# Templates

# Static Files

- Inflexible
- Repetitive
- Content Explosion
- Managebility

# Dynamically Generated Files

# == Templates

# Lets see it in action..

- We will create a template to manage the vhosts instead of static config

- This will take site_name and site_domain variables for each vhost and generate files dynamically.

# Vhost.conf.erb

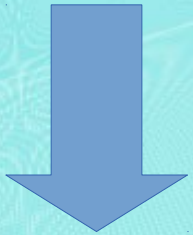
```
$ mkdir modules/nginx/templates
```

# nginx/templates/vhost.conf.erb

```
server {
        listen 80;
        root /var/www/<%=  @site_name  %>;
        server_name  <%= @site_domain %>;
}
```

# Change file resource in configure.pp

```
file { '/etc/nginx/conf.d/default.conf':
        source => 'puppet:///modules/nginx/cinema.conf',
    notify => Service['nginx'],
```



```
  file { '/etc/nginx/conf.d/default.conf':
      content => template('nginx/vhost.conf.erb'),
      notify => Service['nginx'],
}
```

# Create Webroot for the Site

Wherever there is cinema, change it to ${site_name}

Make sure to enclose the complete string containing variables in double quotes.

```
file { "/var/www/cinema":
       ensure => directory,
     }
```

```
file { "/var/www/${site_name}":
              ensure => directory,
     }
```

# Update file resource that servers index.html

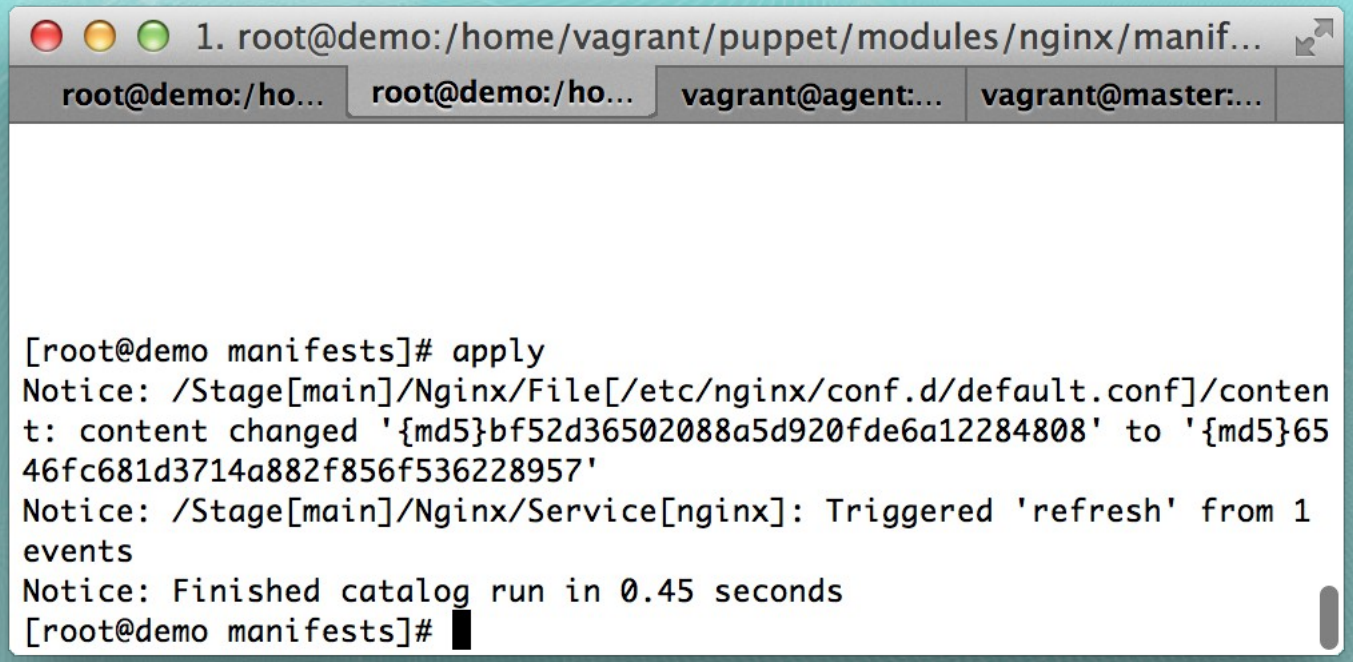
- Modify /var/www/cinema/index.html to
- /var/www/${site_name}/index.html

