

Hands-On: Roles & Profiles

Hunter Haugen

Forge Module Engineer | Puppet Labs
@Hunnur



presented by  puppet
labs

Agenda

- The Problem
- The Solution
- Roles
- Profiles
- Hands-On

Question:

Is anyone here already using Roles & Profiles?

The Problem

These are not the nodes you're looking for.



presented by  puppet
labs

What is classification?

Classification was assigning classes to nodes.

(This used to be the case.)

What are these nodes?

```
node 'hermes.example.com' {  
  include network  
  include users  
  include exim  
  include dovecot  
  include roundcube  
}
```

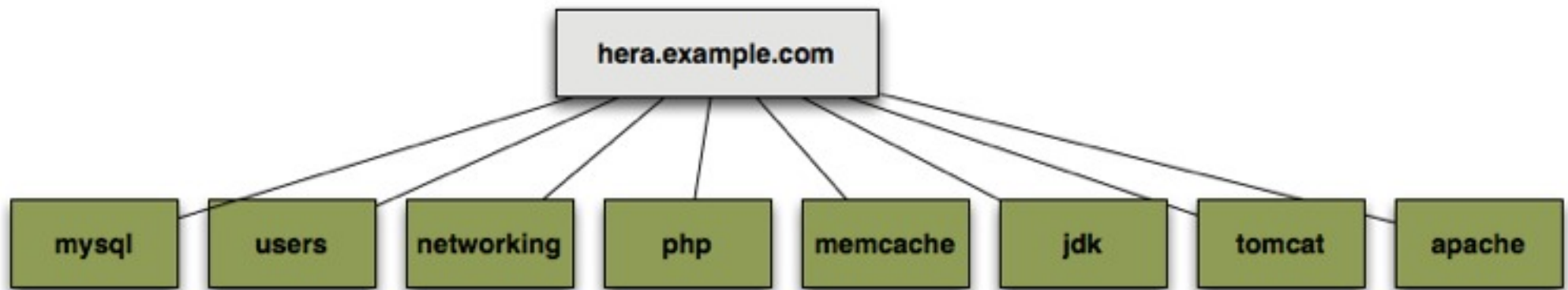
```
node 'ares.example.com' {  
  include network  
  include users  
  include mysql  
}
```

What are these nodes?

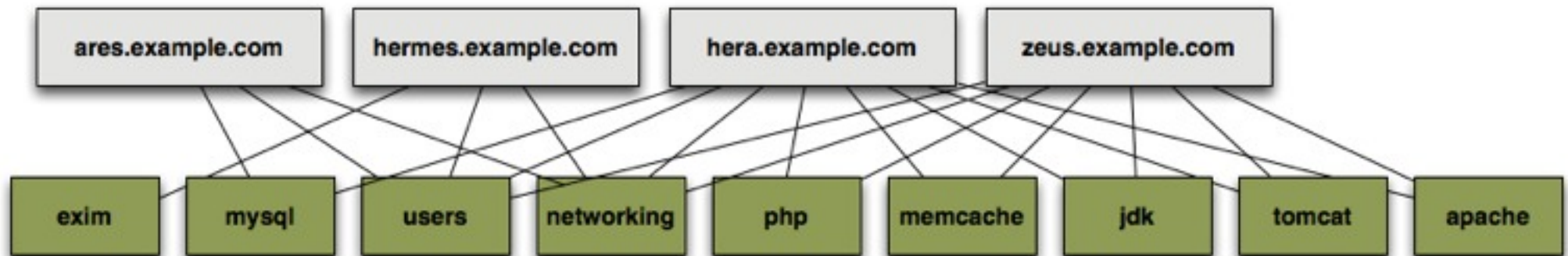
```
node 'hera.example.com' {  
  include network  
  include users  
  include apache  
  include mysql  
  include memcache  
  include jdk  
  include tomcat  
  class { 'php':  
    loglevel => 'debug',  
  }  
}
```

```
node 'zeus.example.com' {  
  include network  
  include users  
  include apache  
  include php  
  include memcache  
  include jdk  
  include tomcat  
}
```

One node has many classes



Many nodes make a tangled mess



This will happen

Boss: “Can you make another development silo for the devs?”

Just like production, but different.

The Solution

If you aren't part of the solution then you are the precipitate.



