

# Desired State and Data

Extracting the Content for Clarity

# Objectives

After completing this module, you should be able to

- Explain when to use a template resource
- Create a template file
- Use ERB tags to display node data in a template
- Define a template resource



## Cleaner Recipes

In the last section we updated our two cookbooks to display information about our node.

We added this content to the file resource in their respective recipes.

# Apache Recipe

 `~/cookbooks/apache/recipes/server.rb`

```
package 'httpd'

file '/var/www/html/index.html' do
  content "<h1>Hello, world!</h1>
<h2>ipaddress: #{node['ipaddress']}</h2>
<h2>hostname: #{node['hostname']}</h2>
"
end

service 'httpd' do
  action [ :enable, :start ]
end
```

# Double Quotes Close Double Quotes



Double quoted strings are terminated by double quotes.

```
"<h1 style="color: red;">Hello, World!</h1>"
```

The diagram illustrates the concept of double quotes closing double quotes. It shows an HTML tag: `"<h1 style="color: red;">Hello, World!</h1>"`. The opening double quote at the start of the tag and the closing double quote at the end of the tag are highlighted with red rectangular boxes. Two blue arrows point upwards from the bottom of the slide to these two red boxes, indicating that the outer double quotes are terminated by the inner double quotes.

# Backslash



We can use double-quotes as long as we prefix them with a backslash.

```
"<h1 style=\"color: red;\">Hello, World!</h1>"
```





## Backslash

Backslashes are reserved characters. So to use them you need to use a backslash.

"Root Path: \"



# CONCEPTS

## Backslash



Backslashes are reserved characters. So to use them you need to use a backslash.

"Root Path: \\"



# Unexpected Formatting

```
file '/etc/motd' do
  content 'This is the first line of the file.
          This is the second line. If I try and line it up...'

  Don't even think about pasting ASCII ART in here!
'
end
```

```
This is the first line of the file.
          This is the second line. If I try and line
it up...
```

```
Don't even think about pasting ASCII ART in here!
```

## Copy Paste



This process is definitely error prone. Especially because a human has to edit the file again before it is deployed.

## What We Need

We need the ability to store the data in another file, which is in the native format of the file we are writing out but that still allows us to insert ruby code...

...specifically, the node attributes we have defined.





## GL: Cleaner Recipes

*Adding the node attributes to our recipes did make it harder to read.*

### Objective:

- ❑ Decide which resource will help us address this issue



## GL: Let's Check the Docs...

Use the file resource to manage files directly on a node.

Use the **cookbook\_file** resource to copy a file from a cookbook's **/files** directory. Use the **template** resource to create a file based on a template in a cookbook's **/templates** directory. And use the **remote\_file** resource to transfer a file to a node from a remote location.

[https://docs.chef.io/resource\\_file.html](https://docs.chef.io/resource_file.html)

## remote\_file



Use the **remote\_file** resource to transfer a file from a remote location using file specificity. This resource is similar to the file resource.

[https://docs.chef.io/resource\\_remote\\_file.html](https://docs.chef.io/resource_remote_file.html)

## cookbook\_file



Use the **cookbook\_file** resource to transfer files from a sub-directory of COOKBOOK\_NAME/files/ to a specified path located on a host that is running the chef-client.

[https://docs.chef.io/resource\\_cookbook\\_file.html](https://docs.chef.io/resource_cookbook_file.html)

# Demo: cookbook\_file's Source Match Up

```
$ tree cookbooks/apache/files/default
files/default
└─ index.html
```

0 directories, 1 file

```
cookbook_file '/var/www/index.html' do
  source 'index.html'
end
```





## Template

A cookbook template is an Embedded Ruby (ERB) template that is used to generate files ... Templates may contain Ruby expressions and statements and are a great way to...

Use the template resource to add cookbook templates to recipes; place the corresponding Embedded Ruby (ERB) template in a cookbook's /templates directory.

