

Creating modules

```
# cd /etc/puppet/environments/production/modules
# puppet module generate imran-mediawiki --environment production
# mv imran-mediawiki/ mediawiki
# cd mediawiki
# ls -ltr
# cd manifests
# vi init.pp
```

Customizing Modules

Login to puppetmaster

```
# cd /etc/puppet/environments/production/modules/mediawiki/manifests
# vi init.pp
```

```
class mediawiki {

    $phpmysql = $osfamily ? {
        'redhat' => 'php-mysql',
        'debian' => 'php5-mysql',php
        default => 'php-mysql',
    }

    package { $phpmysql:
        ensure => 'present',
    }
}
```

Open nodes.pp file and invoke mediawiki class

```
# cd /etc/puppet/environments/production/manifests
# vi nodes.pp
```

```
node 'wiki'{

    class {'linux': }
    class {'mediawiki':}

}
```

```
node 'wikitest' {

    class {'linux': }
    class {'mediawiki':}
```

```
}  
.....Rest of the file.....
```

Run puppet agent from both the nodes to verify

```
# puppet agent --verbose --no-daemonize --onetime
```

Conditionals

Login to puppetmaster

```
# cd /etc/puppet/environments/production/modules/mediawiki/manifests
```

```
# vi init.pp
```

```
class mediawiki {
```

```
    $phpmysql = $osfamily ? {  
        'redhat' => 'php-mysql',  
        'debian' => 'php5-mysql',  
        default  => 'php-mysql',  
    }
```

```
    package { $phpmysql:  
        ensure => 'present',  
    }
```

```
    if $osfamily == 'redhat' {  
        package {'php-xml':  
            ensure => 'present',  
        }  
    }
```

```
}
```

Run puppet agent from both the nodes to verify

```
# puppet agent --verbose --no-daemonize --onetime
```

Downloading from puppet forge

<https://forge.puppet.com/>

Apache module

Login to puppetmaster

```
# cd /etc/puppet/environments/production/modules
```

```
# puppet module install puppetlabs-apache --modulepath  
/etc/puppet/environments/production/modules/
```

```
# ls -ltr
```

```
# cd mediawiki/manifests/  
# vi init.pp
```

```
class mediawiki {  
  
    $phpmysql = $osfamily ? {  
        'redhat' => 'php-mysql',  
        'debian' => 'php5-mysql',  
        default => 'php-mysql',  
    }  
  
    package { $phpmysql:  
        ensure => 'present',  
    }  
  
    if $osfamily == 'redhat' {  
        package {'php-xml':  
            ensure => 'present',  
        }  
    }  
}
```

```
class {'::apache':  
    docroot    => '/var/www/html',  
    mpm_module => 'prefork',  
    subscribe => Package[$phpmysql],  
}
```

```
class {'::apache::mod::php': }  
}
```

Run puppet agent from both the nodes to verify

```
# puppet agent --verbose --no-daemonize --onetime  
# service httpd status (centos)  
# service apache2 status (ubuntu)
```

VCSREPO MODULE

Login to puppetmaster

```
# cd /etc/puppet/environments/production/modules  
# puppet module install puppetlabs-vcsrepo --modulepath  
/etc/puppet/environments/production/modules/
```

```
# ls -ltr
```

```
# cd mediawiki/manifests/  
# vi init.pp
```

```
class mediawiki {
```

```
  $phpmysql = $osfamily ? {  
    'redhat' => 'php-mysql',  
    'debian' => 'php5-mysql',  
    default => 'php-mysql',  
  }
```

```
  package { $phpmysql:  
    ensure => 'present',  
  }
```

```
  if $osfamily == 'redhat' {  
    package {'php-xml':  
      ensure => 'present',  
    }  
  }
```

```
class { '::apache':  
  docroot    => '/var/www/html',  
  mpm_module => 'prefork',  
  subscribe => Package[$phpmysql],  
}
```

```
class {'::apache::mod::php': }
```

```
vcsrepo { '/var/www/html':  
  ensure   => 'present',  
  provider => 'git',  
  source   => "https://github.com/wikimedia/mediawiki.git",  
  revision => 'REL1_23',  
}  
}
```

Run puppet agent from both the nodes to verify

```
# puppet agent --verbose --no-daemonize --onetime
```

Resource Ordering

```
# vi init.pp  
class mediawiki {
```

```

$phpmysql = $osfamily ? {
  'redhat' => 'php-mysql',
  'debian' => 'php5-mysql',
  default  => 'php-mysql',
}

package { $phpmysql:
  ensure => 'present',
}

if $osfamily == 'redhat' {
  package {'php-xml':
    ensure => 'present',
  }
}

class { '::apache':
  docroot    => '/var/www/html',
  mpm_module => 'prefork',
  subscribe => Package[$phpmysql],
}

class {'::apache::mod::php': }

vcsrepo { '/var/www/html':
  ensure    => 'present',
  provider  => 'git',
  source    => "https://github.com/wikimedia/mediawiki.git",
  revision  => 'REL1_23',
}

file {'/var/www/html/index.html':
  ensure => 'absent',
}
}

```

Run puppet agent from both the nodes to verify
 # puppet agent --verbose --no-daemonize --onetime

This puppet run is gonna remove the index.html file after trying vcsrepo so ultimately it will fail.

