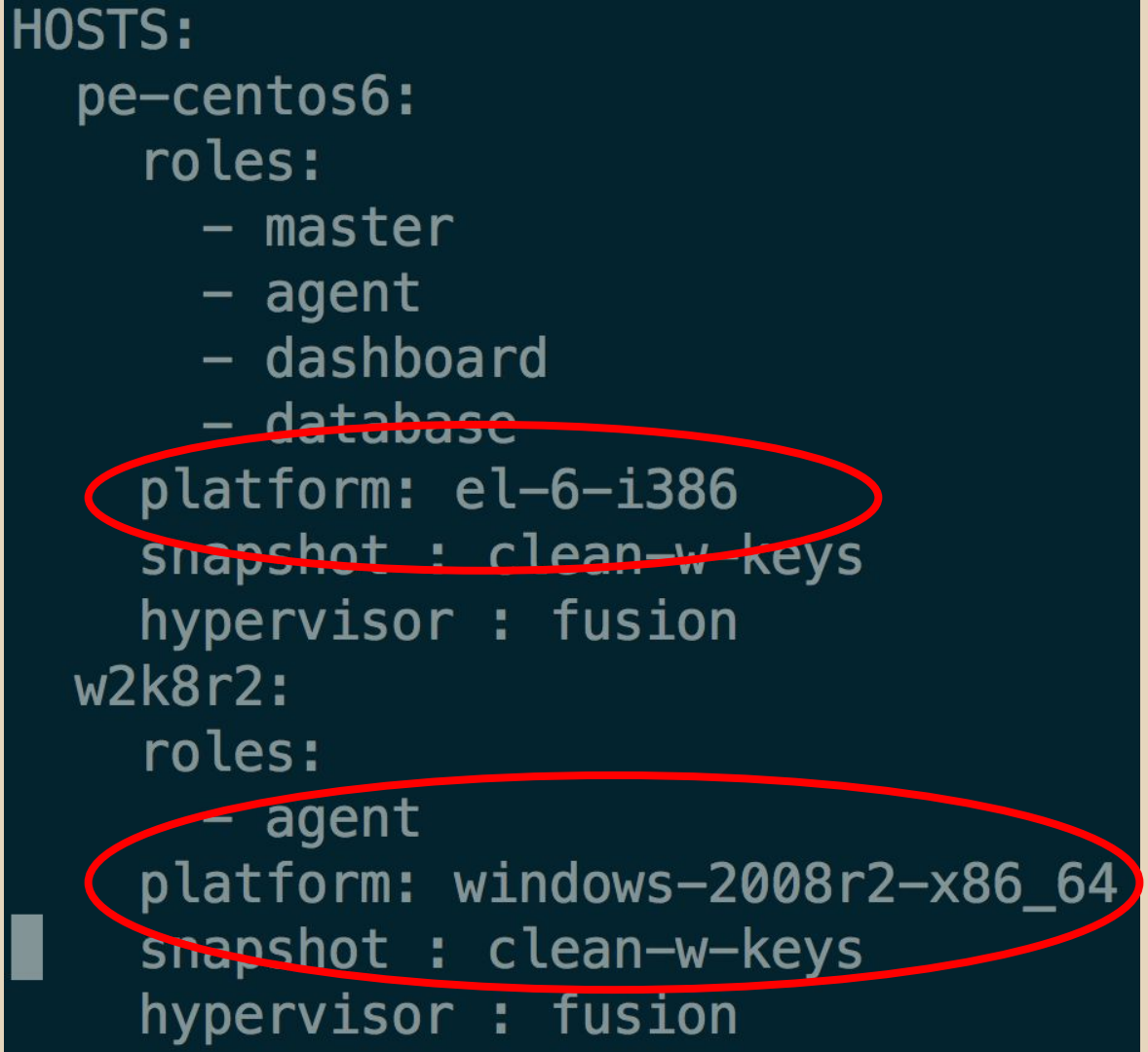

Insects
Centipedes
millipedes
Spiders
Mammals
Birds
Reptiles
Crustaceans
ARTHROPODS
Amphibians
Annelids
VERTEBRATES
Fish
Mollusks
CHORDATES
PROTOSOMES
DEUTEROSTOMES
Tunicates
Roundworms
COELOMATES
PSEUDOCOELOMATES
Flatworms
Echinoderms
ACOELOMATES
RADIATES
Sponges
ANCESTRAL
PROTISTA
Cnidarians