[https://docs.chef.io/resource\\_template.html](https://docs.chef.io/resource_template.html)

# Demo: Template File's Source Matches Up

```
$ tree cookbooks/apache/templates/default
cookbooks/apache/templates/default
├── index.html.erb
```

```
0 directories, 1 file
```

```
template '/var/www/index.html' do
  source 'index.html.erb'
end
```



# Template



To use a template, two things must happen:

1. A template resource must be added to a recipe
2. An Embedded Ruby (ERB) template must be added to a cookbook

[https://docs.chef.io/resource\\_template.html#using-templates](https://docs.chef.io/resource_template.html#using-templates)



## GL: Cleaner Apache Recipe

*Adding the node attributes to the index page did make it harder to read the recipe.*

### Objective:

- ☐ Create a template with chef generate
- ☐ Define the contents of the ERB template
- ☐ Change the file resource to the template resource in the 'apache' cookbook

# GL: What Can chef generate Do?



```
$ chef generate --help
```

```
Usage: chef generate GENERATOR [options]
```

```
Available generators:
```

app	Generate an application repo
cookbook	Generate a single cookbook
recipe	Generate a new recipe
attribute	Generate an attributes file
template	Generate a file template
file	Generate a cookbook file
lwrp	Generate a lightweight resource/provider
repo	Generate a Chef policy repository
policyfile	Generate a Policyfile for use with the install/push commands (experimental)

# GL: What Can chef generate template Do?



```
$ chef generate template --help
```

```
Usage: chef generate template [path/to/cookbook] NAME [options]
```

-C, --copyright COPYRIGHT	Name of the copyright holder - defaults to 'The Authors'
-m, --email EMAIL	Email address of the author - defaults to ...
-a, --generator-arg KEY=VALUE	Use to set arbitrary attribute KEY to VALUE in the
-I, --license LICENSE	all_rights, apache2, mit, gplv2, gplv3 - defaults to
-s, --source SOURCE_FILE	Copy content from SOURCE_FILE
-g GENERATOR_COOKBOOK_PATH,	Use GENERATOR_COOKBOOK_PATH for the
code_generator	
--generator-cookbook	

# GL: Use chef to Generate a Template



```
$ cd ~
```

```
$ chef generate template cookbooks/apache index.html
```

```
Compiling Cookbooks...
```

```
Recipe: code_generator::template
```

```
  * directory[cookbooks/apache/templates/default] action create
```

```
    - create new directory cookbooks/apache/templates/default
```

```
  * template[cookbooks/apache/templates/default/index.html.erb] action create
```

```
    - create new file cookbooks/apache/templates/default/index.html.erb
```

```
    - update content in file cookbooks/apache/templates/default/index.html.erb  
from none to e3b0c4
```

```
    (diff output suppressed by config)
```

# GL: Lets Look at the Template File



```
$ tree cookbooks/apache/templates
```

```
cookbooks/apache/templates/  
└─ default  
    └─ index.html.erb
```

```
1 directory, 1 file
```





## Cleaner Recipes

*Adding the node attributes to the default page did make it harder to read the recipe.*

### Objective:

- ✓ Create a template with chef generate
- ❑ Define the contents of the ERB template
- ❑ Change the file resource to the template resource in the 'apache' cookbook

# CONCEPTS

## ERB



An Embedded Ruby (ERB) template allows Ruby code to be embedded inside a text file within specially formatted tags.

Ruby code can be embedded using expressions and statements.

<https://docs.chef.io/templates.html#variables>

# Text Within an ERB Template

```
<% if (50 + 50) == 100 %>
```

```
50 + 50 = <%= 50 + 50 %>
```

```
<% else %>
```

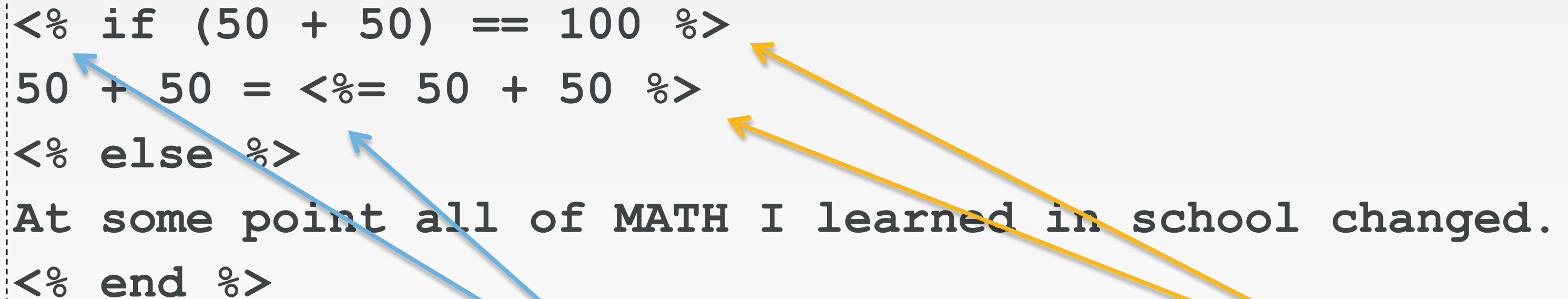
```
At some point all of MATH I learned in school changed.
```

```
<% end %>
```

Each ERB tag has a beginning tag and a matched ending tag.