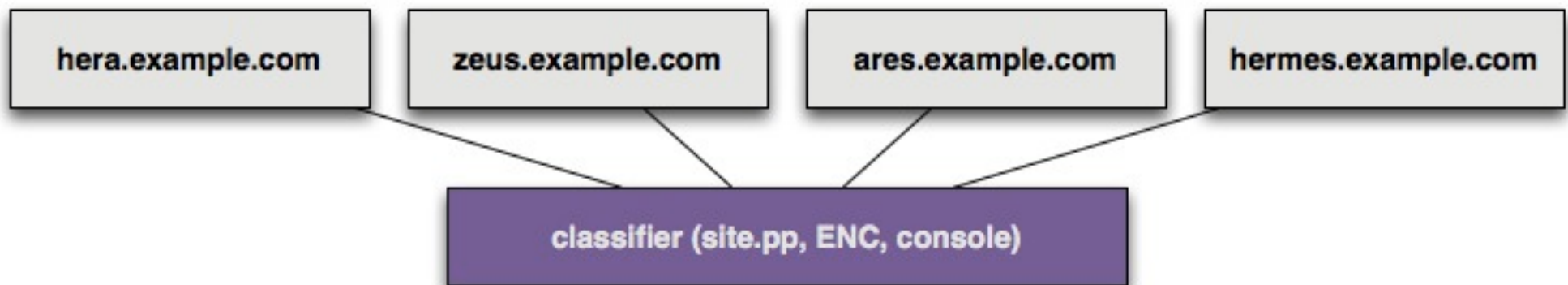
What is classification?

Classification is assigning roles to nodes.

History Lesson... The Past

- Nodes & Classes
- Node-scope Variables
- Node Inheritance
- Class Parameters
- Inheritance == Bad

History Lesson... The Transition



History Lesson... The Present

- External Node Classifiers
- Roles & Profiles Pattern
- Hieradata
- Class Parameter Bindings
- PuppetDB

Craig Dunn's Solution

Roles & Profiles Pattern

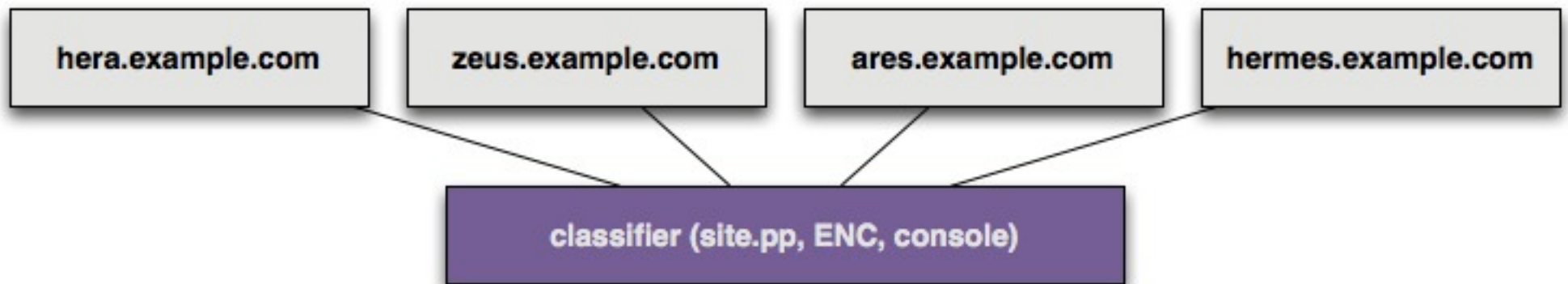


Roles

One role to rule them all, and with profiles bind them.



Roles + Classification



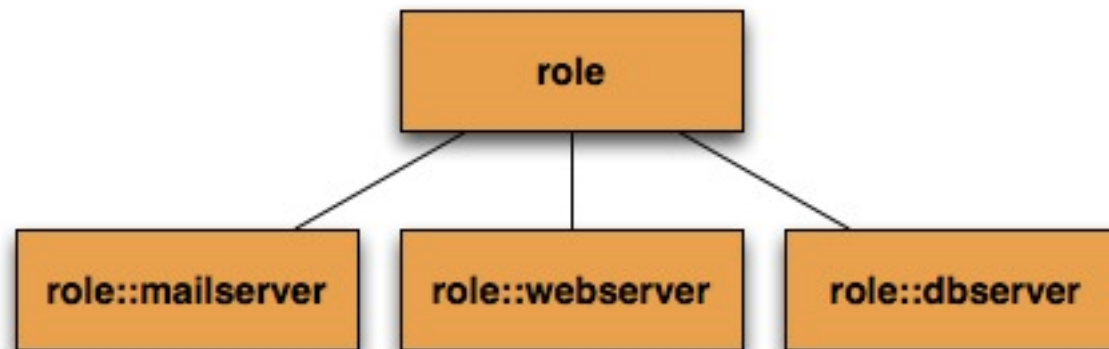
Roles + Classification

```
node 'ares.example.com' {  
  include role::dbserver  
}  
  
node 'zeus.example.com' {  
  include role::webserver  
}  
  
node 'hera.example.com' {  
  include role::webdbserver  
}  
  
node 'hermes.example.com' {  
  include role::webmail  
}
```