basics of host object inheritance




HOST
host.tmpfile
WINDOWS
execute("cygpath -m $(mktemp -t #{name}.XXXXXX)")
PSWINDOWS
execute("echo C:\\Windows\\Temp\\#{name}.%RANDOM%")
UNIX
execute("mktemp -t #{name}.XXXXXX")
FREEBSD
MAC
AIX
execute("rndnum=${RANDOM} && touch /tmp/#{name}.${rndnum} && echo /tmp/#{name}.${rndnum}")


HOST
host.ip
WINDOWS
ipconfig
PSWINDOWS
ipconfig
UNIX
ip or ifconfig (oops!)
FREEBSD
MAC
AIX
ifconfig


HOSTS:
pe-centos6:
roles:
- master
- agent
- dashboard
- database
platform: el-6-i386
snapshot : clean-w-keys
hypervisor : fusion
w2k8r2:
roles:
- agent
platform: windows-2008r2-x86_64
snapshot : clean-w-keys
hypervisor : fusion

platform string to object

osx centos fedora debian oracle redhat scientific sles ubuntu windows solaris aix el eos cumulus

```ruby
def self.create name, options
  case options['HOSTS'][name]['platform']
  when /windows/
    cygwin = options['HOSTS'][name]['is_cygwin']
    if cygwin.nil? or cygwin == true
      Windows::Host.new name, options
    else
      PSWindows::Host.new name, options
    end
  when /aix/
    Aix::Host.new name, options
  when /osx/
    Mac::Host.new name, options
  else
    Unix::Host.new name, options
  end
end
```

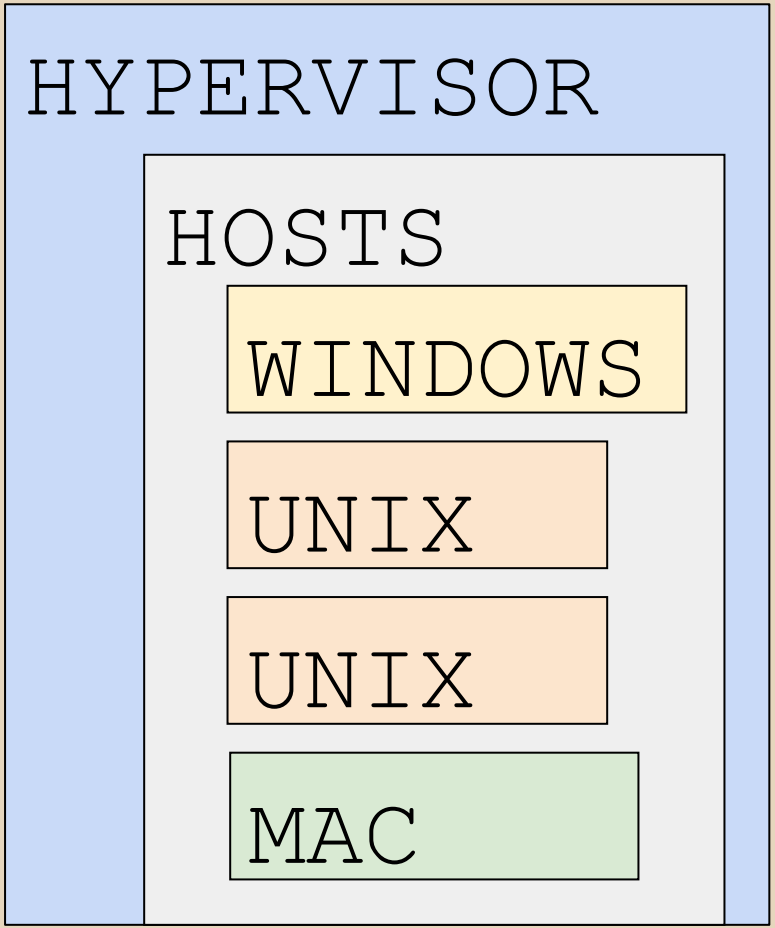
```
test_name "PNSQC Example Test"

step "Cycling through all test hosts"
hosts.each do |host|
  step "creating tmp file on #{host}"
  my_tmp = host.tmpfile('my_tmp')
  if not host.file_exist?(my_tmp)
    put "Failed to create temporary file on #{host}"
  end
  step "discovering ip on #{host}"
  my_ip = host.ip
  puts "#{host} (#{host[:platform]}) has ip #{my_ip}"
end
step "Completed"
```

