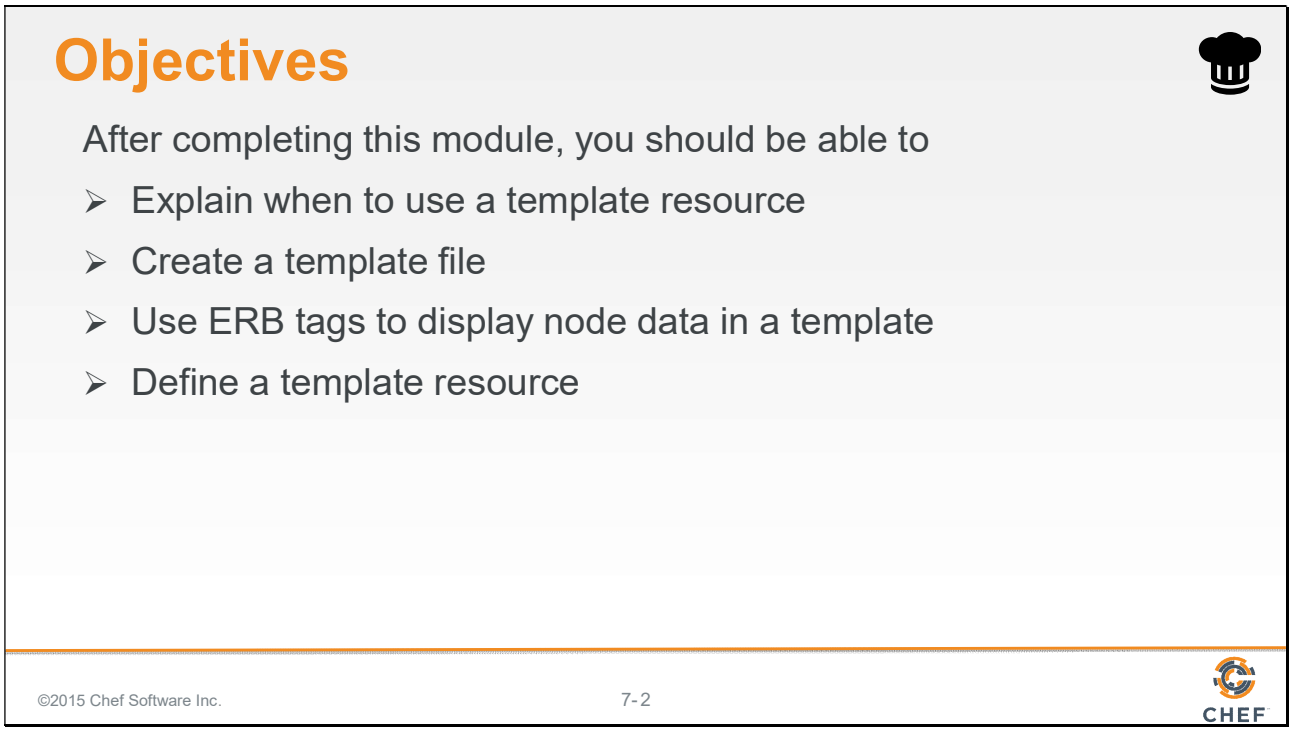
7: Desired State and Data

Slide 2



In this module you will learn how to understand when to use a template resource, create a template file, use ERB tags to display node data in a template, define a template resource.

Slide 3

Cleaner Recipes

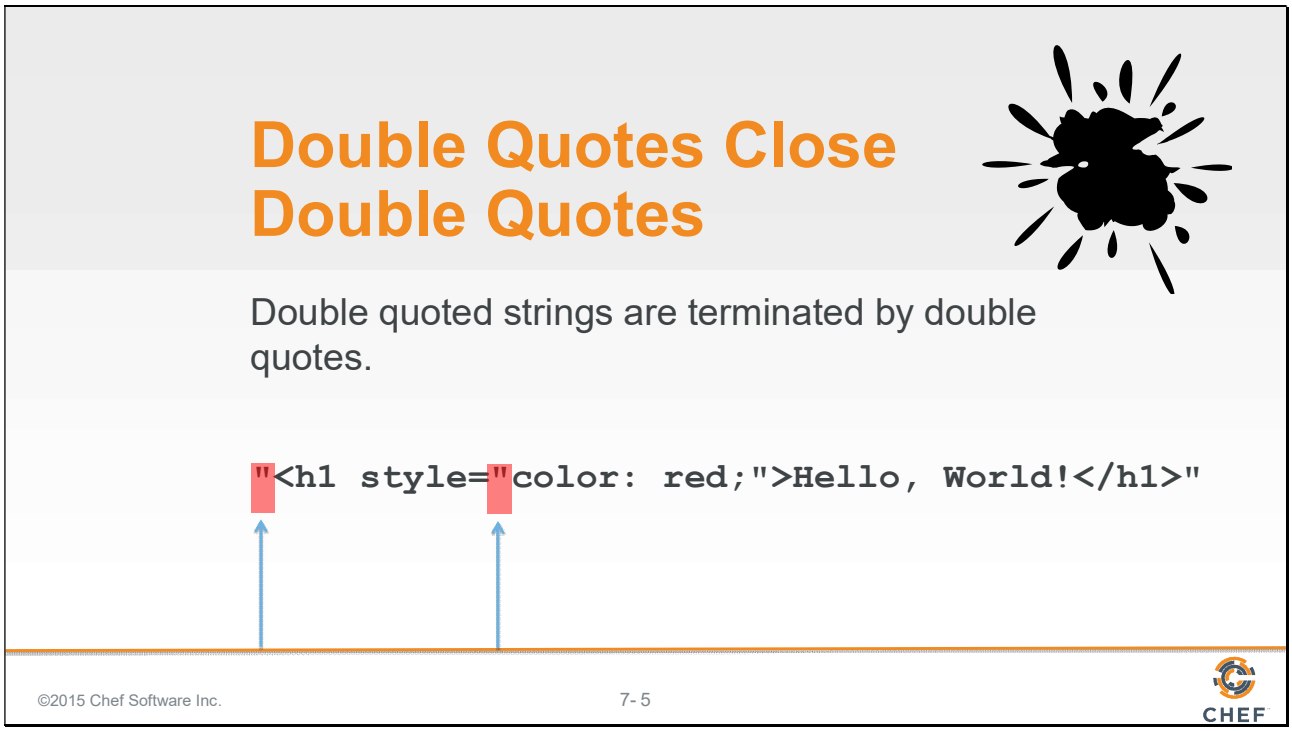
In the last section we updated our two cookbooks to display information about our node.

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

Slide 4

What if new changes are given to us for the website splash page? For each new addition we would need to return to this recipe and carefully paste the contents of the new HTML into the string value of the content attribute.

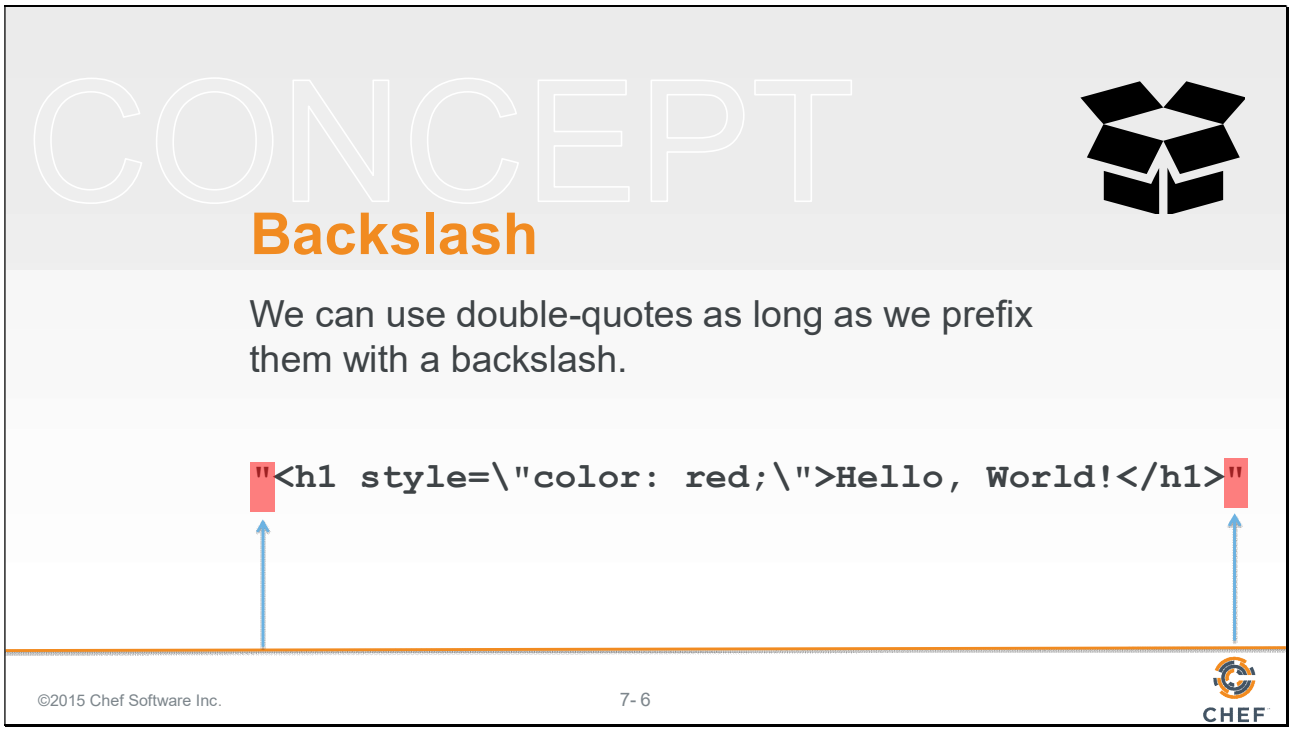
Slide 5



There are some things that you need to be careful of when working with double-quoted strings in Ruby:

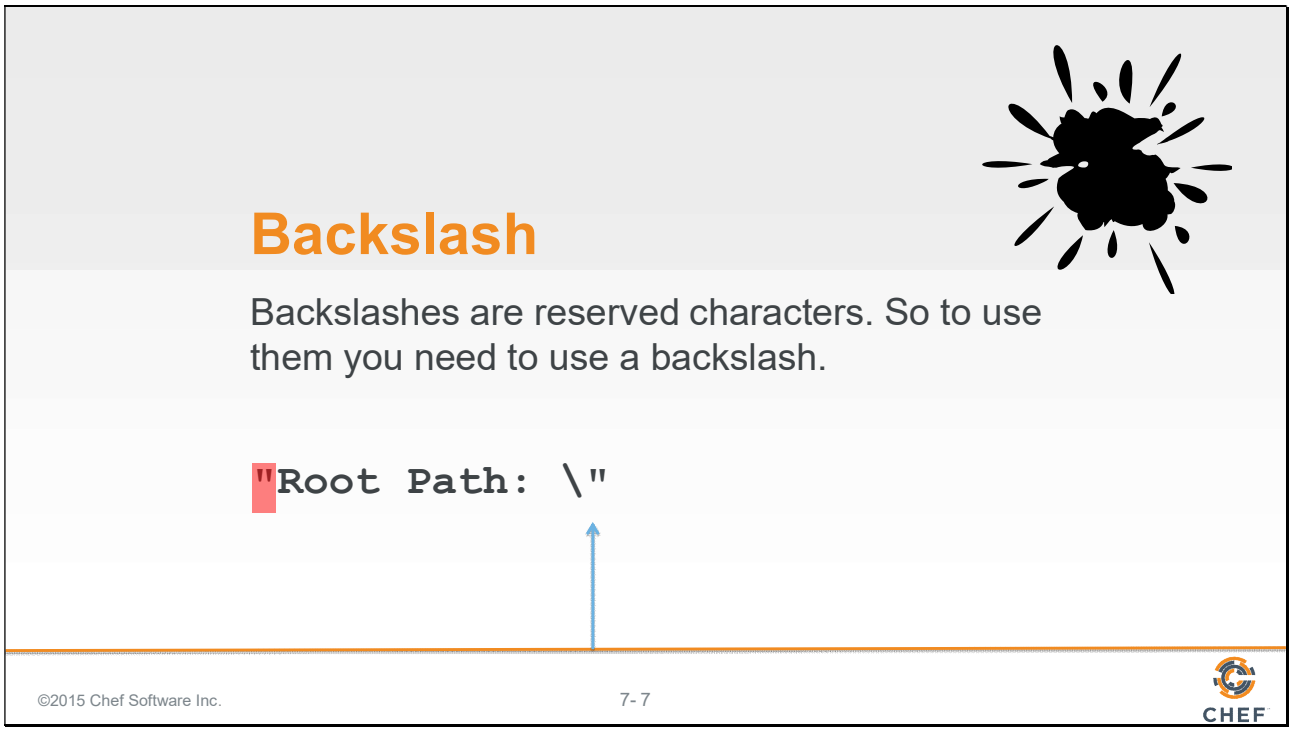
Double-quoted strings are terminated by double-quotes so if any of the text that we paste into this content field has double quotes it is going to have to be escaped.

Slide 6



With Ruby strings you can use the backslash character as an escape character. In this case, if you wanted to have a double-quote inside a double-quoted string, you would need to place a backslash before the double-quote.