Roles + Classification

```
$role = hiera('role')  
include $role
```

Roles Structure



Roles Structure: Basic role

```
class role {  
    include profile::base  
}
```


Roles Structure: Specific roles

```
class role::webserver inherits role {  
  include profile::web  
}
```

```
class role::dbserver inherits role {  
  include profile::db  
}
```

```
class role::webdbserver inherits role {  
  include profile::web  
  include profile::db  
}
```

Roles Structure: Specific roles

```
class role::mailserver inherits role {  
    include profile::mailserver  
}
```

```
class role::webmail inherits role {  
    include profile::mailserver  
    include profile::webmail  
}
```

Roles Structure: Specific Roles

```
class role {  
    include profile::base  
}
```

```
class role::webdbserver inherits role {  
    include profile::web  
    include profile::db  
}
```

Remember

Classification is assigning roles to nodes.

Profiles

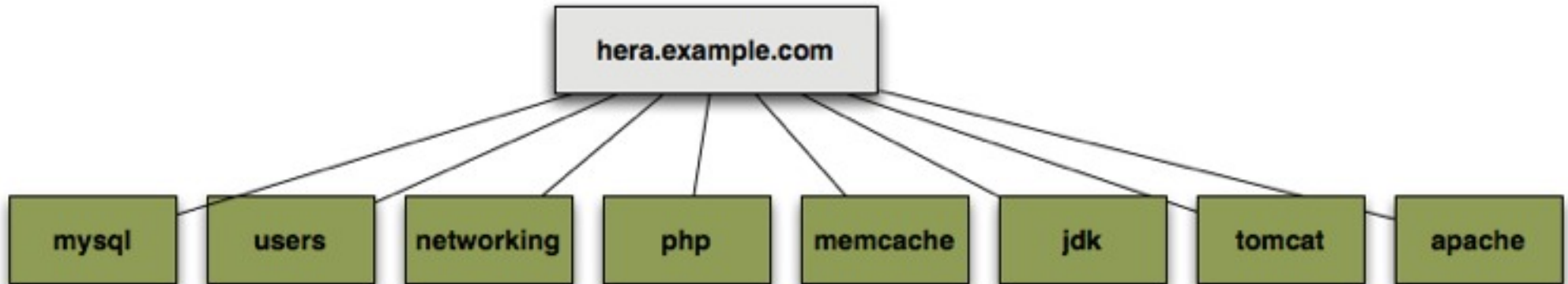
All your profiles are belong to Facebook.



Wireframing

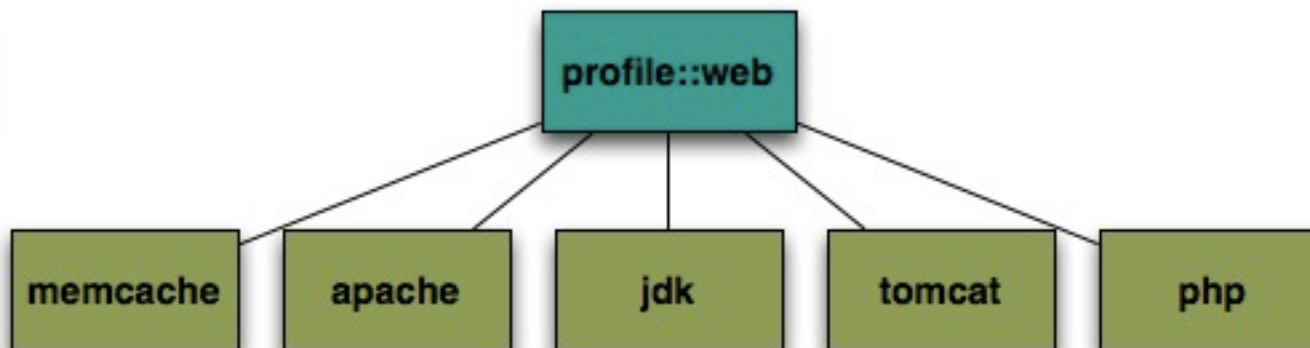
```
node 'hera.example.com' {  
  include role::webdbserver  
}  
  
class role {  
  include profile::base  
}  
  
class role::webdbserver inherits role {  
  include profile::web  
  include profile::db  
}
```