# Text Within an ERB Template

```
<% if (50 + 50) == 100 %>  
50 + 50 = <%= 50 + 50 %>  
<% else %>  
At some point all of MATH I learned in school changed.  
<% end %>
```

The diagram shows an ERB template snippet enclosed in a dashed box. It contains an if-else conditional. Blue arrows point from the opening tags to their corresponding closing tags: from the first opening tag to the first closing tag, from the opening tag of the else block to its closing tag, and from the opening tag of the end block to its closing tag. Yellow arrows point from the closing tags back to the opening tags, indicating the matching pairs.

Each ERB tag has a beginning tag and a matched ending tag.

# Text Within an ERB Template

```
<% if (50 + 50) == 100 %>
```

```
50 + 50 = <%= 50 + 50 %>
```

```
<% else %>
```

```
At some point all of MATH I learned in school changed.
```

```
<% end %>
```

Each ERB tag has a beginning tag and a matched ending tag.

# Text Within an ERB Template

```
<% if (50 + 50) == 100 %>
```

```
50 + 50 = <%= 50 + 50 %>
```

```
<% else %>
```

```
At some point all of MATH I learned in school changed.
```

```
<% end %>
```

Executes the ruby code within the brackets and do not display the result.

# Text Within an ERB Template

```
<% if (50 + 50) == 100 %>
```

```
50 + 50 = <%= 50 + 50 %>
```

```
<% else %>
```

```
At some point all of MATH I learned in school changed.
```

```
<% end %>
```

Executes the ruby code within the brackets and display the results.



## The Angry Squid





# GL: Move Our Source to the Template



```
~/cookbooks/apache/templates/default/index.html.erb
```

```
<h1>Hello, world!</h1>
<h2>ipaddress:  #{node['ipaddress']}</h2>
<h2>hostname:   #{node['hostname']}</h2>
```

# GL: Replace String Interpolation with ERB

 `~/cookbooks/apache/templates/default/index.html.erb`

```
<h1>Hello, world!</h1>
```

```
<h2>ipaddress: <%= node['ipaddress'] %></h2>
```

```
<h2>hostname: <%= node['hostname'] %></h2>
```



## Cleaner Recipes

*Adding the node attributes to the default page did make it harder to read the recipe.*

### Objective:

- ✓ Create a template with chef generate
- ✓ Define the contents of the ERB template
- ❑ Change the file resource to the template resource in the 'apache' cookbook

# GL: Remove the Existing Content Attribute

 `~/cookbooks/apache/recipes/server.rb`

```
package 'httpd'

file '/var/www/html/index.html' do
  content "<h1>Hello, world!</h1>
<h2>ipaddress: #{node['ipaddress']}</h2>
<h2>hostname: #{node['hostname']}</h2>
"
end

service 'httpd' do
  action [ :enable, :start ]
end
```

# GL: Change File Resource to a Template Resource

 ~/cookbooks/apache/recipes/server.rb

```
package 'httpd'
```

```
template '/var/www/html/index.html' do
```

```
end
```

```
service 'httpd' do
```

```
  action [ :enable, :start ]
```

```
end
```

# What to Specify as the Source?

```
~/cookbooks/apache/recipes/server.rb
```

```
package 'httpd'

template '/var/www/html/index.html' do
  source "?????????????"
end

service 'httpd' do
  action [ :enable, :start ]
end
```

# GL: Viewing the Partial Path to the Template



```
$ tree cookbooks/apache/templates/default
```

```
cookbooks/apache/templates/default/
```

```
└─ index.html.erb
```

```
0 directories, 1 file
```

# GL: Update the Source Attribute

```
~/cookbooks/apache/recipes/server.rb
```

```
package 'httpd'

template '/var/www/html/index.html' do
  source 'index.html.erb'
end

service 'httpd' do
  action [ :enable, :start ]
end
```