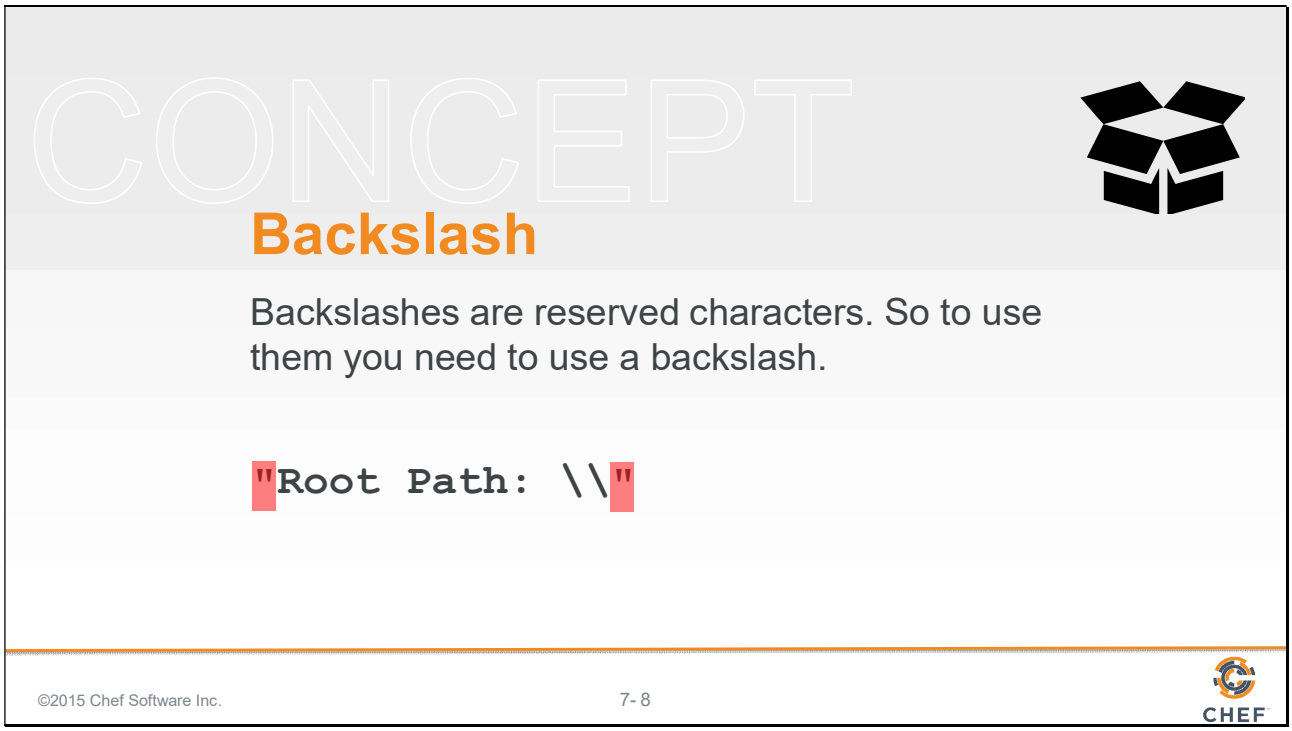
Slide 7



That also brings up an issue with continually-pasting text. You will also need to keep an eye out for backslash characters because backslash characters are now the escape character.

If you want to literally represent a backslash you'll need to use two-backslashes.

Slide 8



So every time text is pasted into the string value of the content attribute, you will need to find and replace all backslashes with double-backslashes and then replace all double-quotes with backslash double-quotes.