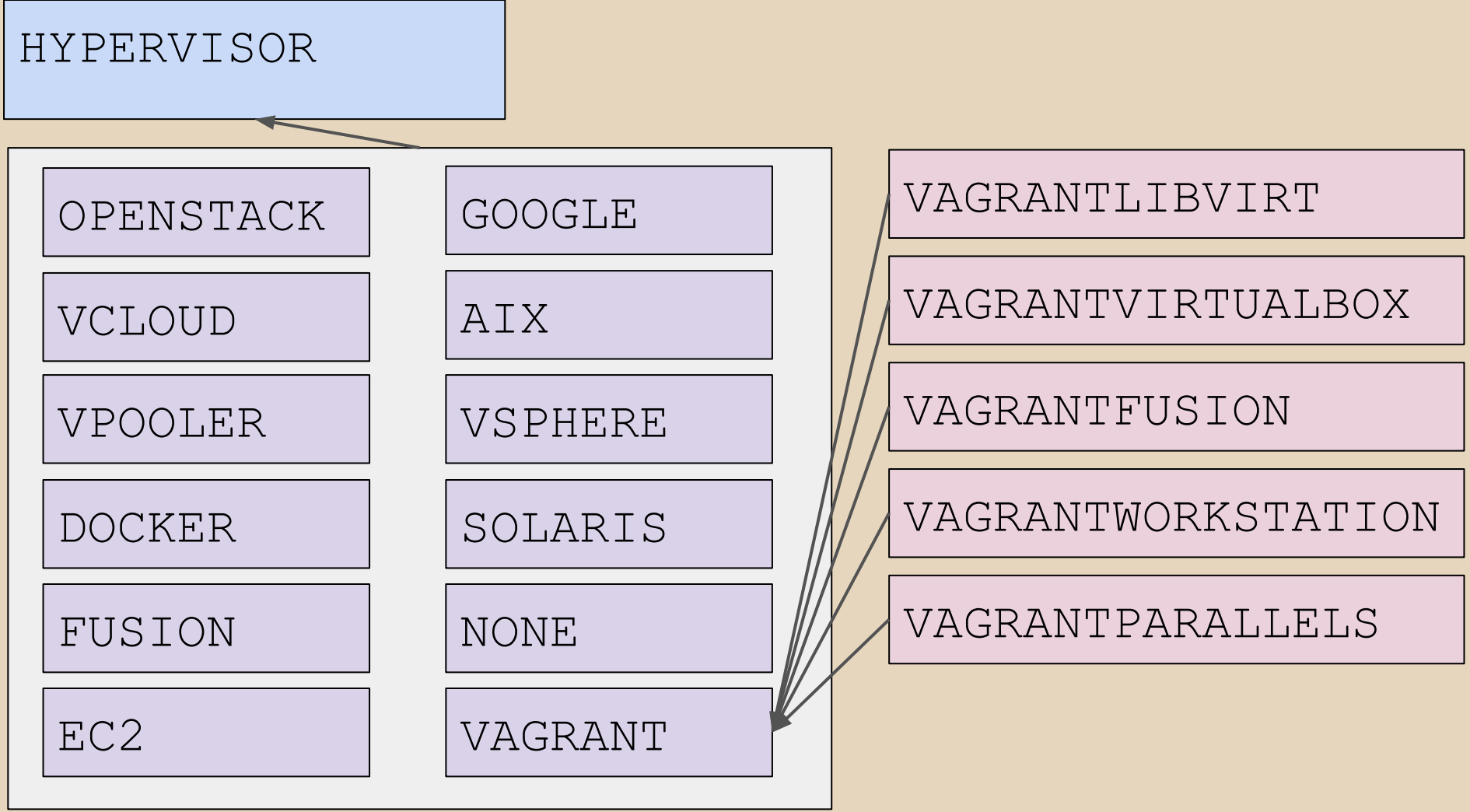
```
  * discovering ip on w2k3r2

w2k3r2 13:36:42$ ipconfig | grep -i 'IP Address' | cut -d: -f2 | head -1
 172.16.146.129

w2k3r2 executed in 0.16 seconds
w2k3r2 (windows-2003r2-x86_64) has ip 172.16.146.129

  * creating tmp file on pe-ubuntu-lucid

pe-ubuntu-lucid 13:36:42$ mktemp -t my_tmp.XXXXXX
/tmp/my_tmp.955RTW

pe-ubuntu-lucid executed in 0.01 seconds

pe-ubuntu-lucid 13:36:42$ test -e /tmp/my_tmp.955RTW

pe-ubuntu-lucid executed in 0.00 seconds

  * discovering ip on pe-ubuntu-lucid

pe-ubuntu-lucid 13:36:42$ ip a|awk '/global/{print$2}' | cut -d/ -f1 | head -1
172.16.146.130
```