Wireframing



Wireframing

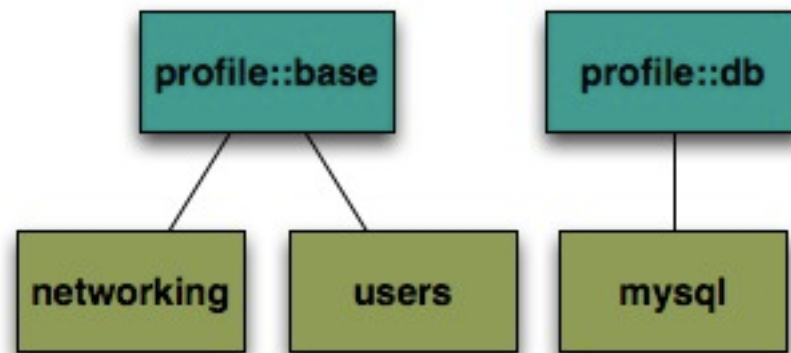
```
class profile::web {  
    include apache  
    include php  
    include tomcat  
    include jdk  
    include memcached  
}
```



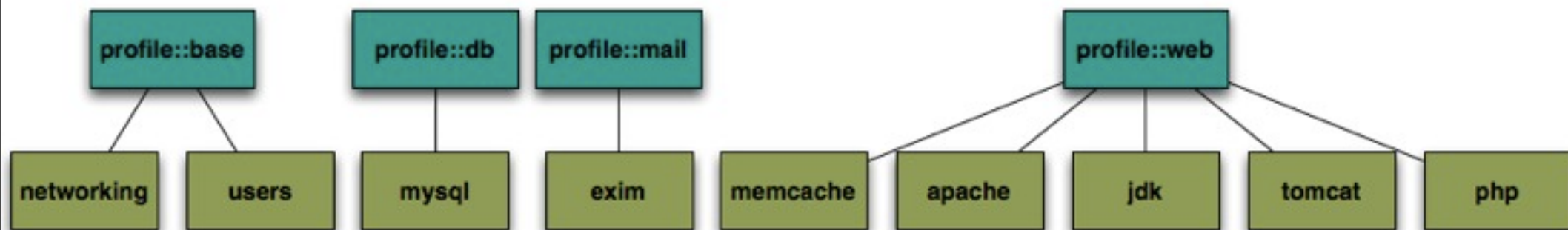
Wireframing

```
class profile::base {  
  include network  
  include users  
}
```