**cinema** **${site_name}**

# nodes.pp

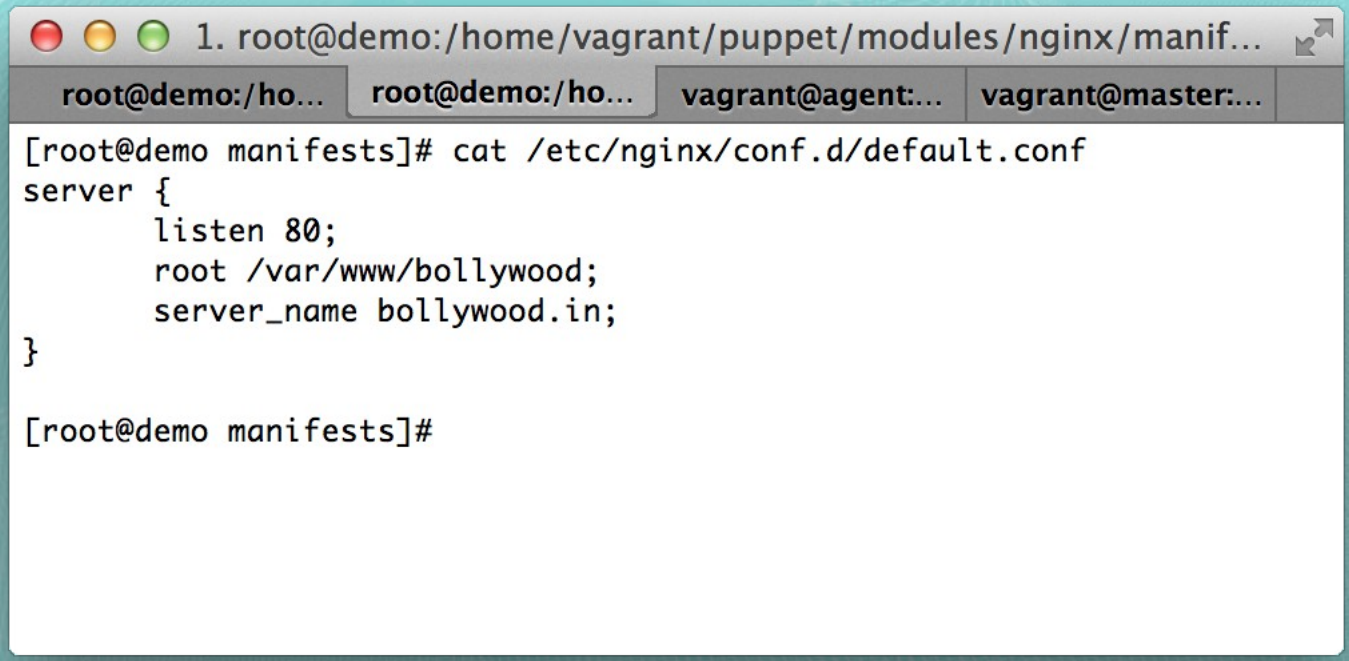
```
node 'demo' {
        $site_name = 'bollywood'
        $site_domain = 'bollywood.in'
        include nginx
}
```

# $ puppet apply /vagrant/puppet/manifests/



# Validate

$ cat /etc/nginx/conf.d/default.conf



# Dynamic Configurations

Now that we have template, we could drive configs using parameters alone.

```
node 'demo2' {
        $site_name = 'kollywood'
        $site_domain = 'kollywood.in'
        include nginx
}
```

# Facts in Templates

Facts can be used just like variables inside templates.

Whats even better is you dont need to define values for the facts ! Those are automatic.