so where are the hosts?

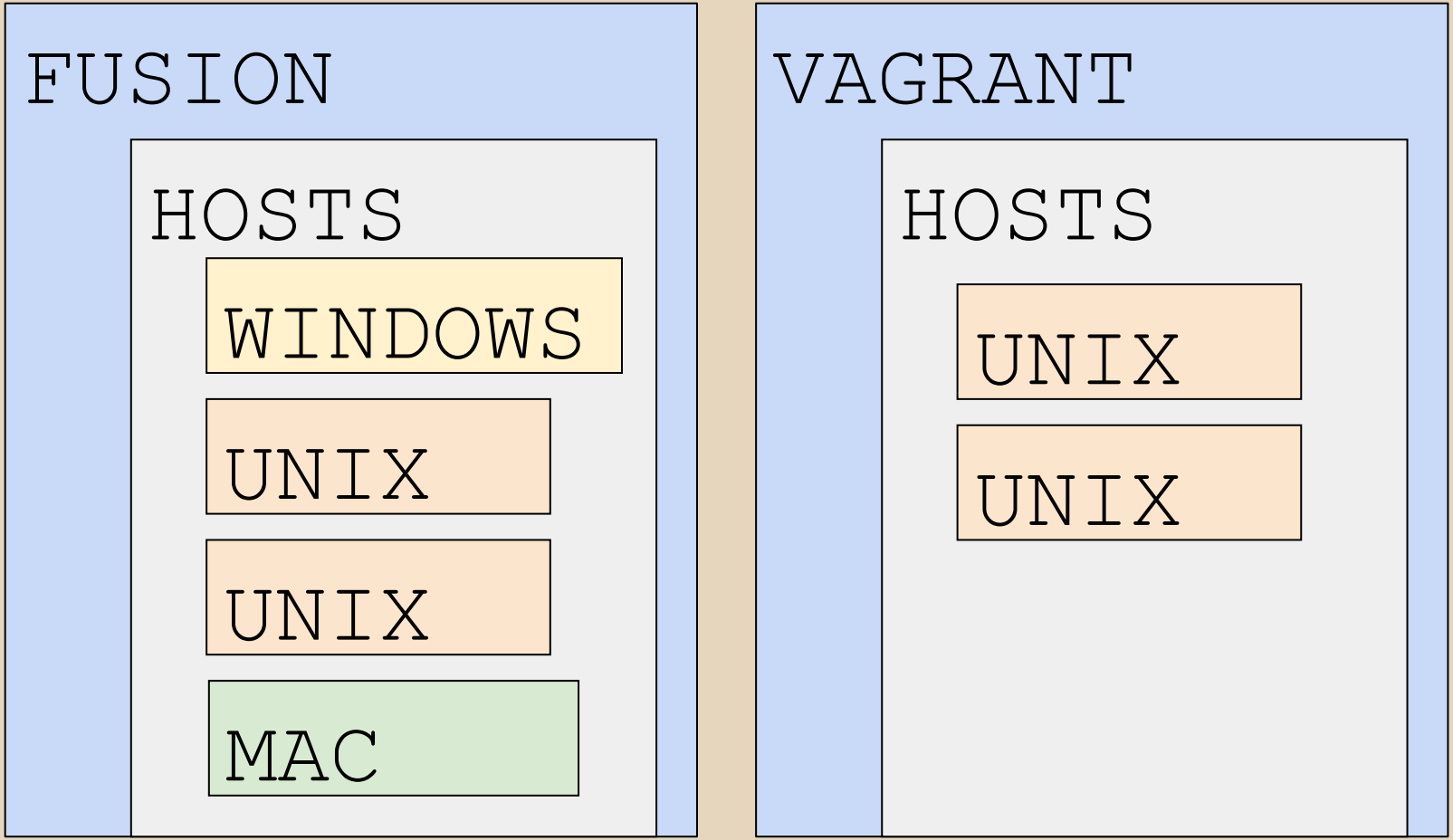
the hypervisor object

“A hypervisor or virtual machine monitor (VMM) is a piece of computer software, firmware or hardware that creates and runs virtual machines.”