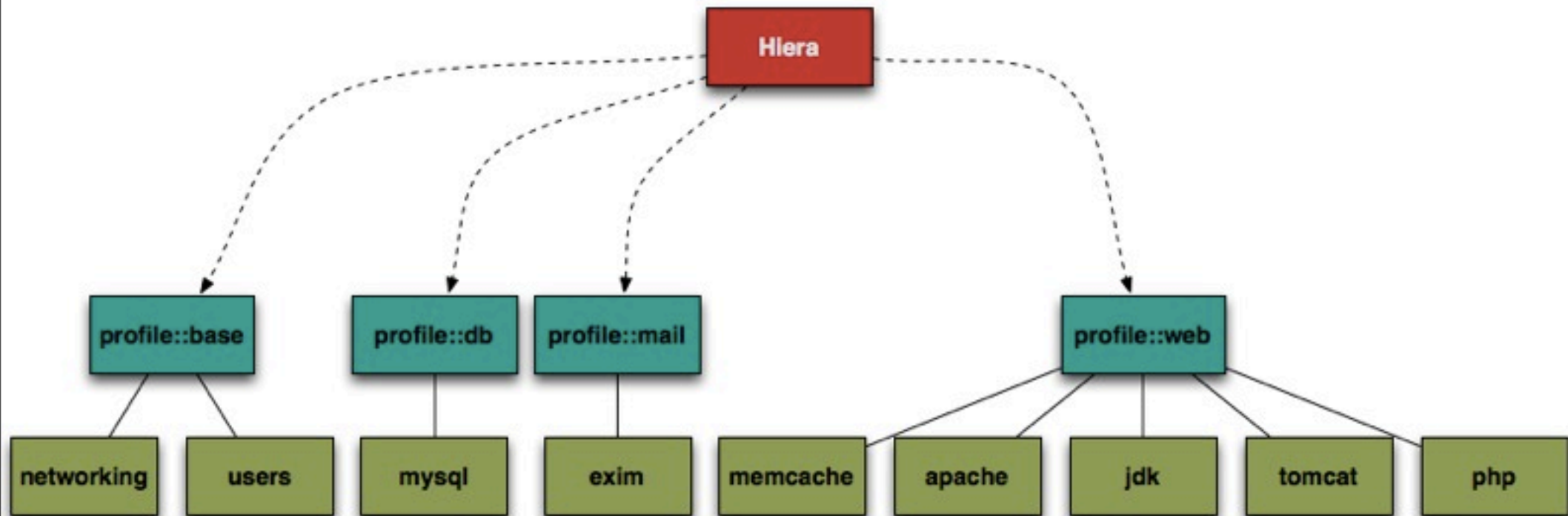
```
class profile::db {  
  include mysql  
}
```