Slide 9

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!
```

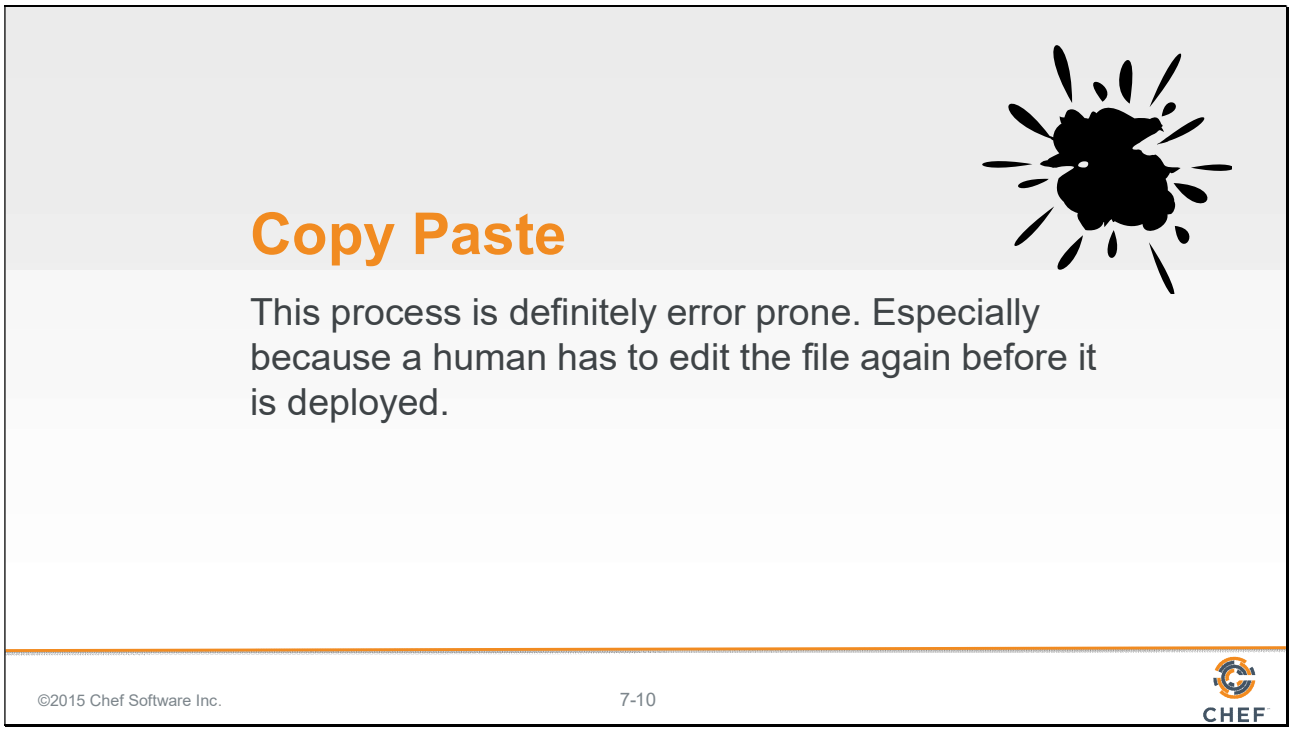

CHEF

It is important to note that the file content may have some important formatting that might be easily overlooked when working with the content in a recipe file.

Besides that, if the size of the string value of the content field grows, it will consume the recipe--making it difficult to understand what is desired state and what is data.

Instructor Note: The Message of the Day file is not a white-space important file. Other configuration files that could be managed with Chef may be white-space important.

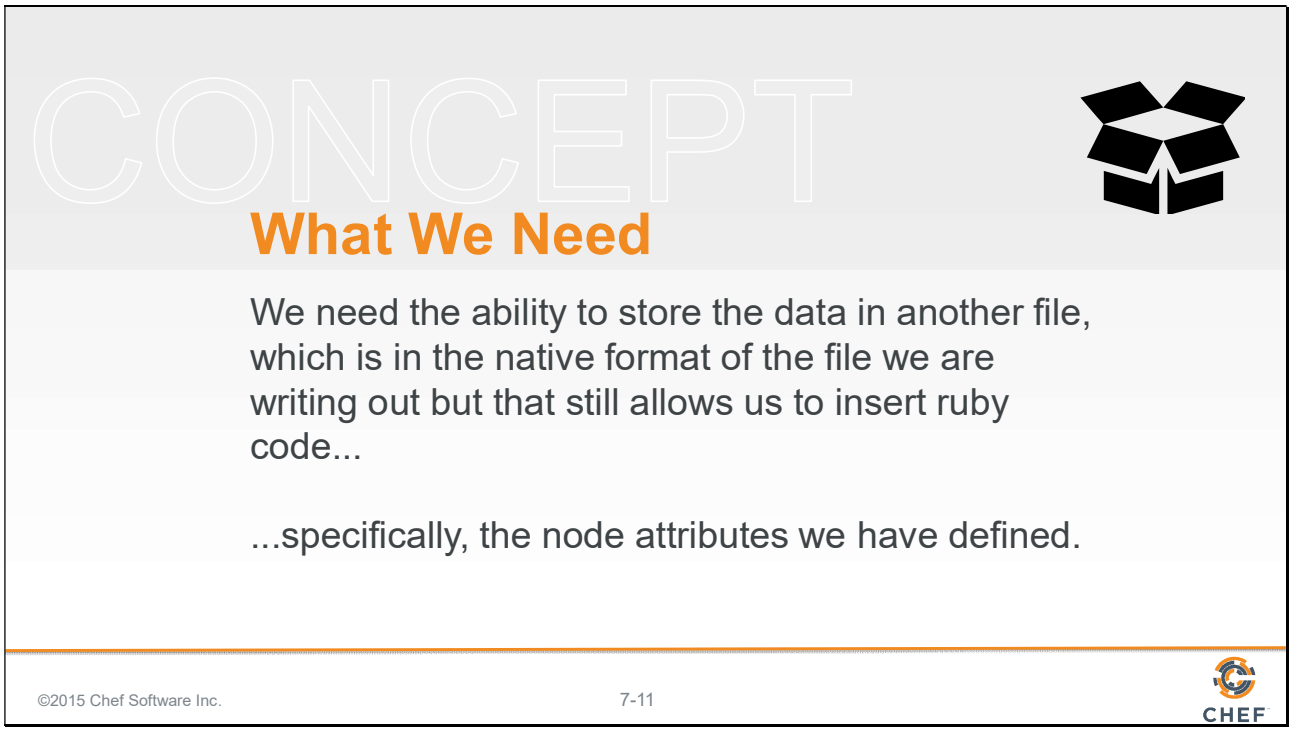
Slide 10



This could sound like a bug waiting to happen.

Any process that requires you to manually copy and paste values and then remember to escape out characters in a particular order, is likely going to lead to issues later when you deploy this recipe to production.

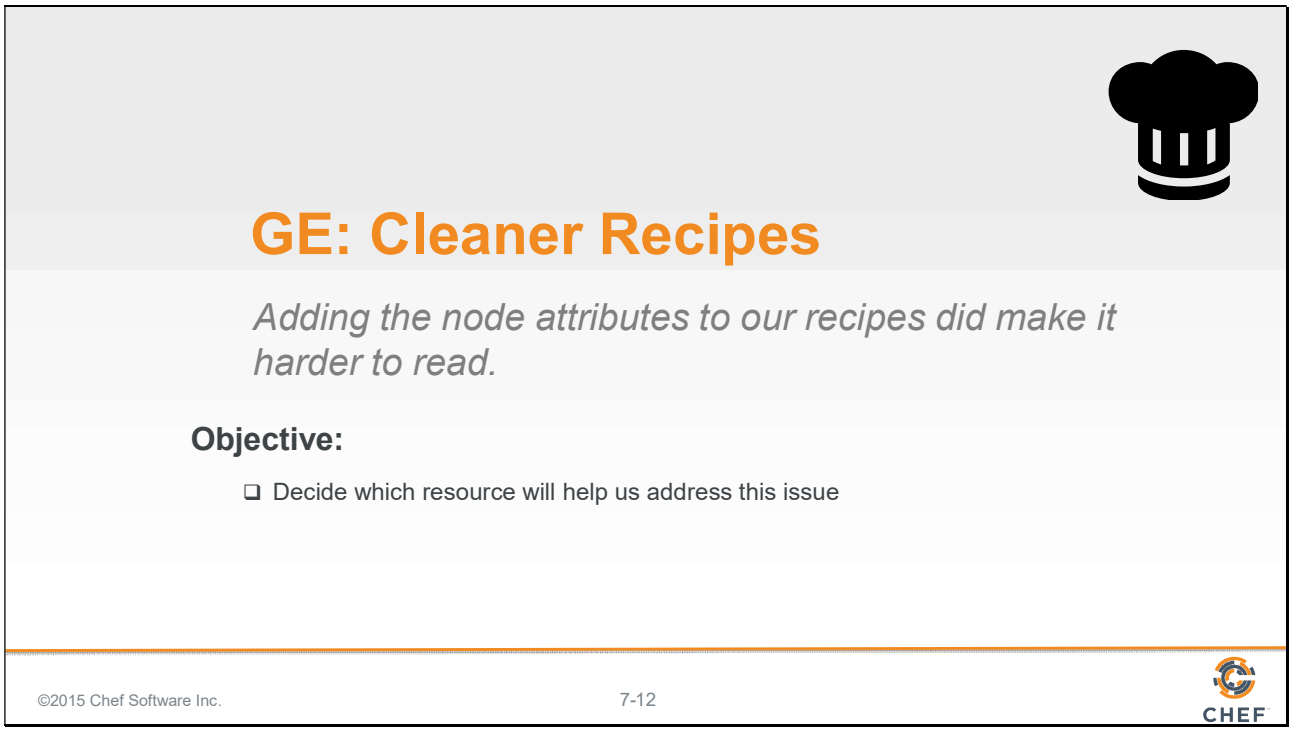
Slide 11



It is better to store this data in another file. The file would be native to whatever format is required so it you wouldn't need to escape any common characters.

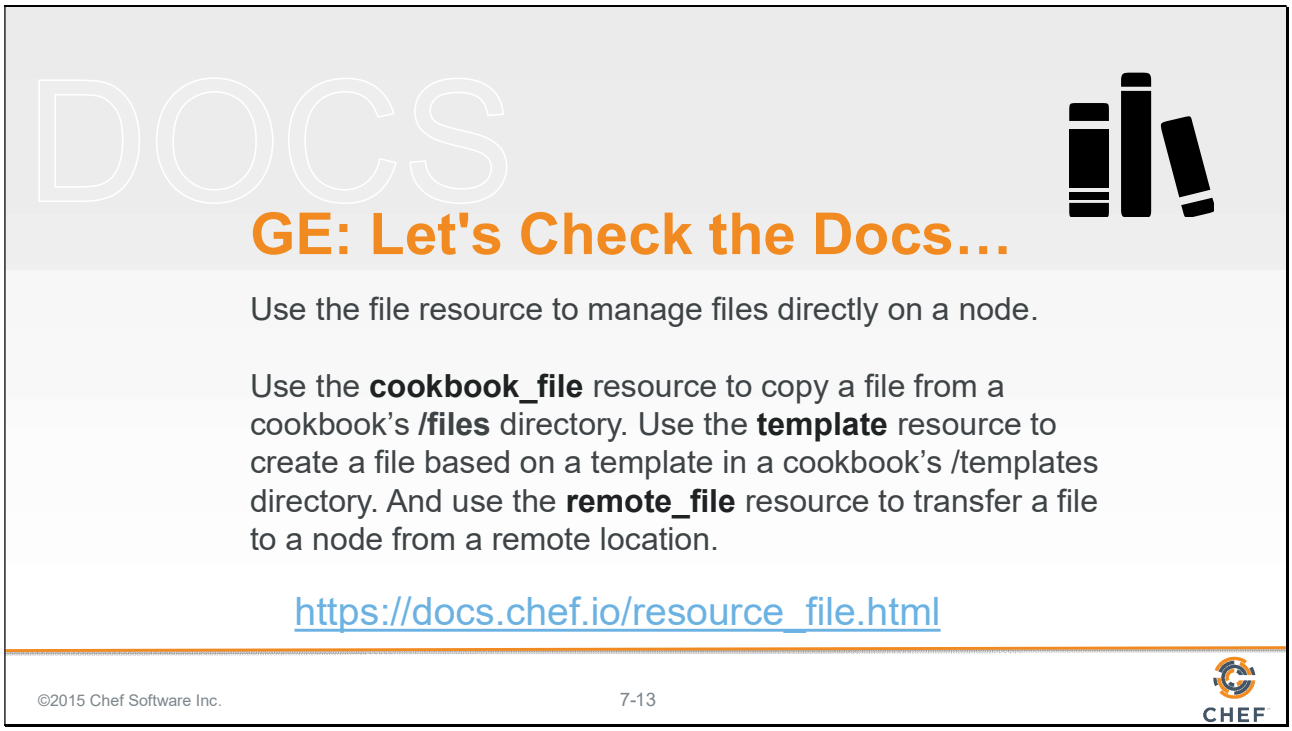
But you still need a way to insert node attributes. So you really need a native file format that allows us to escape out to ruby.

Slide 12



To solve this problem, we need to read up on the file resource more or see if Chef provides alternatives.

Slide 13

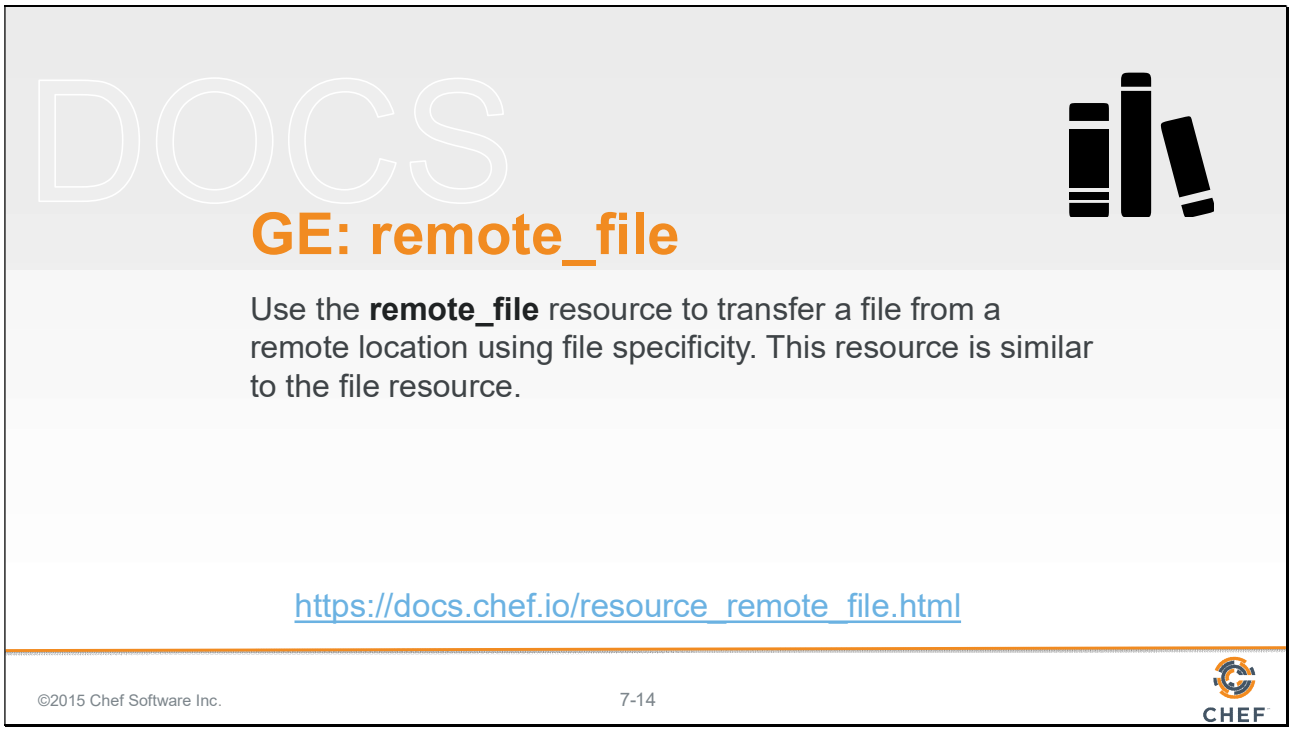


Let's start from what we know--the file resource. Open the documentation and see what it says and see if it gives us a clue to finding alternatives.

The file resource documentation suggests a couple of alternatives to using the file resource: cookbook_file resource; template resource; and remote_file resource.

Let's start with the remote_file resource.

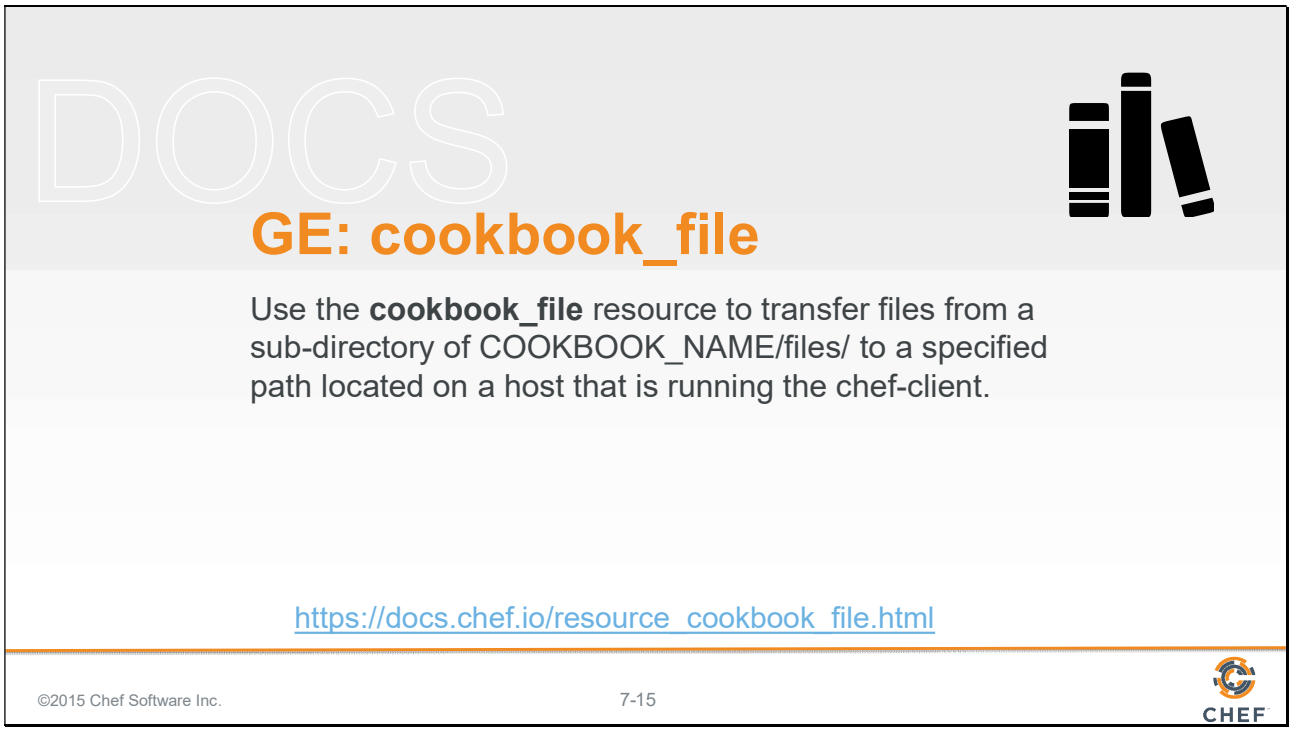
Slide 14