HYPERVISOR
HOSTS
WINDOWS
UNIX
UNIX
MAC

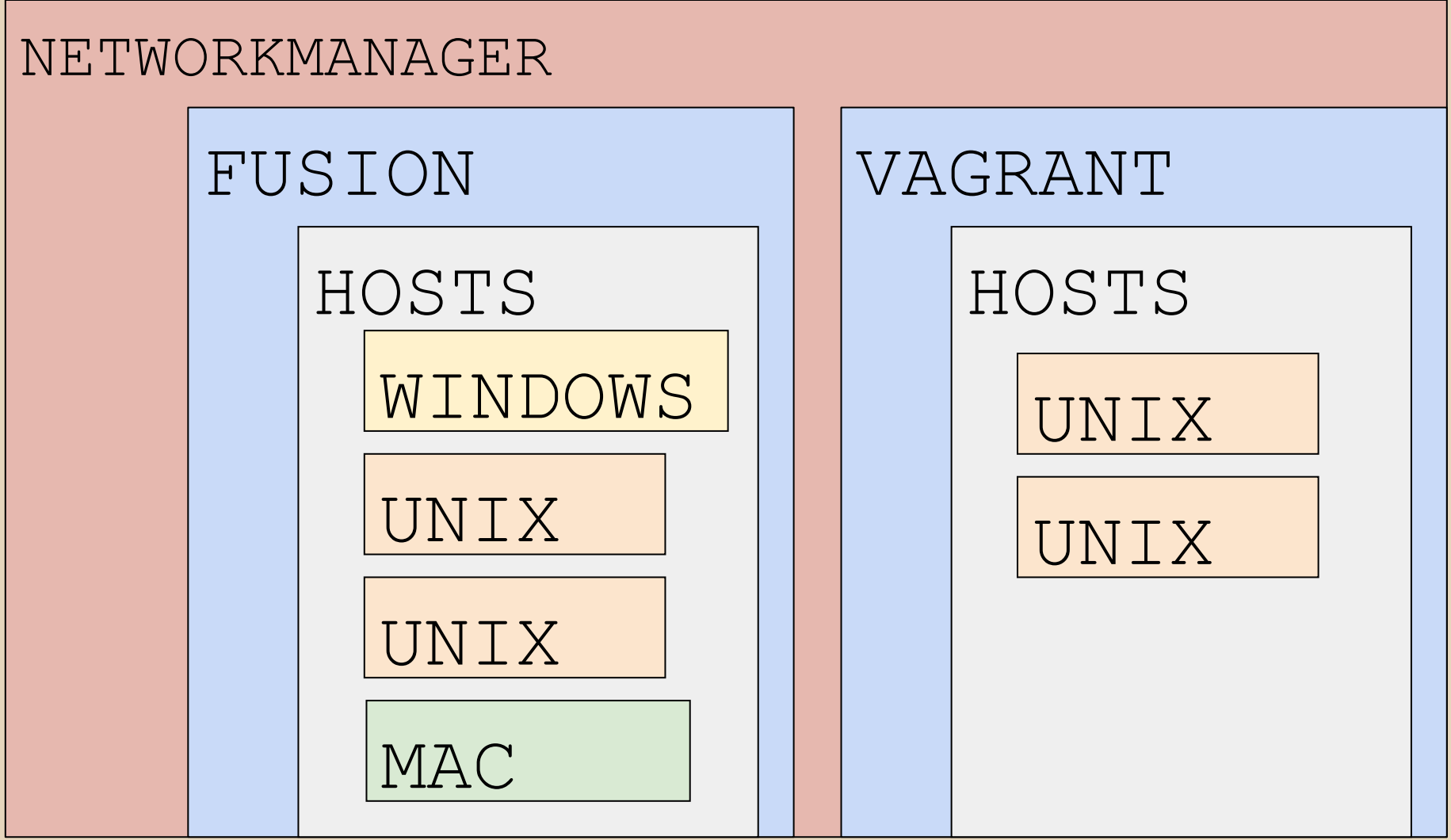

HYPERVISOR
OPENSTACK
GOOGLE
VCLOUD
AIX
VPOOLER
VSPHERE
DOCKER
SOLARIS
FUSION
NONE
EC2
VAGRANT
VAGRANTLIBVIRT
VAGRANTVIRTUALBOX
VAGRANTFUSION
VAGRANTWORKSTATION
VAGRANTPARALLELS

```ruby
module Beaker
  class New < Beaker::Hypervisor

    def initialize(new_hosts, options)
      @options = options
      @logger = options[:logger]
      @hosts = new_hosts
    end

    def provision
      # create your hosts here, once this is complete
      # it assumed that you can log in by hostname or vmname or ip
    end

    # OPTIONAL - override if you have special actions here
    #def configure
    #end

    # OPTIONAL - override if you have special actions here
    #def validate
    #end

    def cleanup
      # cleanup hosts here
    end

  end
end
```


FUSION
HOSTS
WINDOWS
UNIX
UNIX
MAC
VAGRANT
HOSTS
UNIX
UNIX

NetworkManager object

manages your hypervisor objects


NETWORKMANAGER
FUSION
HOSTS
WINDOWS
UNIX
UNIX
MAC
VAGRANT
HOSTS
UNIX
UNIX

hypervisor string to object

it’s in the hosts file!

ec2

```
HOSTS:
  centos-5-64-1:
    roles:
      - master
      - dashboard
      - database
      - agent
    vmname: centos-5-x86-64-west
    platform: el-5-x86_64
    hypervisor: ec2
    amisize: c1.medium
    snapshot: pe
```

GCE

```
HOSTS:
  centos-6-master:
    roles:
      - master
      - agent
      - database
      - dashboard
    image: centos-6-x86_64
    platform: el-6-x86_64
    hypervisor: google
```

vagrant

```
HOSTS:
  centos-65-x64:
    roles:
      - master
    platform: el-6-x86_64
    box: puppetlabs/centos-6.5-64-nocm
    box_url: https://vagrantcloud.com/puppetlabs/boxes/centos-6.5-64-nocm
    hypervisor: vagrant
```

fusion

```
HOSTS:
  pe-centos6:
    roles:
      - master
      - agent
      - dashboard
      - database
      - myrole
    platform: el-6-x86_64
    snapshot : clean-w-keys
    hypervisor : fusion
```