## Cleaner Recipes

*Adding the node attributes to the default page did make it harder to read the recipe.*

### Objective:

- ✓ Create a template with chef generate
- ✓ Define the contents of the ERB template
- ✓ Change the file resource to the template resource in the 'apache' cookbook



## Lab: Update the Version

- ☐ Use kitchen test on the "apache" cookbook
- ☐ Use chef-client to apply the "apache" cookbook's "default" recipe
- ☐ Update the "apache" cookbook's version for this patch
- ☐ Commit the changes

# Lab: Test the Cookbook



```
$ cd ~/cookbooks/apache  
$ kitchen test
```

```
-----> Starting Kitchen (v1.4.0)  
-----> Cleaning up any prior instances of <default-centos-67>  
-----> Destroying <default-centos-67>...  
        Finished destroying <default-centos-67> (0m0.00s) .  
-----> Testing <default-centos-67>  
-----> Creating <default-centos-67>...  
        Sending build context to Docker daemon  2.56 kB  
        Sending build context to Docker daemon  
        Step 0 : FROM centos:centos6  
        ---> 72703a0520b7
```

# Lab: Change Directories and Apply the Cookbook



```
$ cd ~
```

```
$ sudo chef-client --local-mode -r "recipe[apache]"
```

```
[2016-09-16T14:18:05+00:00] WARN: No config file found or specified on command line,  
using command line options.
```

```
Starting Chef Client, version 12.3.0
```

```
resolving cookbooks for run list: ["apache"]
```

```
Synchronizing Cookbooks:
```

```
- apache
```

```
Compiling Cookbooks...
```

```
[2016-09-16T14:18:09+00:00] WARN: Cloning resource attributes for service[httpd] from  
prior resource (CHEF-3694)
```

```
[2016-09-16T14:18:09+00:00] WARN: Previous service[httpd]: /root/.chef/local-mode-  
cache/cache/cookbooks/apache/recipes/server.rb:8:in `from_file'
```

```
[2016-09-16T14:18:09+00:00] WARN: Current service[httpd]: /root/.chef/local-mode-  
cache/ ...
```

# Lab: Update the Cookbook's Patch Number

```
~/cookbooks/apache/metadata.rb
```

```
name                'apache'  
maintainer           'The Authors'  
maintainer_email     'you@example.com'  
license              'all_rights'  
description           'Installs/Configures apache'  
long_description      'Installs/Configures apache'  
version              '0.2.1'
```

## Lab: Commit the Changes

```
$ cd ~/cookbooks/apache  
$ git add .  
$ git status  
$ git commit -m "Update default recipe to use  
template"
```





## Lab: Update the Version

- ✓ Use kitchen test on the "apache" cookbook
- ✓ Use chef-client to apply the "apache" cookbook's "default" recipe
- ✓ Update the "apache" cookbook's version for this patch
- ✓ Commit the changes



## Lab: Use the Template

For the "workstation" cookbook:

- ☐ Use chef generate to create a template named "motd.erb".
- ☐ Copy the source attribute from the file named '/etc/motd' into the template file "motd.erb"
- ☐ Remove a resource: The file named '/etc/motd'
- ☐ Add a resource: The template named '/etc/motd' is created with the source 'motd.erb'
- ☐ Use kitchen to test it and chef-client to locally apply the default recipe.



# Lab: Return Home and Generate the Template



```
$ cd ~
```

```
$ chef generate template cookbooks/workstation motd
```

```
Compiling Cookbooks...
```

```
Recipe: code_generator::cookbook
```

```
  * directory[/home/chef/template] action create
```

```
    - create new directory /home/chef/template
```

```
  * template[/home/chef/template/metadata.rb] action create_if_missing
```

```
    - create new file /home/chef/template/metadata.rb
```

```
    - update content in file /home/chef/template/metadata.rb from none to 000bce  
      (diff output suppressed by config)
```

```
  * template[/home/chef/template/README.md] action create_if_missing
```

```
    - create new file /home/chef/template/README.md
```

# Lab: Copy the Existing Source into the Template

 `~/cookbooks/workstation/templates/default/motd.erb`

Property of ...

