

Enterprise Configuration Management

Puppet (doh),
puppetmanaged.org &
Puppet Common Modules

Byte-Code Meetup, 2009 - Jeroen van Meeuwen, rockstar

Puppet??

- System resource management
- Highly abstract manifests
 - Puppet speaks the local language
 - Heterogeneous environments
- System resource dependencies
- Modular, Extensible
- Staging environments (*dev -> test -> prod*)
- Active upstream
- Open community
- Awesome
- First designed outside of Red Hat

CFEngine

- Closed “Enterprise” development
- SUSE-like community (all @novell.com)
- Describe every tiny detail taking into account every single way to manage a given system resource:
 - yum/apt/rpm/dpkg/.../foo/bar/baz
- Inefficient
- Perfect for managing homogeneous environments
- Set up a different infra for each of your X-node clusters
- Not portable (between systems or environments)

Modules

- Consistent sets of system resources
 - “webserver” module
- Share with other puppet managed environments
- Fully integrated with staging environments
- a.k.a. “recipes”

Upstream's initiative:

Puppet Common Modules

Puppet Common Modules:

Let's find a way to build modules that
anyone can download,
and then just start using

K.I.S.S.

Two approaches:

- Little programs in manifests
- Have users set \$variables
- Make decisions based on those
- Not K.I.S.S.:
 - Requires setup
- Simple manifest
- No \$variable based decisions
- All variables are facts
- Put files in the correct place to customize
- As simple as possible, but
 - Not flexible enough

puppetmanaged.org

One way to go about
Puppet Common Modules

puppetmanaged.org?

- Collection of modules
- Multi-domain puppet management environments
 - Yes, using a single puppet master...
 - Yes, securely...
 - Apparently scalable
 - Used by:
 - Royal Dutch Meteorological Institute
 - University Medical Center Utrecht
 - Operator Groep Delft (*unimportant, but my employer :P*)
 - unknown number of people you only hear about when `git://git.puppetmanaged.org` is not available

Example module: YUM (pm.org)

- Simple straightforward module
- Manages */etc/yum.conf*, */etc/yum.repos.d/*.repo*
- Offers three profiles;
 - standard
 - testing
 - development
- Extensible; Create:

```
class yum::newprofile inherits yum::standard {  
  yum::repository { "mycompany_custom":  
    enable => true  
  }  
}
```

So how do you start using it?

- `cd /var/lib/puppet/modules/ (?)`
- `git clone git://git.puppetmanaged.org/puppet/yum/`
- include the “profile” in your node manifest
- If you have one custom repository, include:

```
yum::repository { “foo”: enable => true }
```

- If it's morning: Go get a *caffè doppio*
- If it's afternoon: Go get a beer
 - But not Heineken, that's Dutch

fedora[®]