For this will use resource ordering.

Login to puppetmaster

```
# cd /etc/puppet/environments/production/modules//mediawiki/manifests
```

```
# vi init.pp
```

```
class mediawiki {
```

```
    $phpmysql = $osfamily ? {  
        'redhat' => 'php-mysql',  
        'debian' => 'php5-mysql',  
        default => 'php-mysql',  
    }
```

```
    package { $phpmysql:  
        ensure => 'present',  
    }
```

```
    if $osfamily == 'redhat' {  
        package {'php-xml':  
            ensure => 'present',  
        }  
    }
```

```
class { '::apache':  
    docroot    => '/var/www/html',  
    mpm_module => 'prefork',  
    subscribe => Package[$phpmysql],  
}
```

```
class {'::apache::mod::php': }
```

```
vcsrepo { '/var/www/html':  
    ensure    => 'present',  
    provider  => 'git',  
    source    => "https://github.com/wikimedia/mediawiki.git",  
    revision  => 'REL1_23',  
}
```

```
file {'/var/www/html/index.html':  
    ensure => 'absent',  
}
```

File['/var/www/html/index.html'] -> Vcsrepo ['/var/www/html']

```
}
```

Run puppet agent from both the nodes to verify

```
# touch /var/www/html/index.html
```

```
# puppet agent --verbose --no-daemonize --onetime
```

MYSQL MODULE

Login to puppetmaster

```
# cd /etc/puppet/environments/production/modules
```

```
# puppet module install puppetlabs-mysql --modulepath
```

```
/etc/puppet/environments/production/modules/
```

```
# ls -ltr
```

```
# cd mediawiki/manifests/
```

```
# vi init.pp
```

```
class mediawiki {
```

```
    $phpmysql = $osfamily ? {  
        'redhat' => 'php-mysql',  
        'debian' => 'php5-mysql',  
        default  => 'php-mysql',  
    }
```

```
    package { $phpmysql:  
        ensure => 'present',  
    }
```

```
    if $osfamily == 'redhat' {  
        package {'php-xml':  
            ensure => 'present',  
        }  
    }
```

```
class { '::apache':  
    docroot    => '/var/www/html',  
    mpm_module => 'prefork',  
    subscribe => Package[$phpmysql],  
}
```

```
class {'::apache::mod::php': }
```

```
vcsrepo { '/var/www/html':
```

```
ensure => 'present',
provider => 'git',
source => "https://github.com/wikimedia/mediawiki.git",
revision => 'REL1_23',
}
```

```
file {'/var/www/html/index.html':
  ensure => 'absent',
}
```

```
File['/var/www/html/index.html'] -> Vcsrepo ['/var/www/html']
```

```
class {'::mysql::server':
  root_password => 'training',
}
```

```
}
```

Run puppet agent from both the nodes to verify
puppet agent --verbose --no-daemonize --onetime

FIREWALL MODULE

Login to puppetmaster

```
# cd /etc/puppet/environments/production/modules
# puppet module install puppetlabs-firewall --modulepath
/etc/puppet/environments/production/modules/
```

```
# ls -ltr
# cd mediawiki/manifests/
# vi init.pp
class mediawiki {
```

```
  $phpmysql = $osfamily ? {
    'redhat' => 'php-mysql',
    'debian' => 'php5-mysql',
    default => 'php-mysql',
  }
```

```
  package { $phpmysql:
    ensure => 'present',
  }
```

```
  if $osfamily == 'redhat' {
```



```
package {'php-xml':  
  ensure => 'present',  
}  
}
```

```
class {'::apache':  
  docroot    => '/var/www/html',  
  mpm_module => 'prefork',  
  subscribe => Package[$phpmysql],  
}
```

```
class {'::apache::mod::php': }
```

```
vcsrepo { '/var/www/html':  
  ensure    => 'present',  
  provider  => 'git',  
  source    => "https://github.com/wikimedia/mediawiki.git",  
  revision  => 'REL1_23',  
}
```

```
file {'/var/www/html/index.html':  
  ensure => 'absent',  
}
```

```
File['/var/www/html/index.html'] -> Vcsrepo ['/var/www/html']
```

```
class {'::mysql::server':  
  root_password => 'training',  
}
```

```
class { 'firewall': }
```

```
firewall { '000 allow http access':  
  port    => '80',  
  proto   => 'tcp',  
  action  => 'accept',  
}
```

```
}
```

```
}
```

Run puppet agent from both the nodes to verify
puppet agent --verbose --no-daemonize --onetime