Reading the documentation for remote_file, it seems that remote_file is similar to file. Except remote_file is used to specify a file at a remote location that is copied to a specified file path on the system.

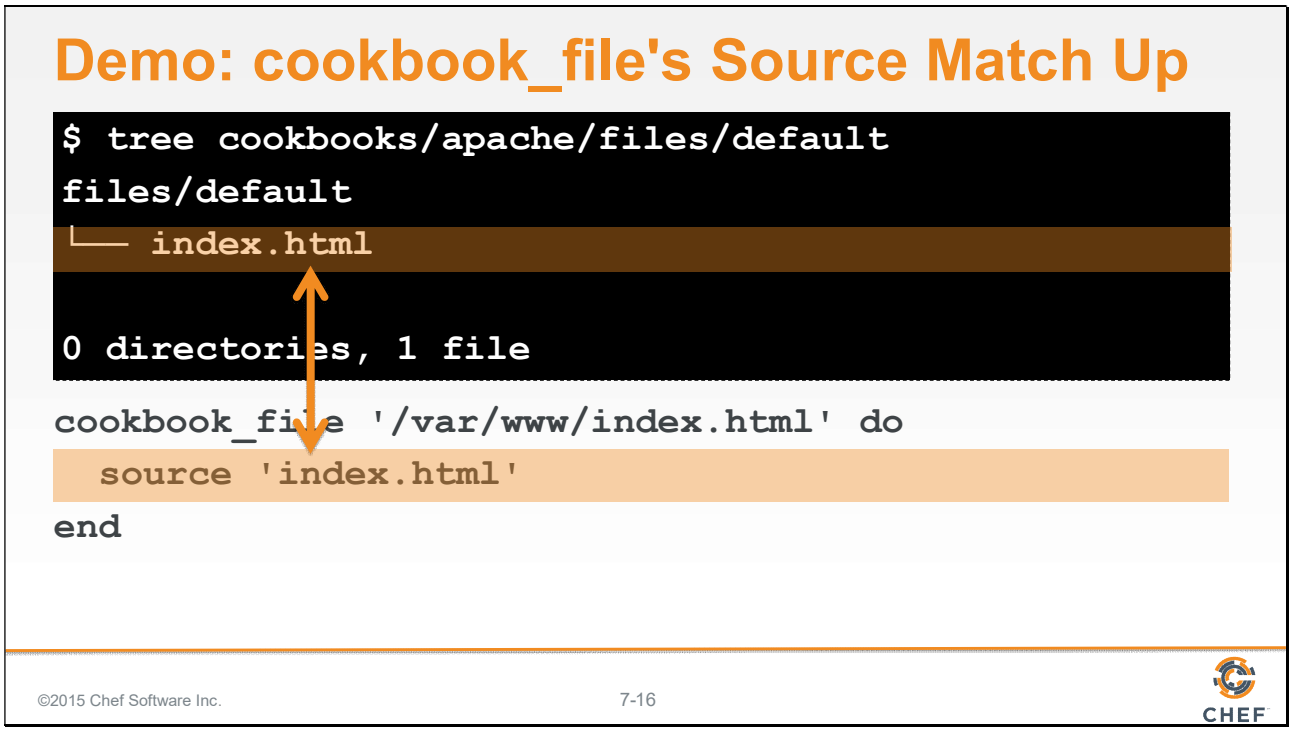
So we could define our index file or message-of-the-day file on a remote system. But that does not allow us to insert attributes about the node we are currently on.

Slide 15



Reading the documentation for cookbook_file, after the boiler-plate resource definition, it sounds as though a cookbook file is capable of...

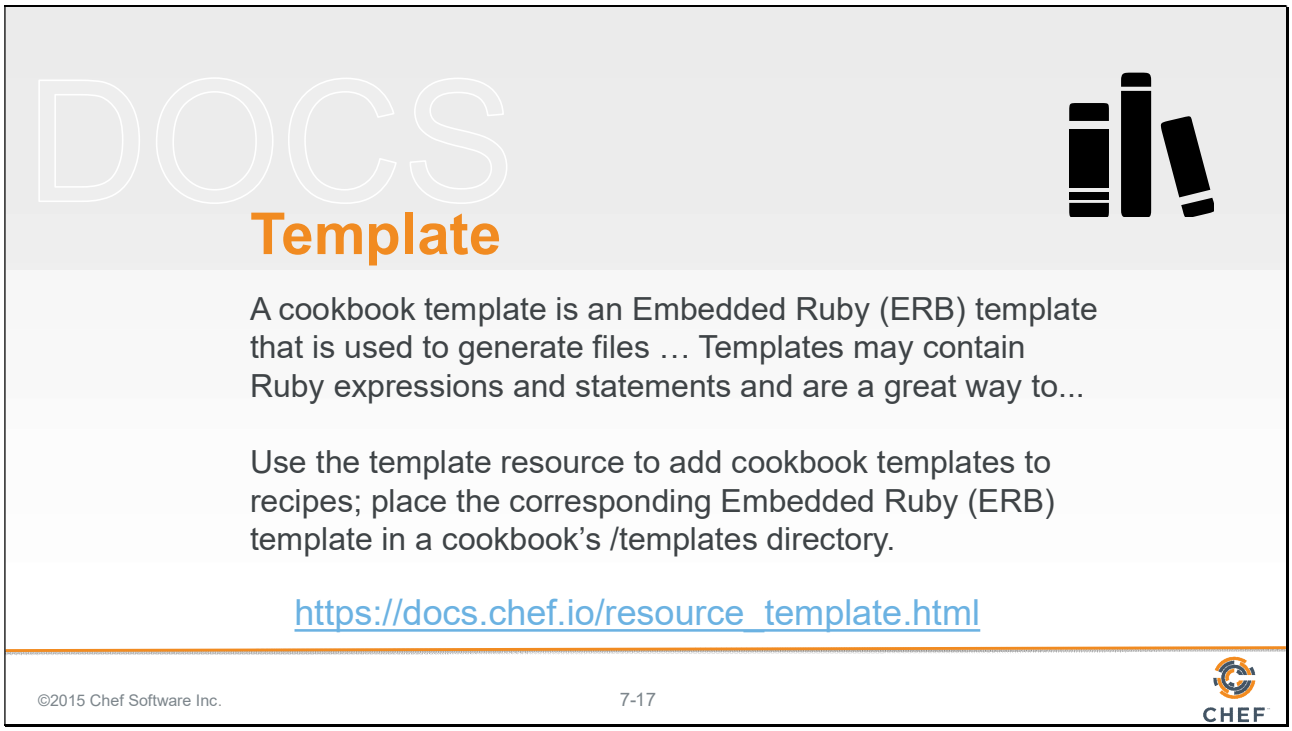
Slide 16



...allowing us to store a file within our cookbook and then have that file transferred to a specified file path on the system.

While it sounds like it allows us to write a file in its native format, it does not sound as though the ability exists to escape out to access the node object and dynamically populate data.

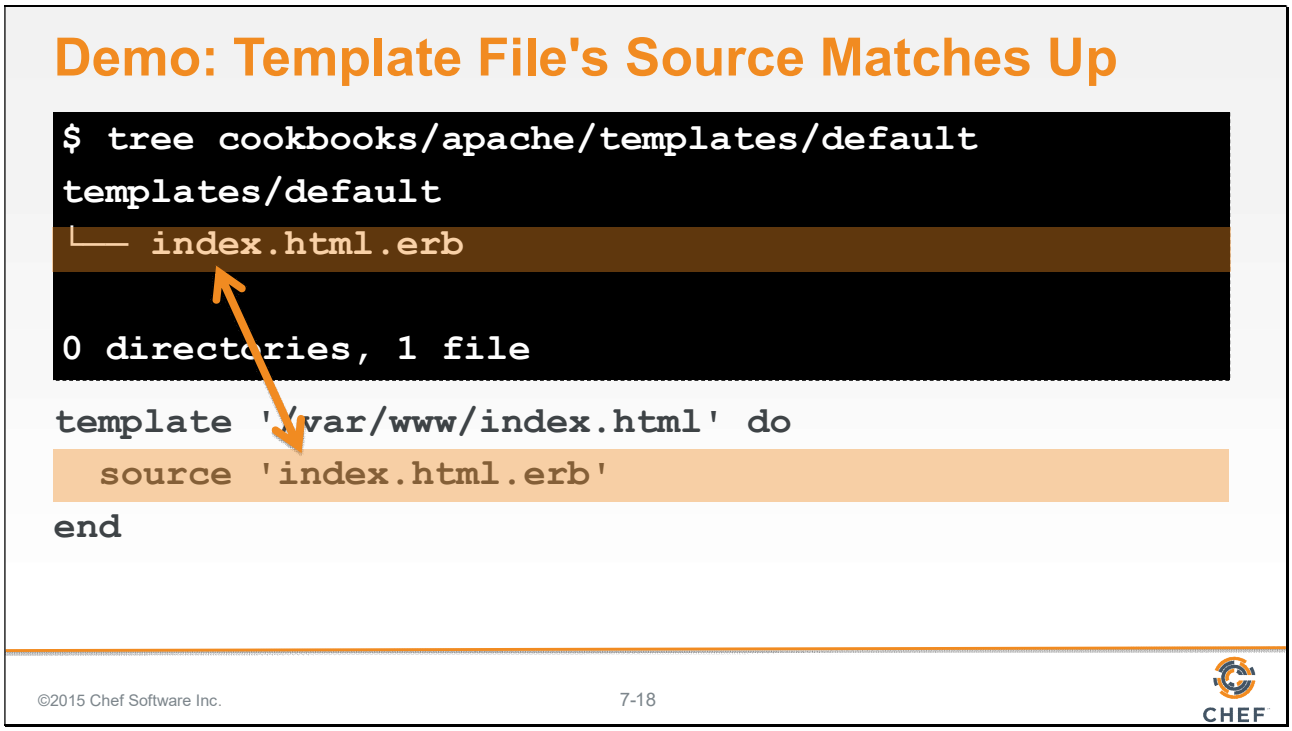
Slide 17



Let's explore templates.

Reviewing the documentation, it seems as though it shares some similarities to the cookbook_file resource.

Slide 18



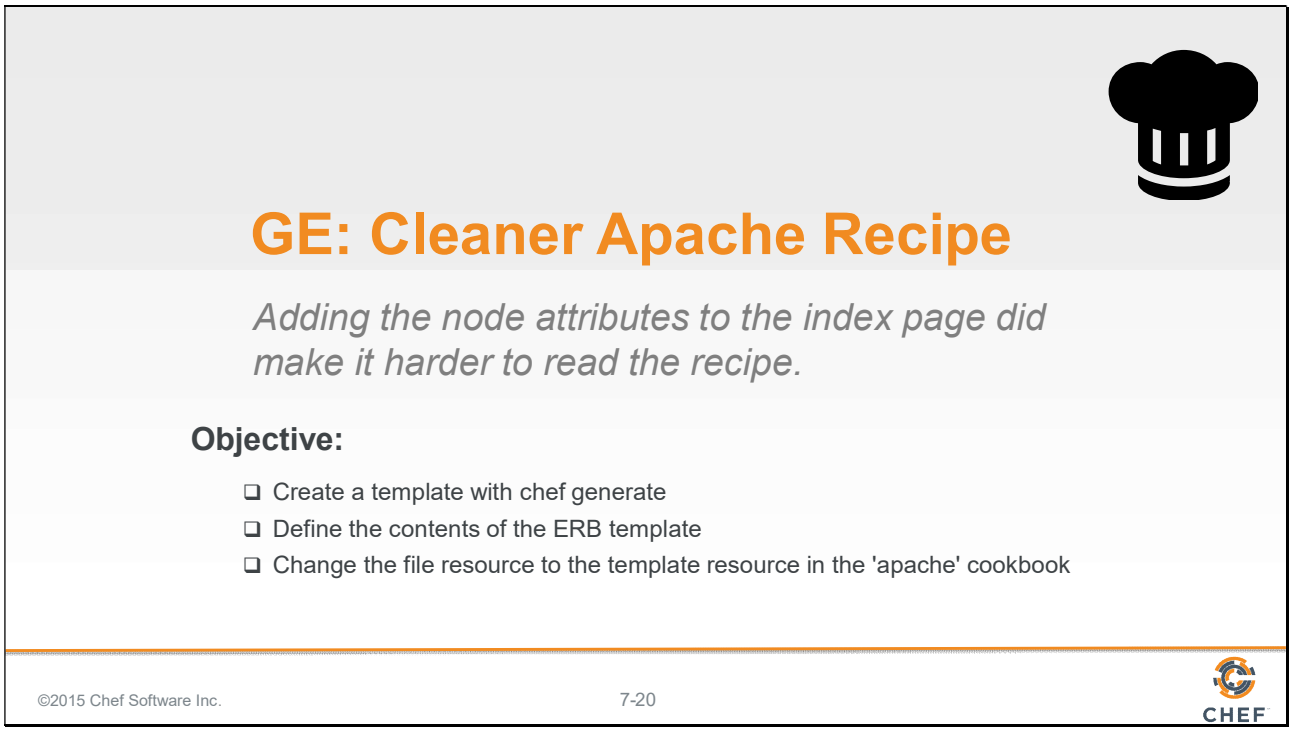
A template can be placed in a particular directory within the cookbook and it will be delivered to a specified file path on the system.

The biggest difference is that it says templates can contain ruby expressions and statements. This sounds like what we wanted: A native file format with the ability to insert information about our node.

Slide 19

And if we look at the bottom section about "Using Templates", we'll see more information about what is required and how we can use them to escape out to execute ruby code.

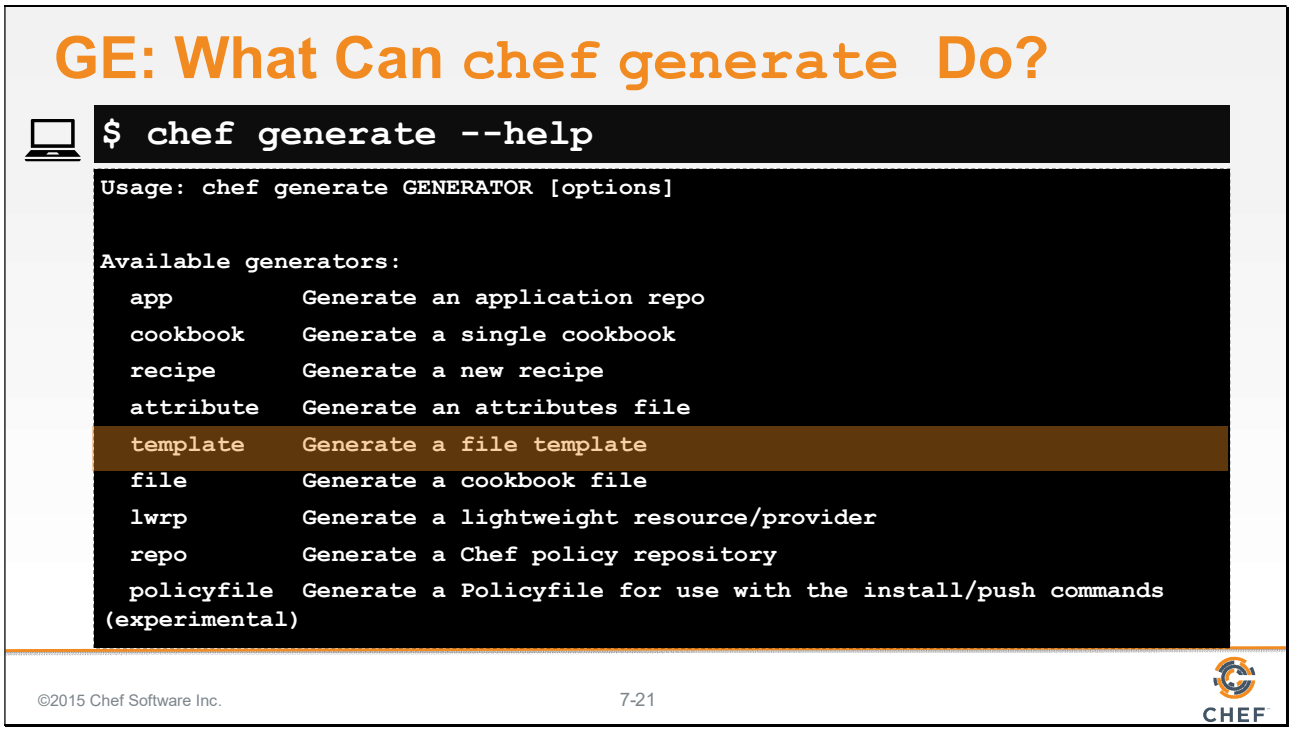
Slide 20



So our objective is clear. We need to use a template resource and create a template and then link them together.

Let's start by creating the actual template file and then we will update the recipe.

Slide 21

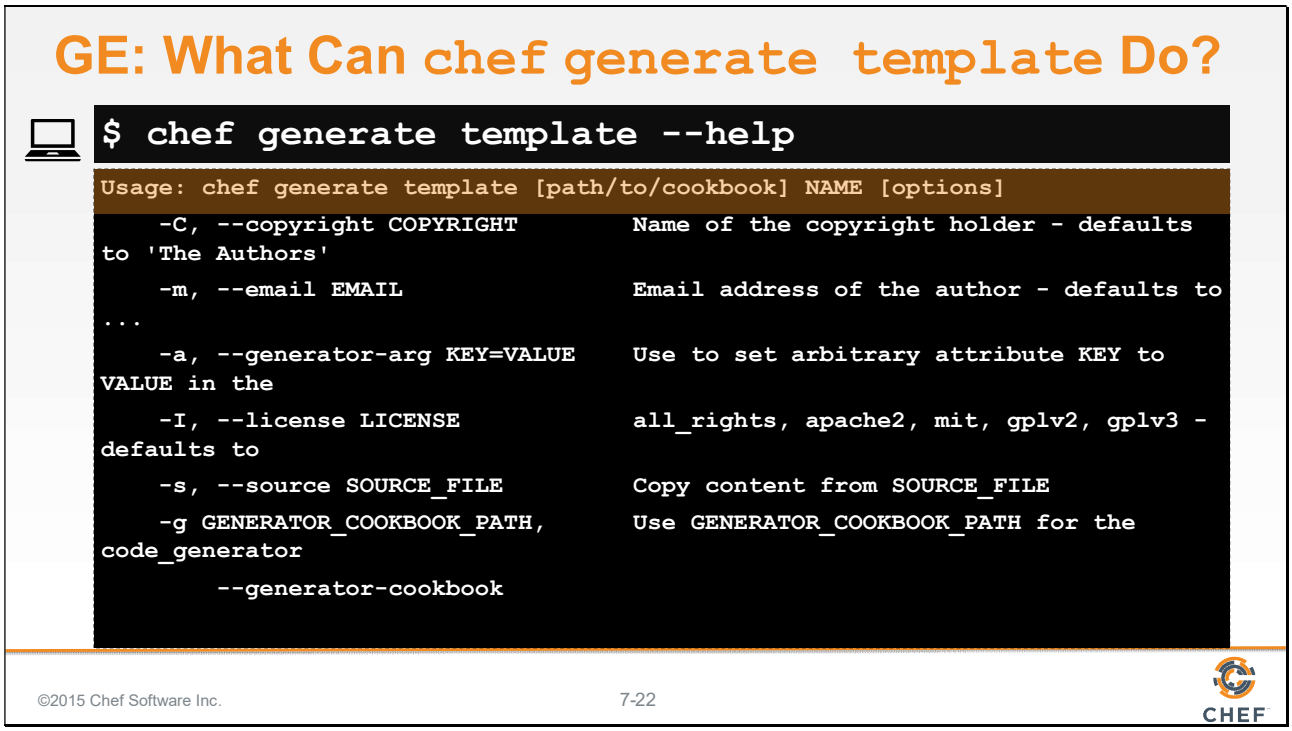