example execution

```
$ beaker --hosts pnsqc.yml --tests tests/pnsqc.rb
```

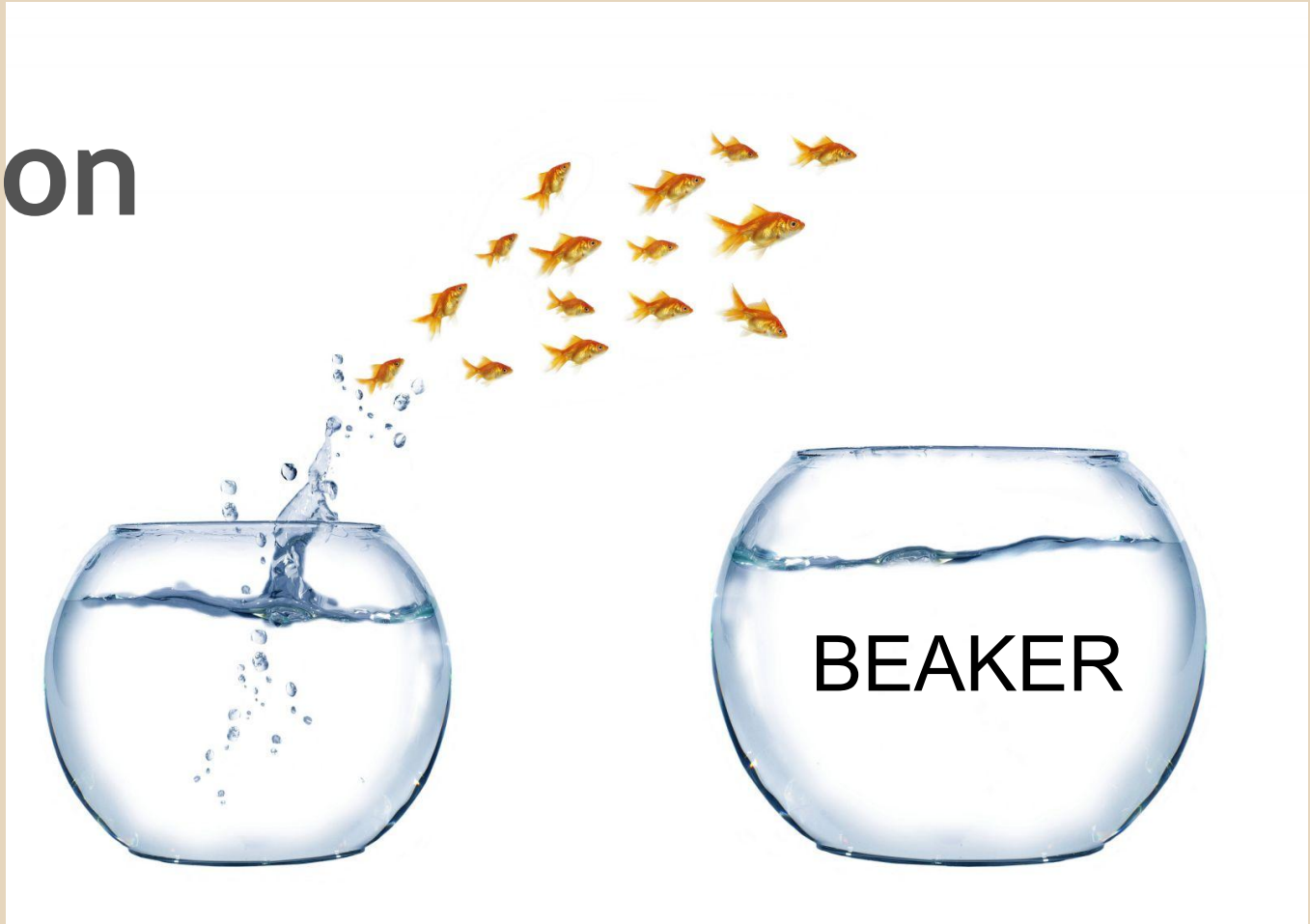
```
test_name "PNSQC Example Test"

step "Cycling through all test hosts"
hosts.each do |host|
  step "creating tmp file on #{host}"
  my_tmp = host.tmpfile('my_tmp')
  if not host.file_exist?(my_tmp)
    put "Failed to create temporary file on #{host}"
  end
  step "discovering ip on #{host}"
  my_ip = host.ip
  puts "#{host} (#{host[:platform]}) has ip #{my_ip}"
end
step "Completed"
```

write once, run everywhere

tests have **no** reference to platform

tests have **no** reference to virtualization

Adoption



FOSS Puppet (master branch)	278
Puppet Enterprise (4.0)	441
PuppetDB	28
Hiera	6
Facter	23
25 Puppet Labs Supported Modules	351
56 Puppet Approved Community Modules	31
Puppet Labs QA team internal test suites	917
	2075

Beaker Code

Beaker Wiki

Beaker DSL

Beaker Google Group

Beaker JIRA Bugs

Pretty Beaker Junit Output