Round Up: Profiles



Round Up: Profiles + Hiera



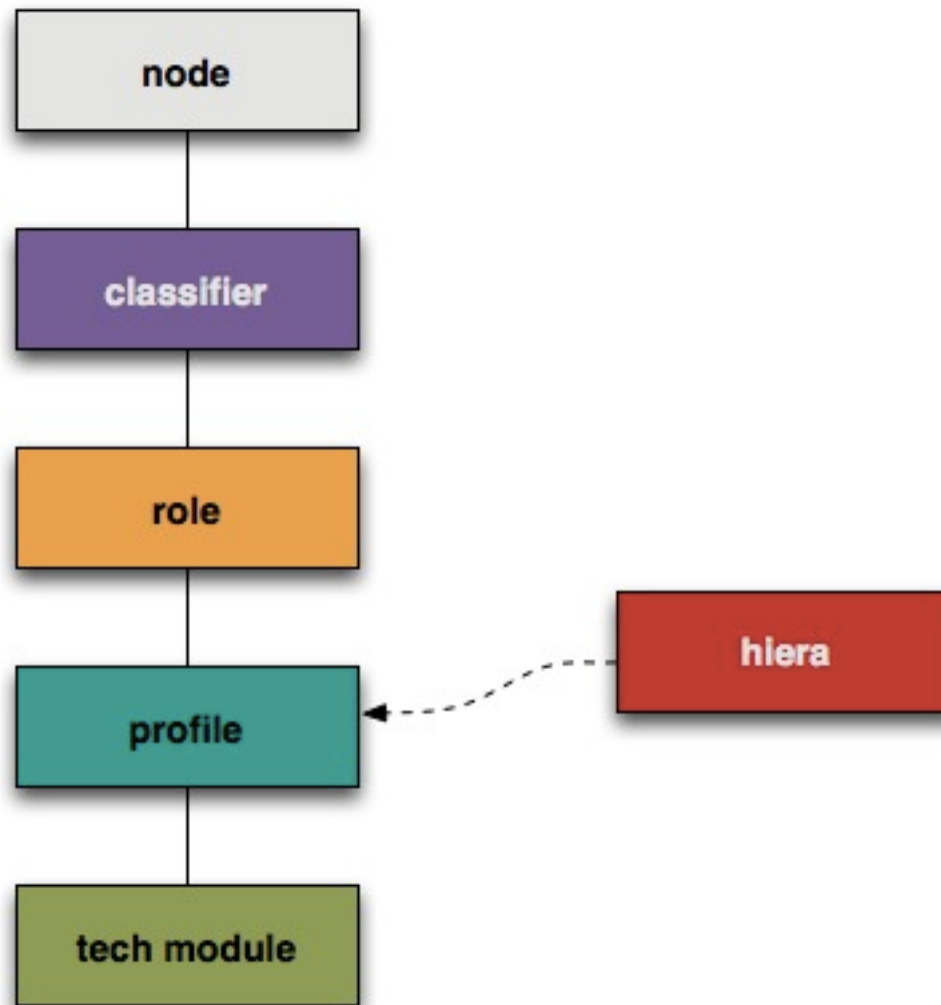
Profiles + Facts + Hierarchies = Win

- \$osfamily
- \$datacenter
- \$silo
- \$team
- \$product

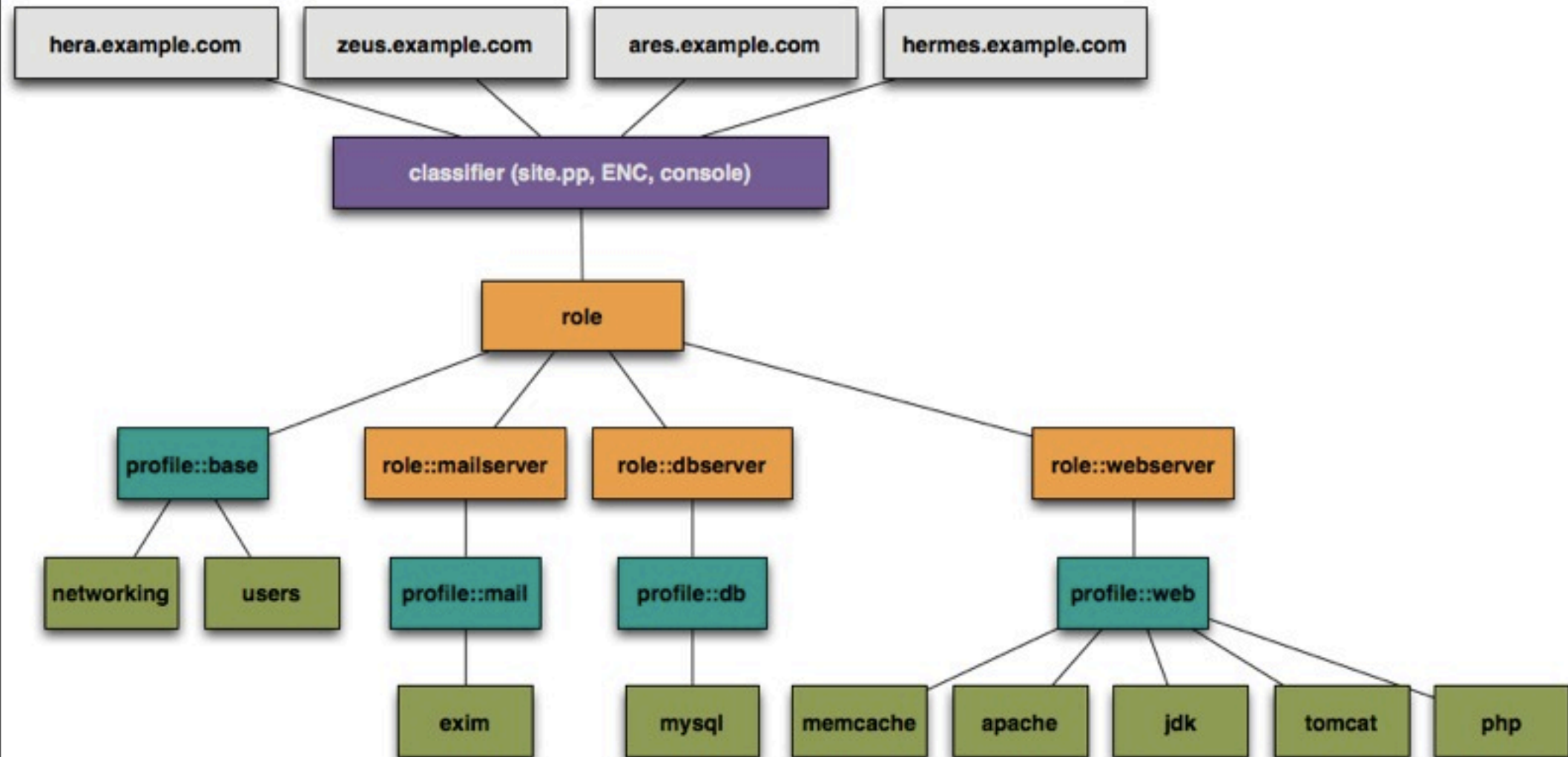
Profiles + Facts + Hiera = Win

```
class profile::rabbitmq {  
  include yumrepos::rabbit  
  include profile::erlang  
  class { 'rabbitmq':  
    rmq_user => hiera('rabbitmq::rmq_user'),  
    rmq_pass => hiera('rabbitmq::rmq_pass'),  
    rmq_port => hiera('rabbitmq::rmq_port'),  
    ip_min   => hiera('rabbitmq::ip_min'),  
    ip_max   => hiera('rabbitmq::ip_max'),  
    require => [  
      Class[ 'yumrepos::rabbit' ],  
      Package[ 'erlang' ],  
    ],  
  }  
}
```