Remember that application Chef--the one that generated our cookbooks. Well it is able to generate cookbook components as well.

Templates and files (for cookbook_files) are a few of the other things it can generate for us.

Let’s use help to review the command again. And let's ask for help about the 'generate' subcommand.

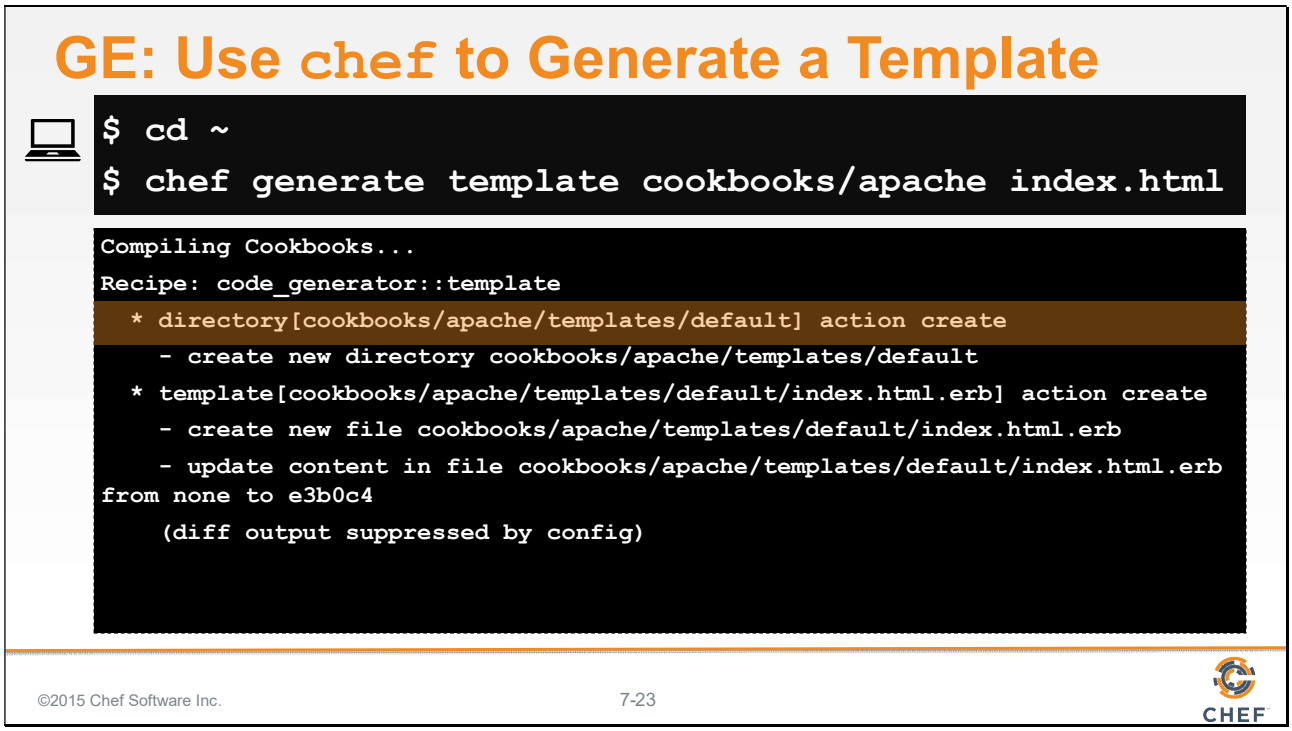
Slide 22



Finally let's ask for help for generating templates.

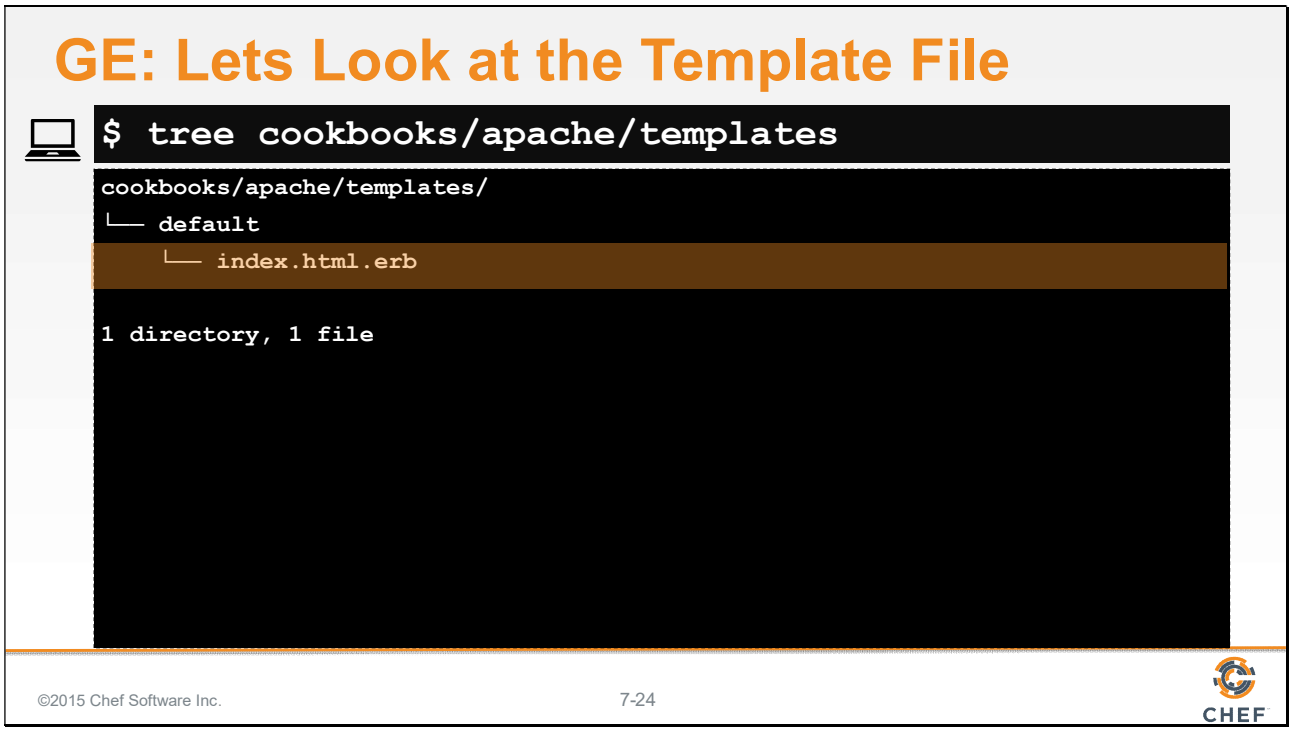
The command requires two parameters--the path to where the cookbook is located and the name of the template to generate. There are some other additional options but these two seem like the most important.

Slide 23



Use '**chef generate template**' to create a template in the apache cookbook found in the cookbooks/apache directory and the file we want to create is named index.html.

Slide 24

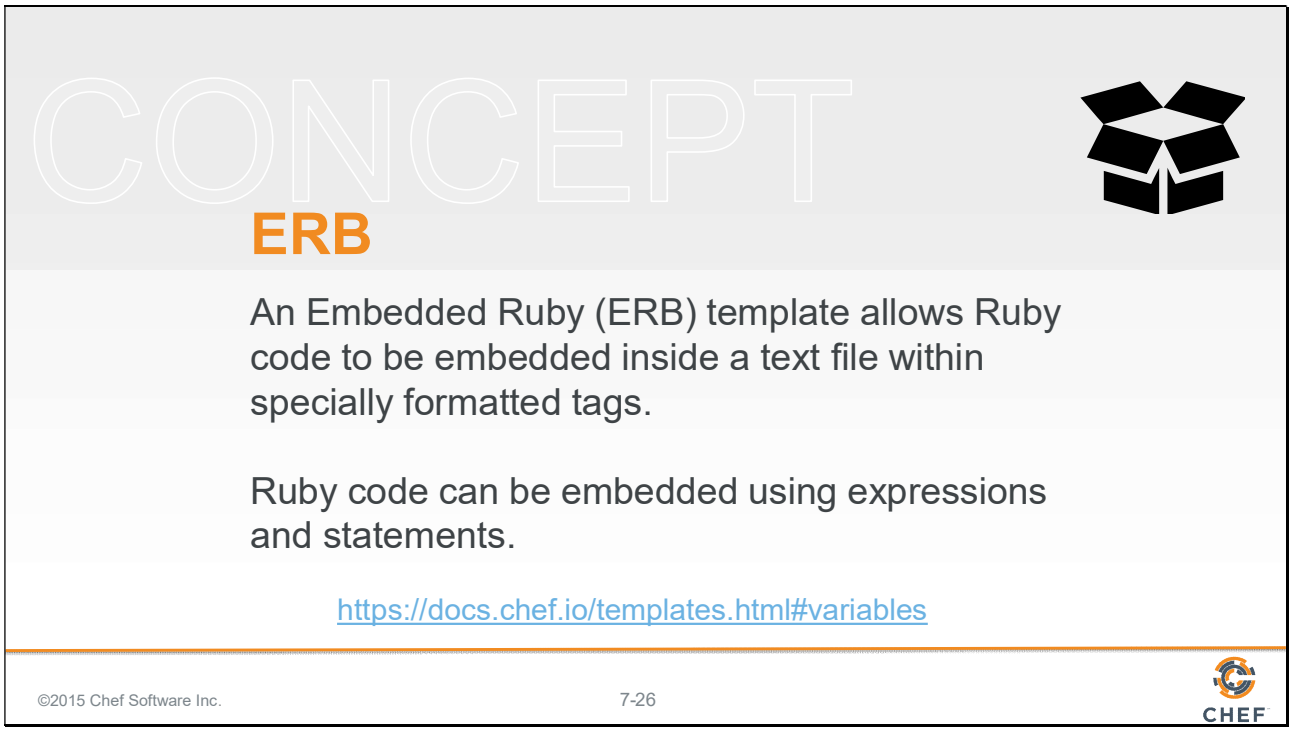


That is the first step. Now that the template exists, we are ready to define the content within the template file.

Slide 25

Now we need to understand what ERB means.

Slide 26



ERB template files are special files because they are the native file format we want to deploy but we are allowed to include special tags to execute ruby code to insert values or logically build the contents.

Slide 27

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.

Here is an example of a text file that has several ERB tags defined in it.

Slide 28

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.

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

Slide 29

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.

The beginning tag is a less-than sign followed by a percent sign. The closing tag is a percent sign followed by a greater-than sign.

Slide 30

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.

These tags are used to execute ruby but the results are not displayed.

Slide 31

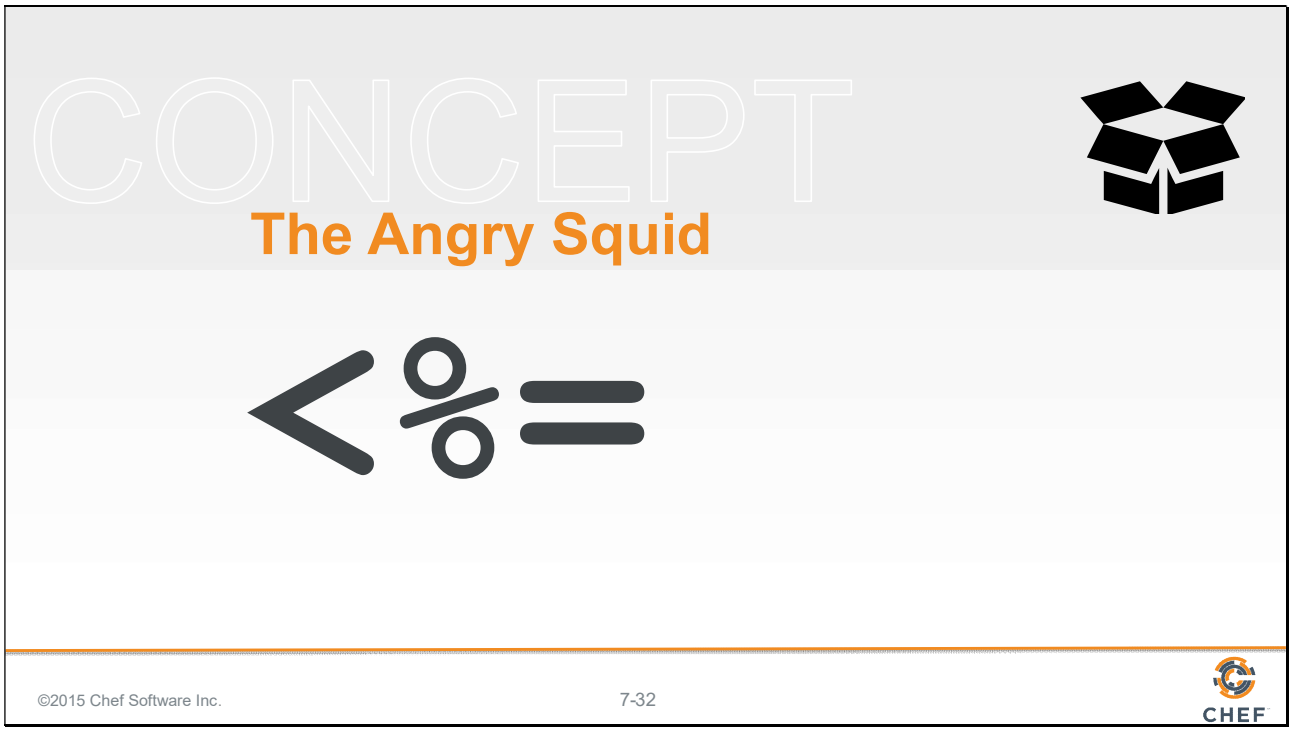
Text Within an ERB Template