# ERB

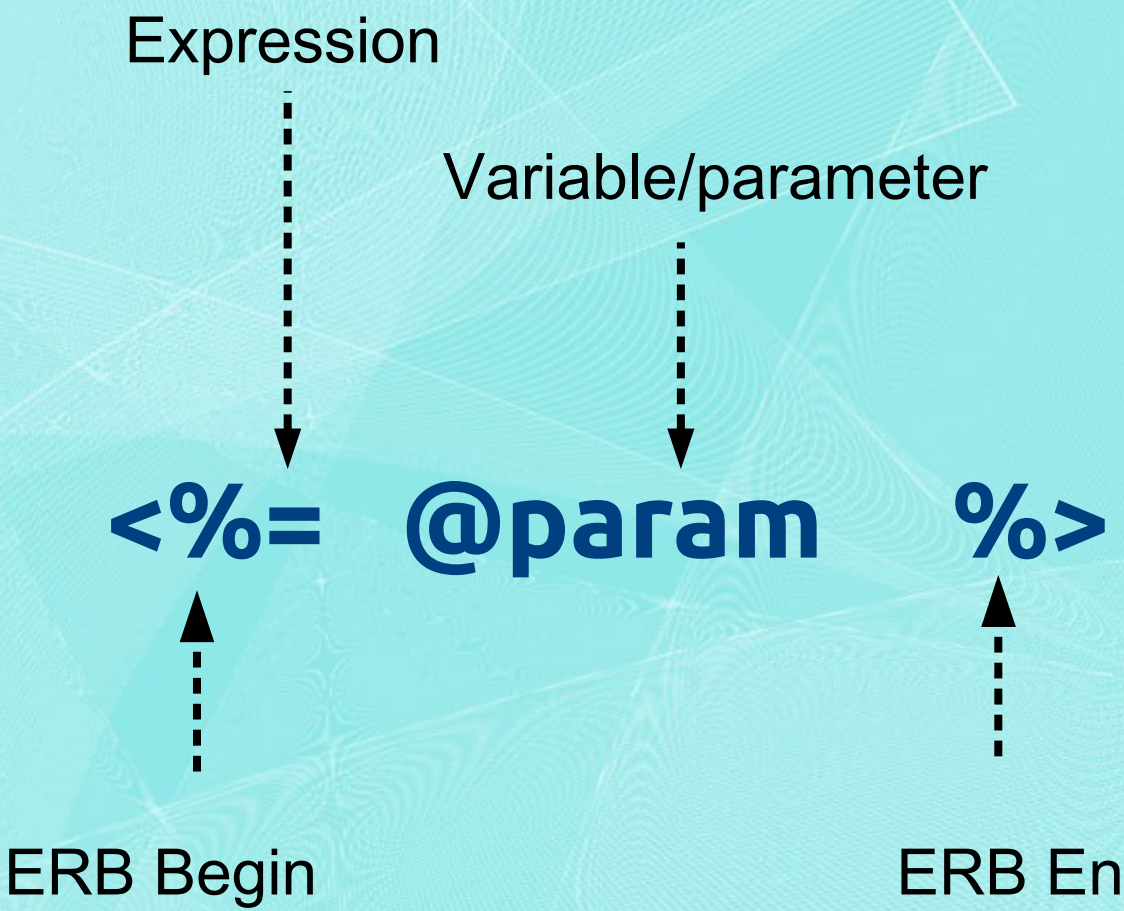
- Stands for Embedded Ruby
- We are embedding ruby into the text files.

Lets now look at the ERB Tags that we have added to the template.

# ERB

- ERB is a common ruby based template language
- Looks similar to normal config files
- Contains occasional tags <% %>

.


Expression
Variable/parameter
<%= @param %>
ERB Begin
ERB End

# Expression

```
<%= expression %>
```

Print the value of the expression e.g. variables/facts.

# Code

```
<% Ruby code %>
```

Execute the code, but do not print the value

```
<% Ruby code -%>
```

Suppress blank lines

# Exercise

Extend the nginx module by templatizing index.html and include some facts, erb logic etc.

# Exercise

Templatize index.html which will display the following,

- IP Address
- Operating System
- System Memory

**Steps**

- Create index.html.erb
- Add ERB Logic
- Update file resource to use template

# Validate

- Edit /etc/hosts and add an alias “**bollywood.in**” to 127.0.0.1 or 127.0.0.2

Run the following command to validate your changes

# curl bollywood.in