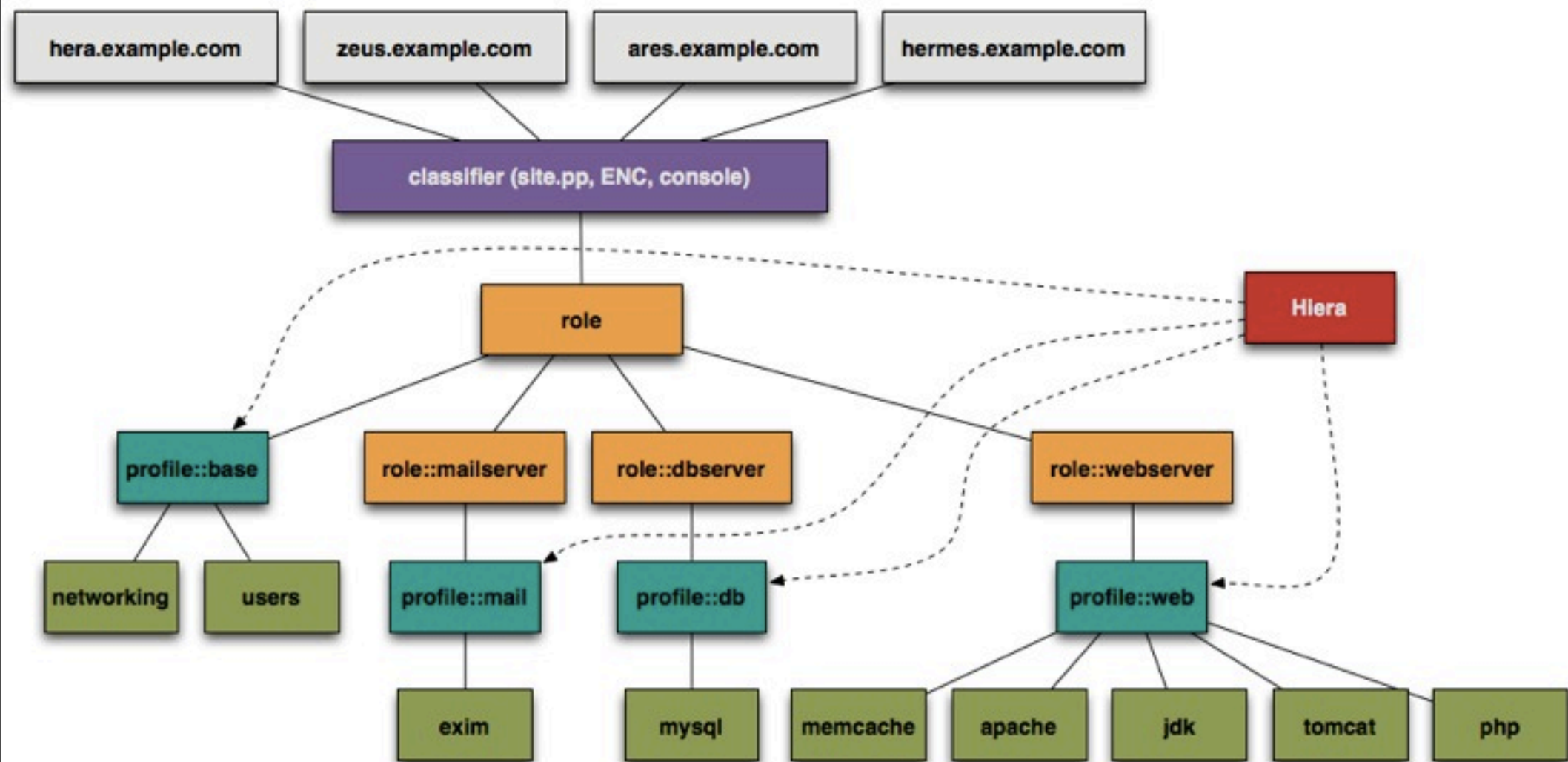
Roles + Profiles + Hieradata = Win



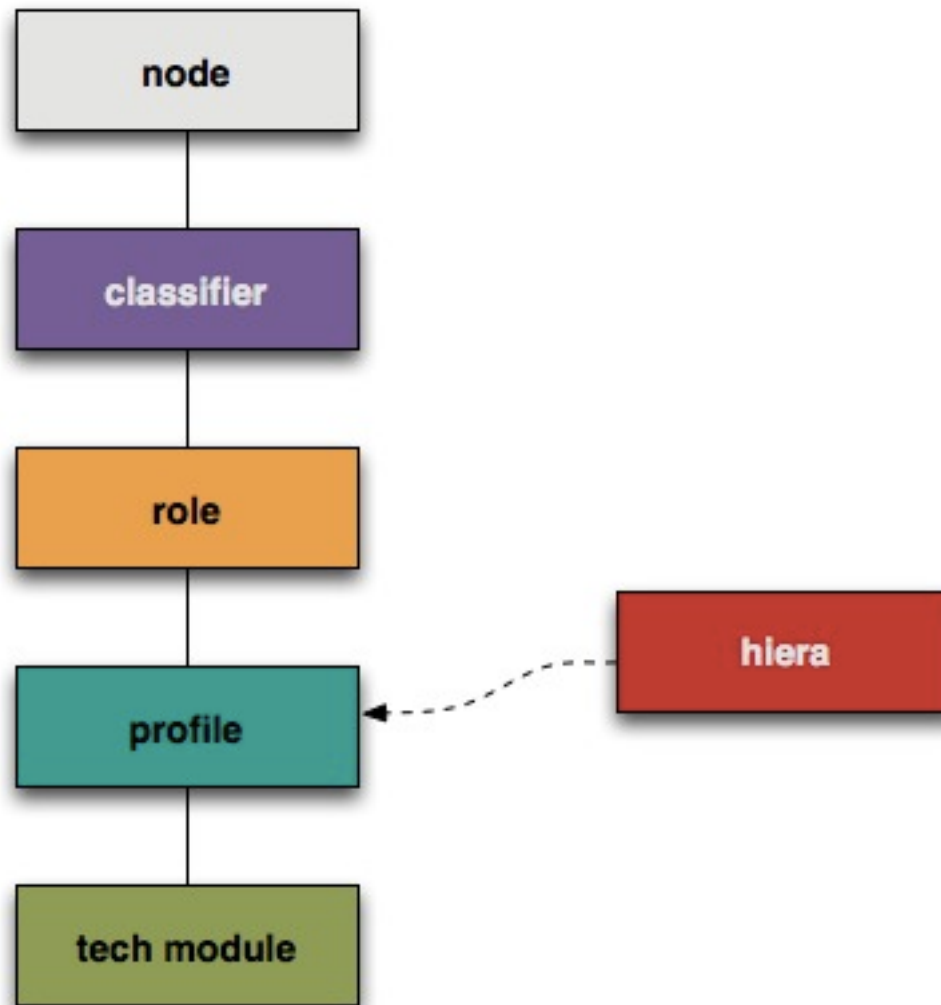
Roles + Profiles



Roles + Profiles + Hiera



Roles + Profiles + Hieradata = Win



Thank You

Hunter Haugen

Forge Module Engineer | Puppet Labs
@Hunnur

Collaborate. Automate. Ship.

puppetconf²⁰¹³

presented by  puppet
labs

Hands-On

Look with your hands, not with your eyes.



presented by  puppet
labs

Hands-On Infrastructure

- <http://puppet-vagrant-boxes.puppetlabs.com/>
- Training VM on USB sticks
- Invent the next-gen Iaas

Design Ideas: Construct R&P Pattern

- Suggestion: Apache + Mysql + Wordpress
- Suggestion: Exim + Dovecot
- Suggestion: Nginx + HAProxy
- Suggestion: Puppet + PuppetDB + PostgreSQL
- Suggestion: Tomcat + JBoss (?)
- Suggestion: Openstack (<trollface>)
- Suggestion: <http://goo.gl/JU81FR>



Follow us on Twitter @puppetlabs



youtube.com/puppetlabsinc



slideshare.net/puppetlabs

Collaborate. Automate. Ship.

puppetconf²⁰¹³

presented by  puppet
labs