```
<% 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.

ERB supports additional tags, one of those is one that allows you to output some variable or some ruby code. Here the example is going to display that 50 plus 50 equals the result of ruby calculating 50 plus 50 and then displaying the result.

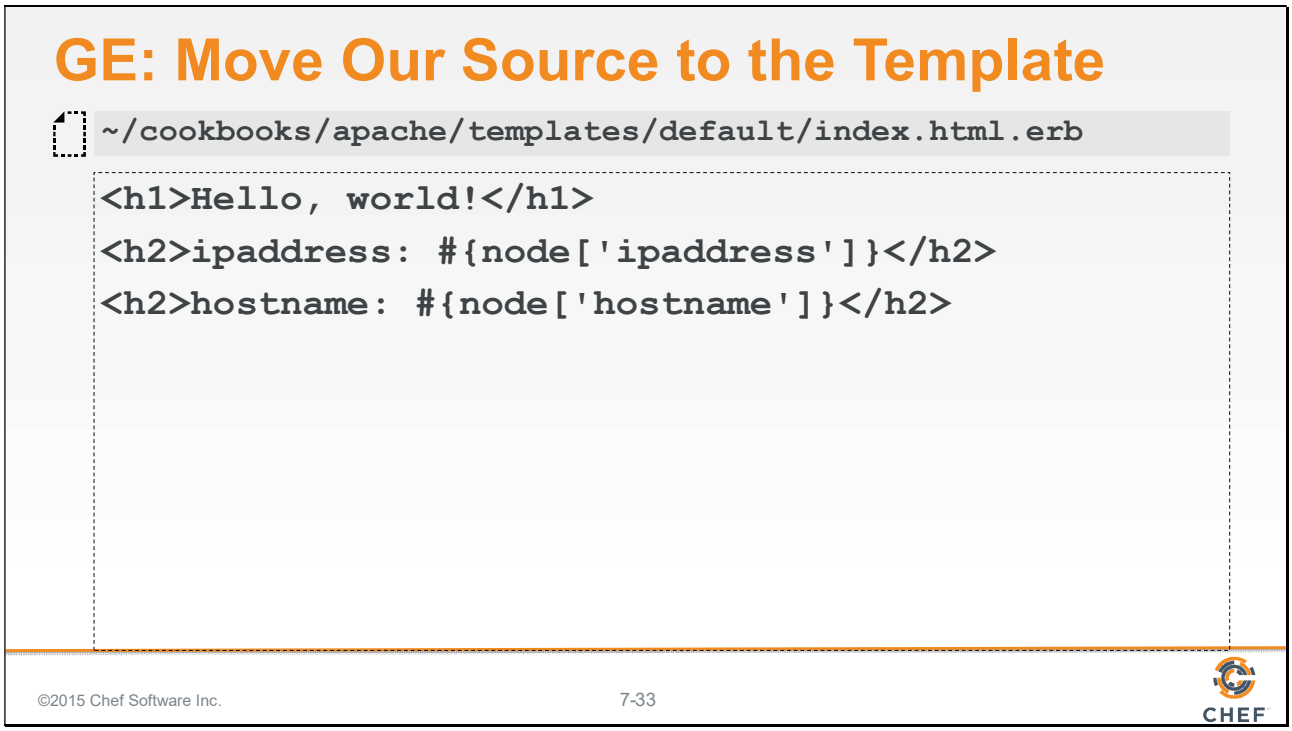
Slide 32



The starting tag is different. It has an equals sign. This means show the value stored in a variable or the result of some calculation.

We often refer to this opening tag that outputs the content as the Angry Squid. The less-than is its head, the percent sign as its eyes, and the equals sign its tentacles shooting away after blasting some ink.

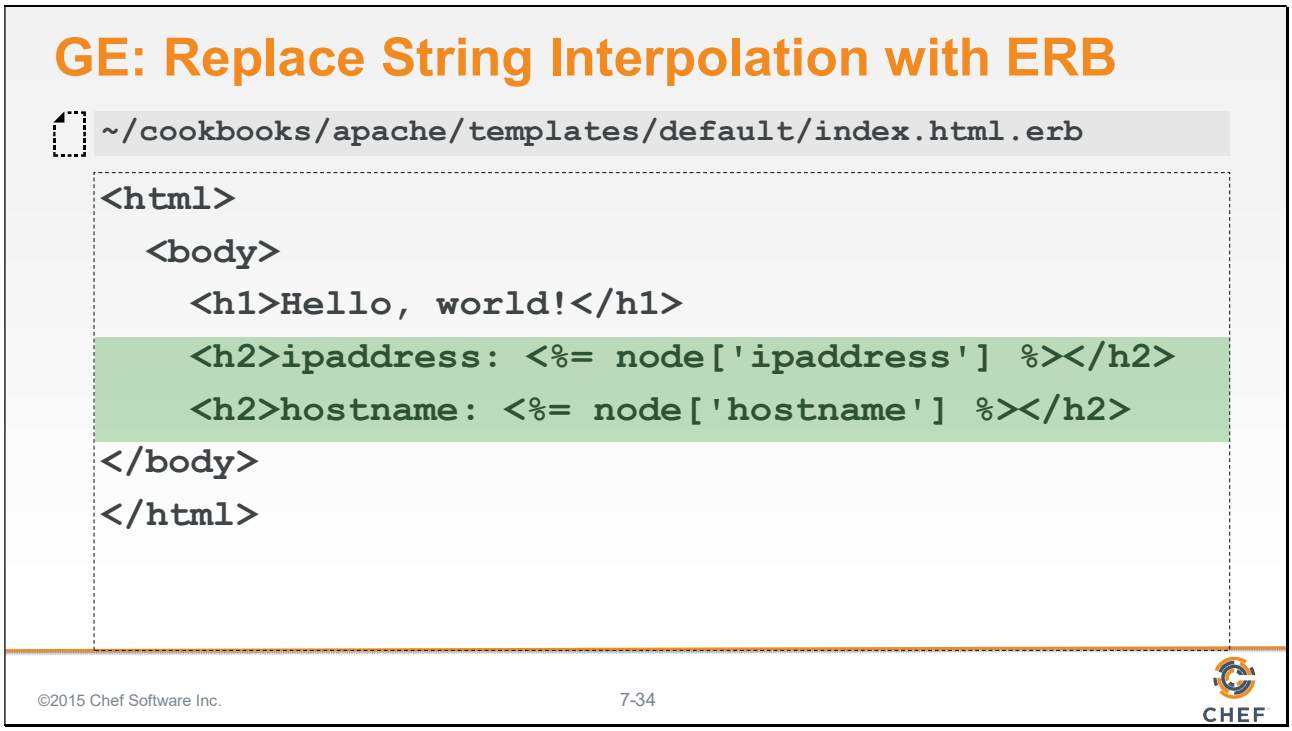
Slide 33



With that in mind let's update the template with the current value of the file resource's content field.

Copying this literally into the file does not work because we no longer have the ability to use string interpolation within this html file. String interpolation only works within a ruby file between a double-quoted String.

Slide 34



We are going to need to change string interpolation sequence with the ERB template syntax. And it seems for this content we want to display the output so we want to make sure that we are using ERB's angry squid opening tag.

Slide 35

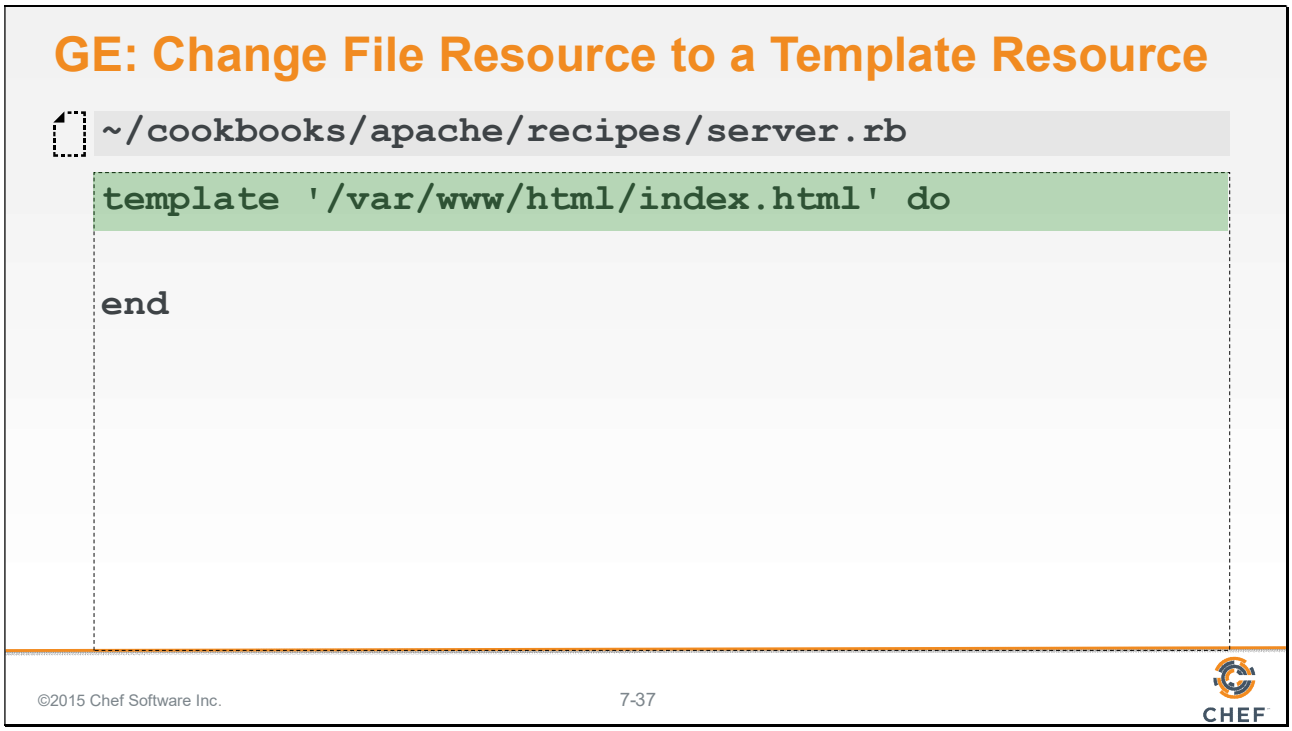
The template is created and the contents are correctly defined. It is time to update the recipe.