IPADDRESS: `{node['ipaddress']}`

HOSTNAME : `{node['hostname']}`

MEMORY : `{node['memory']['total']}`

CPU : `{node['cpu']['0']['mhz']}`

# Lab: Update the motd.erb to Use ERB

 `~/cookbooks/workstation/templates/default/motd.erb`

Property of ...

```
IPADDRESS: <%= node['ipaddress'] %>
```

```
HOSTNAME  : <%= node['hostname'] %>
```

```
MEMORY    : <%= node['memory']['total'] %>
```

```
CPU       : <%= node['cpu']['0']['mhz'] %>
```

# Lab: Remove the file Resource

```
~/cookbooks/workstation/recipes/setup.rb
```

```
# ... PACKAGE RESOURCES ...

file '/etc/motd' do
  content "Property of ..."

  IPADDRESS: #{node['ipaddress']}
  HOSTNAME  : #{node['hostname']}
  MEMORY    : #{node['memory']['total']}
  CPU       : #{node['cpu']['0']['mhz']}
"
  mode '0644'
  owner 'root'
  group 'root'
end
```

# Lab: Replace it with the Template Resource

 `~/cookbooks/workstation/recipes/setup.rb`

```
# ... PACKAGE RESOURCES ...
```

```
template '/etc/motd' do  
  source 'motd.erb'
```

```
  mode '0644'
```

```
  owner 'root'
```

```
  group 'root'
```

```
end
```

# Lab: Test the Cookbook



```
$ cd ~/cookbooks/workstation  
$ kitchen test
```

```
-----> Starting Kitchen (v1.4.0)  
-----> Cleaning up any prior instances of <default-centos-67>  
-----> Destroying <default-centos-67>...  
        Finished destroying <default-centos-67> (0m0.00s) .  
-----> Testing <default-centos-67>  
-----> Creating <default-centos-67>...  
        Sending build context to Docker daemon  2.56 kB  
        Sending build context to Docker daemon  
        Step 0 : FROM centos:centos6  
        ---> 72703a0520b7
```

# Lab: Change Directories and Apply the Cookbook



```
$ cd ~
```

```
$ sudo chef-client --local-mode -r "recipe[workstation]"
```

```
[2016-09-16T14:18:05+00:00] WARN: No config file found or specified on command line, using command line options.
```

```
Starting Chef Client, version 12.3.0
```

```
resolving cookbooks for run list: ["apache"]
```

```
Synchronizing Cookbooks:
```

```
- apache
```

```
Compiling Cookbooks...
```

```
[2016-09-16T14:18:09+00:00] WARN: Cloning resource attributes for service[httpd] from prior resource (CHEF-3694)
```

```
[2016-09-16T14:18:09+00:00] WARN: Previous service[httpd]: /root/.chef/local-mode-cache/cache/cookbooks/apache/recipes/server.rb:8:in `from_file'
```

```
[2016-09-16T14:18:09+00:00] WARN: Current service[httpd]: /root/.chef/local-mode-cache/ ...
```



## Lab: Use the Template

For the "workstation" cookbook:

- ✓ Use chef generate to create a template named "motd.erb".
- ✓ Copy the source attribute from the file named '/etc/motd' into the template file "motd.erb"
- ✓ Remove a resource: The file named '/etc/motd'
- ✓ Add a resource: The template named '/etc/motd' is created with the source 'motd.erb'
- ✓ Use kitchen to test it and chef-client to locally apply the default recipe.





## Lab: Update the Version

- ☐ Update the "workstation" cookbook's version for this patch
- ☐ Commit the changes to the "workstation" cookbook to version control

# Lab: Update the Cookbook's Patch Number

```
~/cookbooks/workstation/metadata.rb
```

```
name                'workstation'  
maintainer          'The Authors'  
maintainer_email    'you@example.com'  
license             'all_rights'  
description         'Installs/Configures workstation'  
long_description    'Installs/Configures workstation'  
version             '0.2.1'
```

## Lab: Commit the Changes

```
$ cd ~/cookbooks/workstation  
$ git add .  
$ git status  
$ git commit -m "Update default recipe to use  
template"
```





## Lab: Update the Version

- ✓ Update the "workstation" cookbook's version for this patch
- ✓ Commit the changes to the "workstation" cookbook to version control

# DISCUSSION

## Discussion



What is the benefit of using a template over defining the content within a recipe? What are the drawbacks?

What do each of the ERB tags accomplish?



**CHEF**™

---