Slide 36

Let's open the apache cookbook's recipe named 'server'.

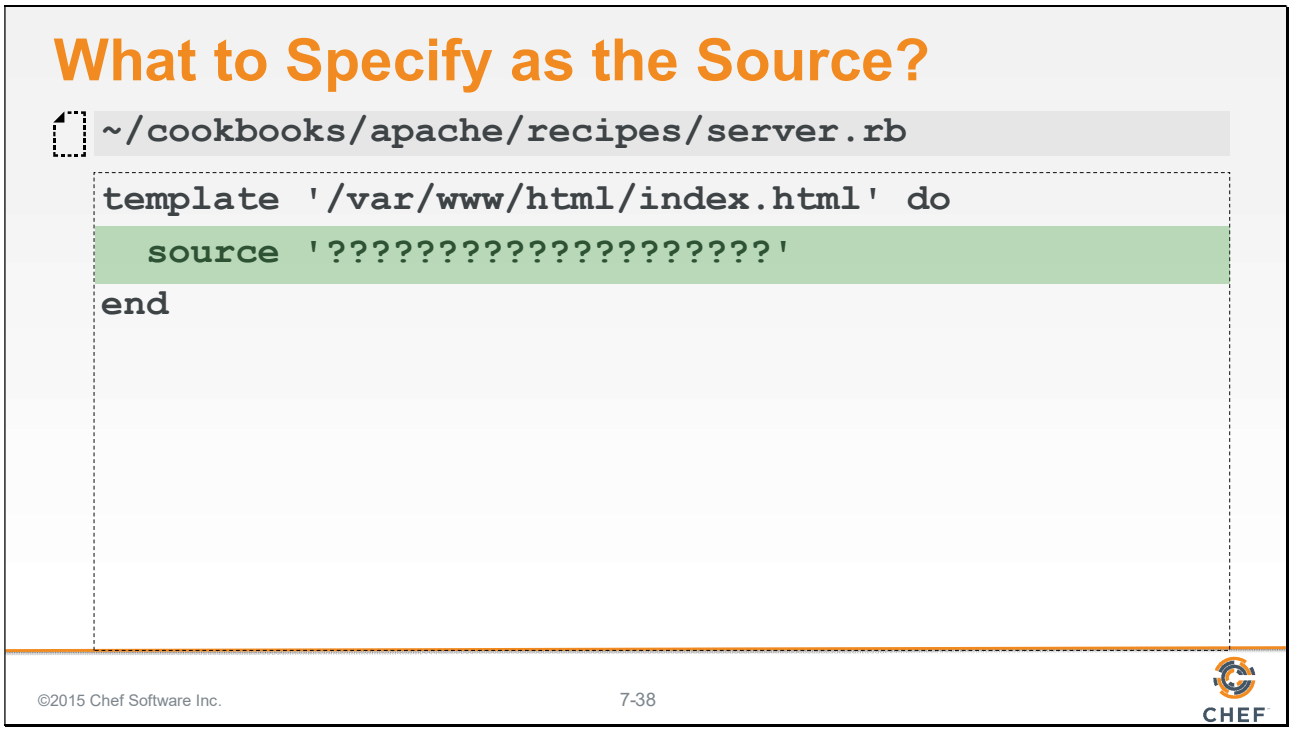
We will want to remove the content attribute from the file resource. Because that content is now in the template. But only if we use a template resource.

Slide 37



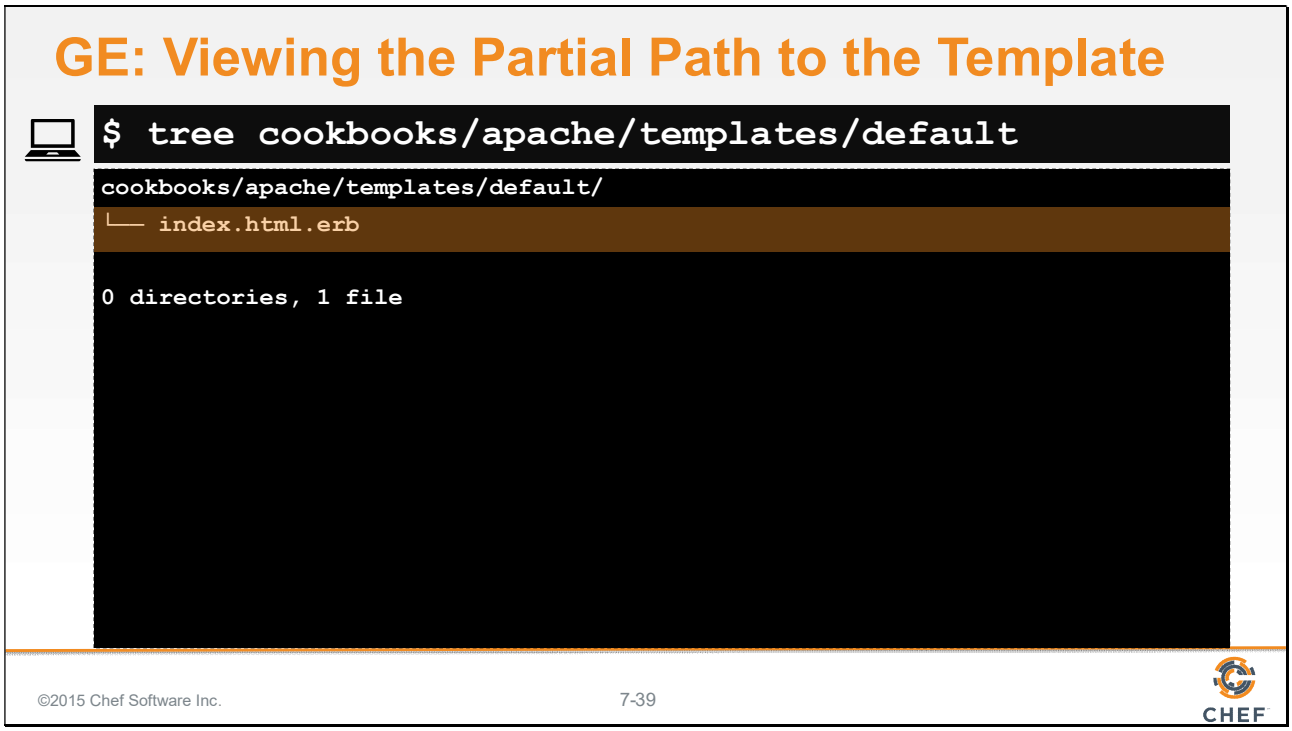
So it's time to change the file resource to a template resource so that it can use the template file that we have defined.

Slide 38



Lastly we need to specify a source attribute which contains that path to the template we generated. This path is relative starting from within the cookbook's template directory.

Slide 39

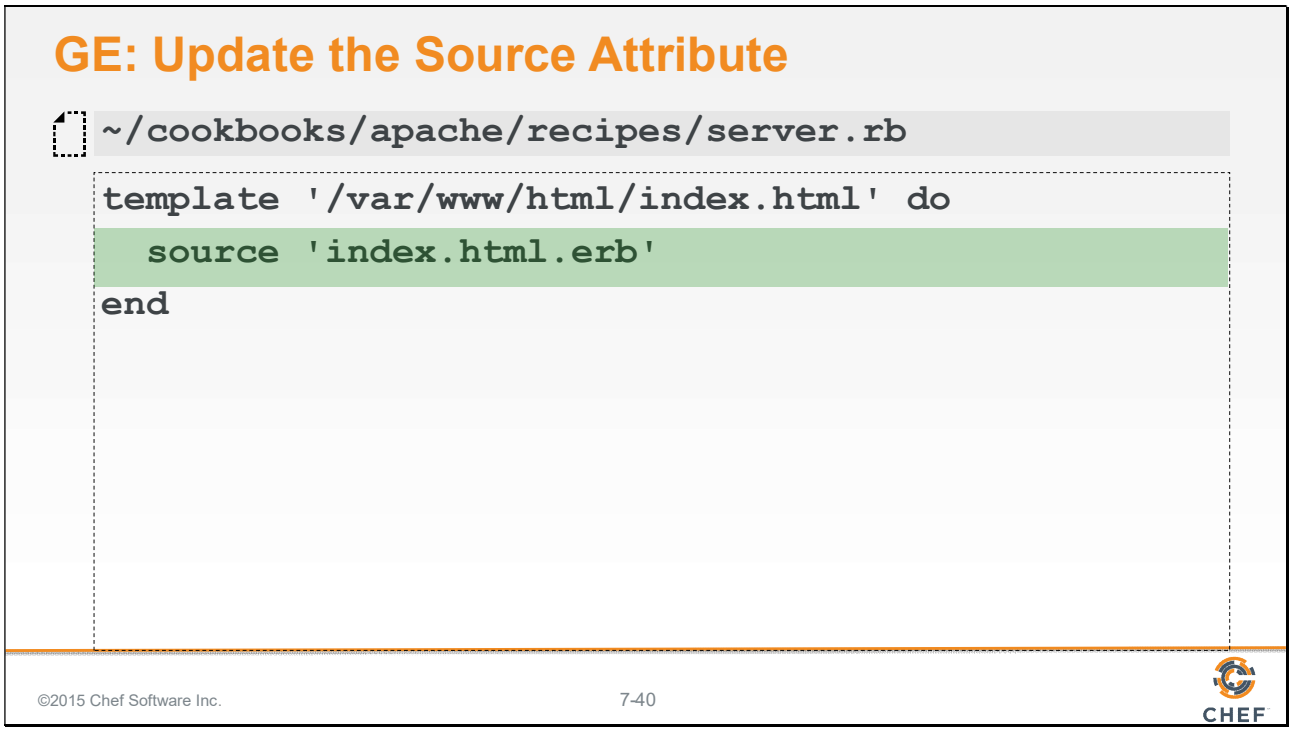


To visualize that with 'tree' we can run it with a path that places us right at the templates directory. So the results will be relative paths from the point specified.

And we see the filepath index.html.erb.

Instructor Note: The default folder denotes that we want to use this file for all platforms.

Slide 40



Now we have the path to our template so we can update the template resource's source attribute value.

Slide 41

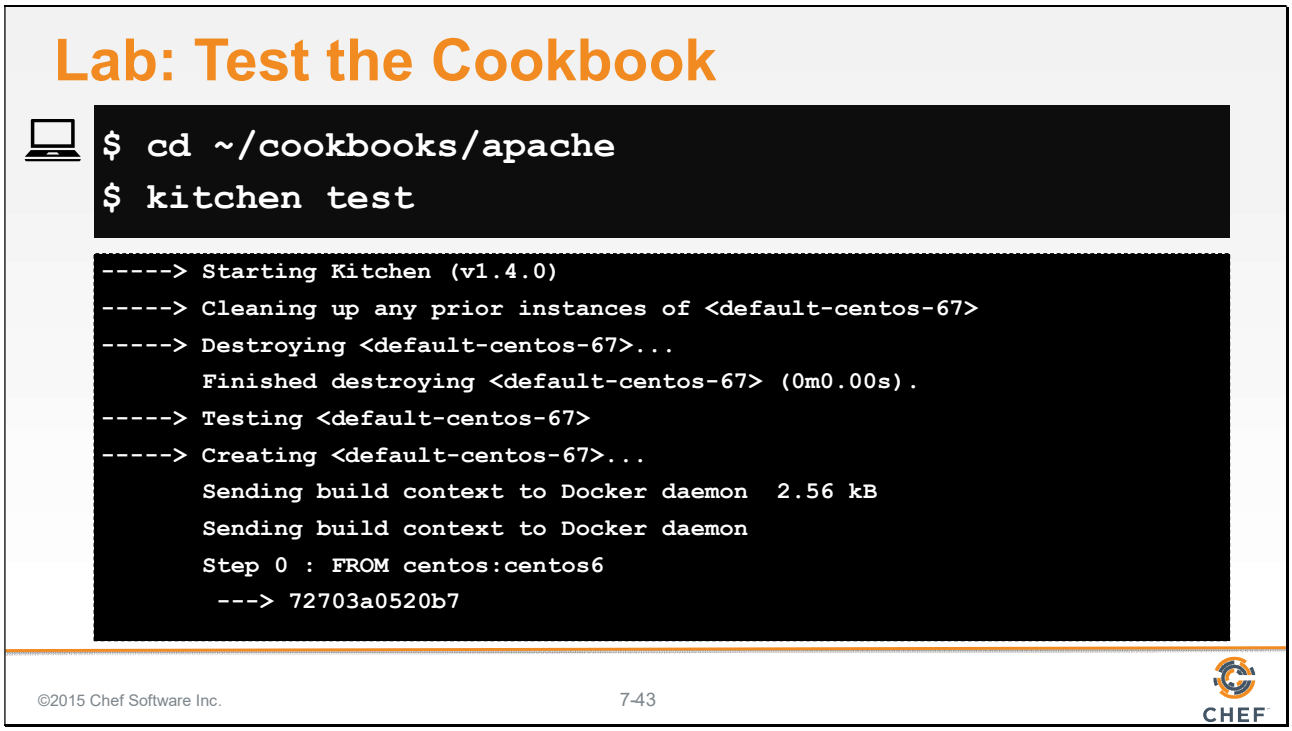
We hopefully haven't changed the original goal of our recipe but we have made some changes.

Slide 42

In this lab, you will use 'kitchen' to verify the cookbook and use 'chef-client' to apply the cookbook. If everything is working then update the patch number and commit the changes to version control.

Instructor Note: Allow 8 minutes to complete this exercise.

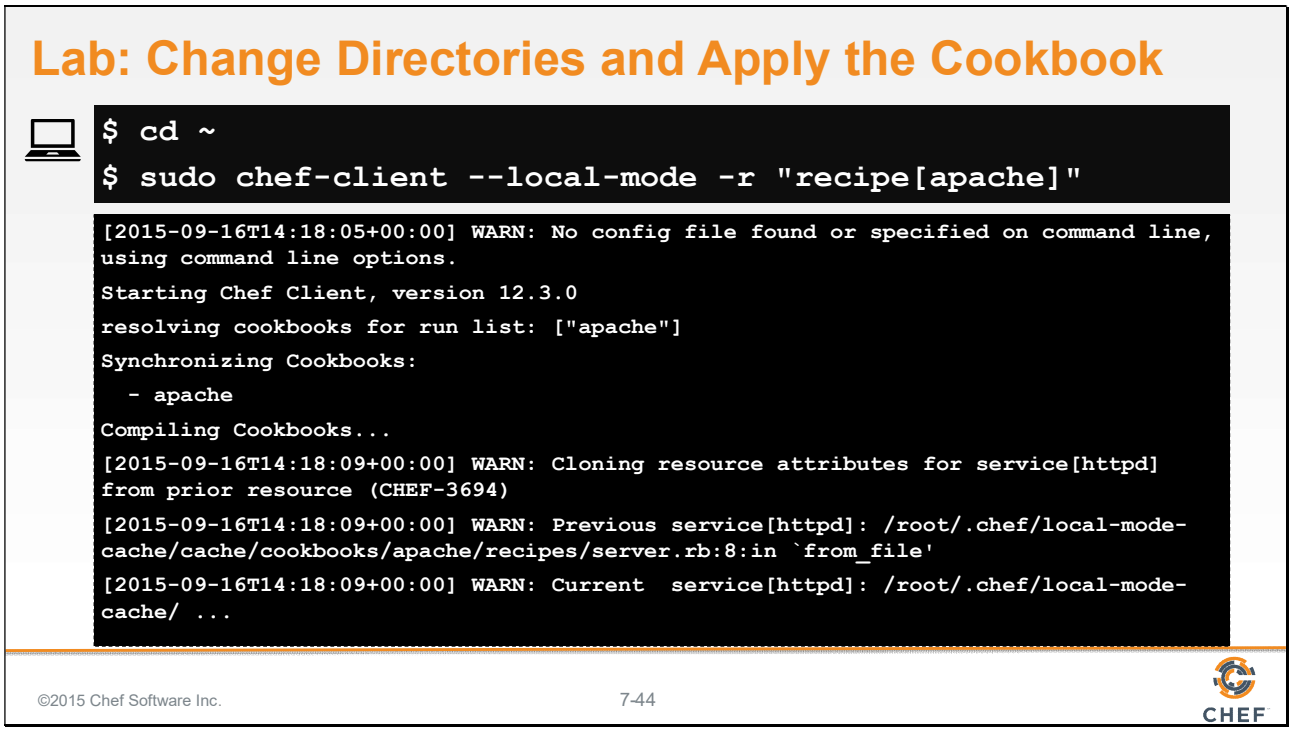
Slide 43



Since kitchen is a cookbook testing tool, you need to move into the cookbook's directory.

Then run the 'kitchen test' command, addressing any issues if they show up.

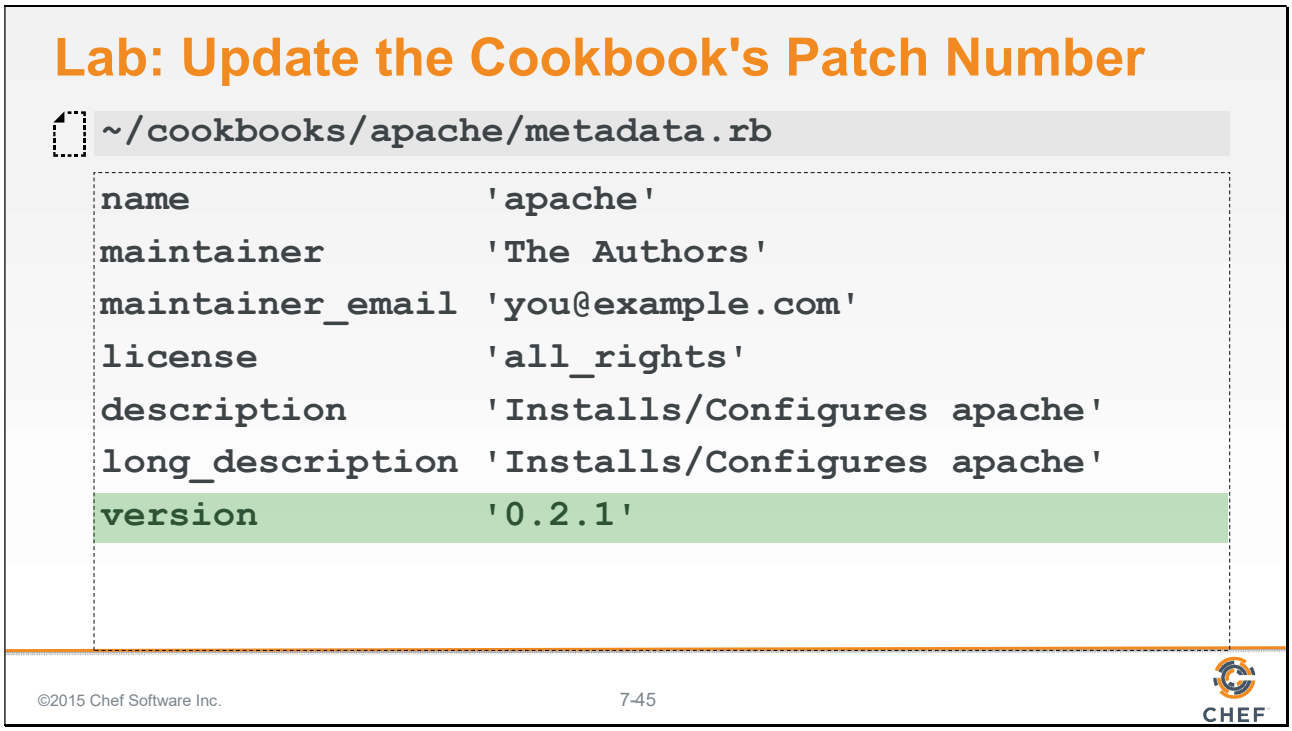
Slide 44



When all the tests pass, return to the home directory, so you can execute 'chef-client'.

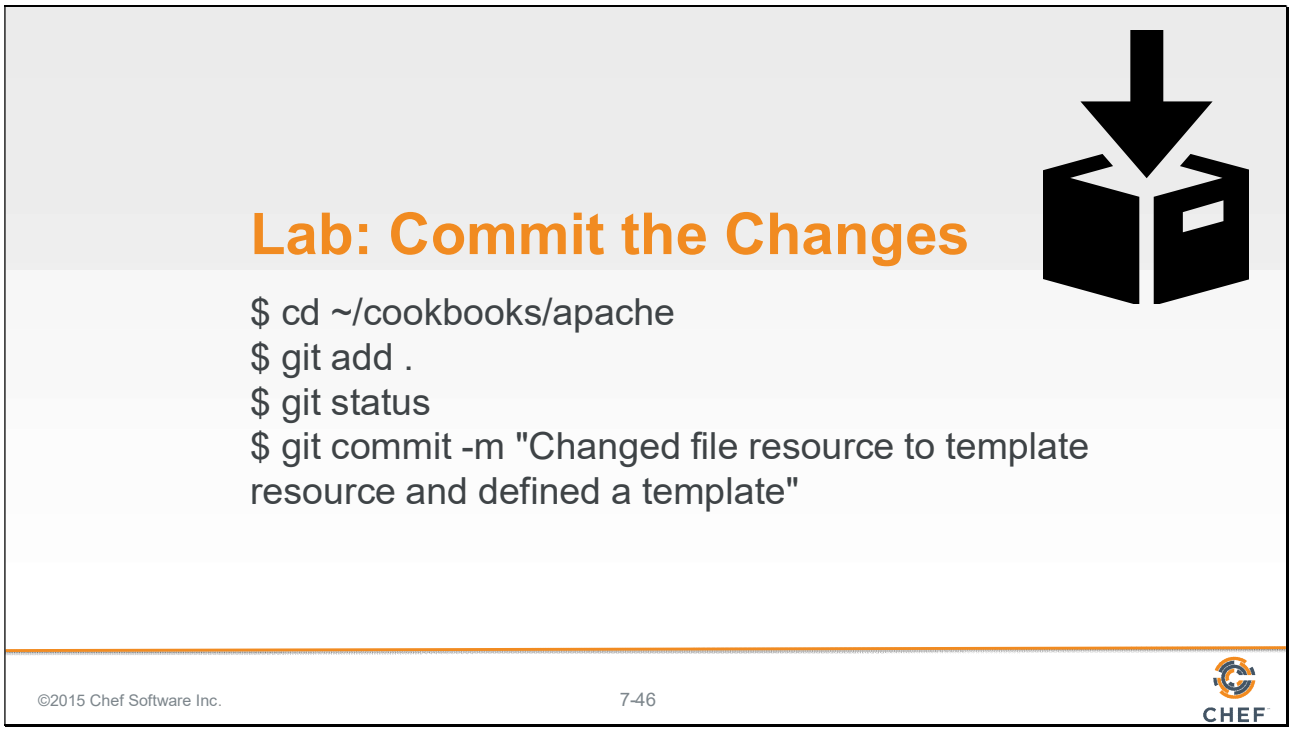
And then apply the apache cookbook's default recipe to the local system.

Slide 45



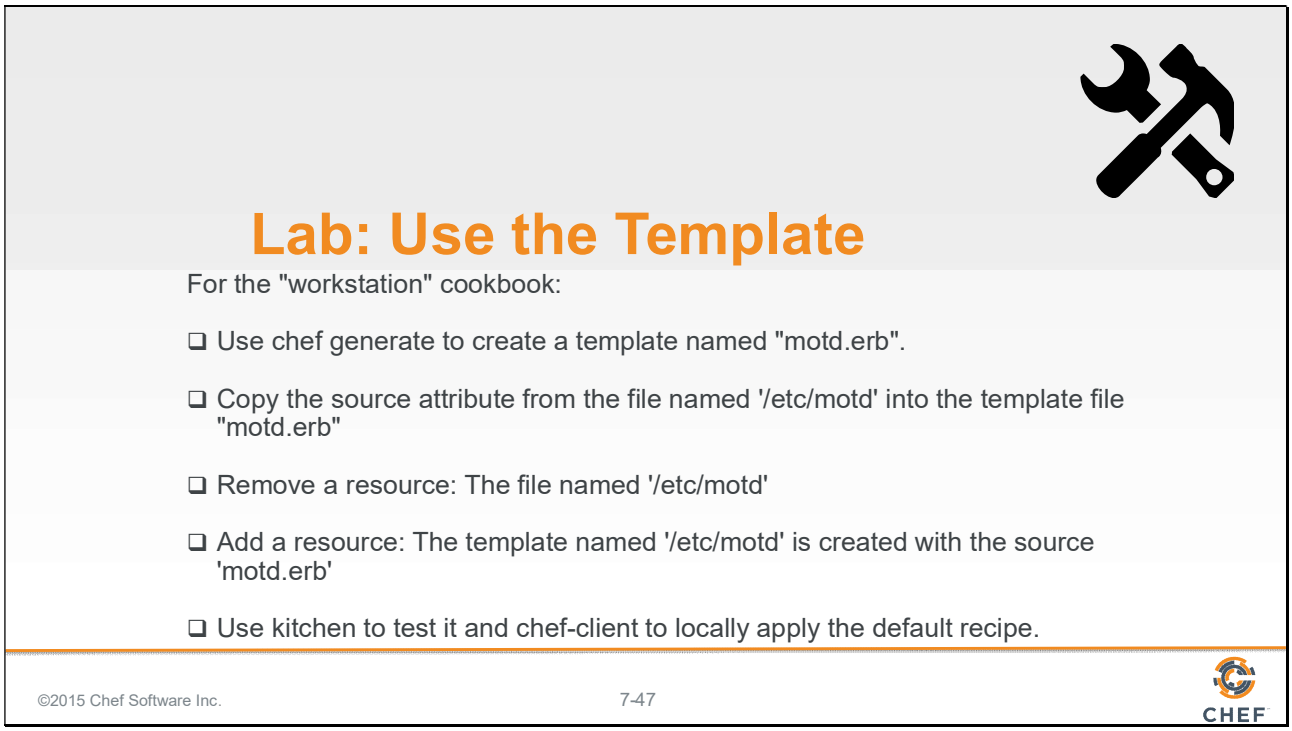
If everything converges correctly, update the version number. As mentioned previously, this is a patch fix.

Slide 46



Return to the cookbook directory and add all the changed files and commit them with a message.

Slide 47



It's time to do that again--this time for the workstation cookbook.

Generate a template named 'motd', copy in the source attribute from the file resource, and then update it to use ERB tags.

Then come back to the recipe. Change it to a template resource and then add a source attribute whose value is that partial path to the new template you created.

Instructor Note: Allow 15 minutes to complete this exercise.

Slide 48

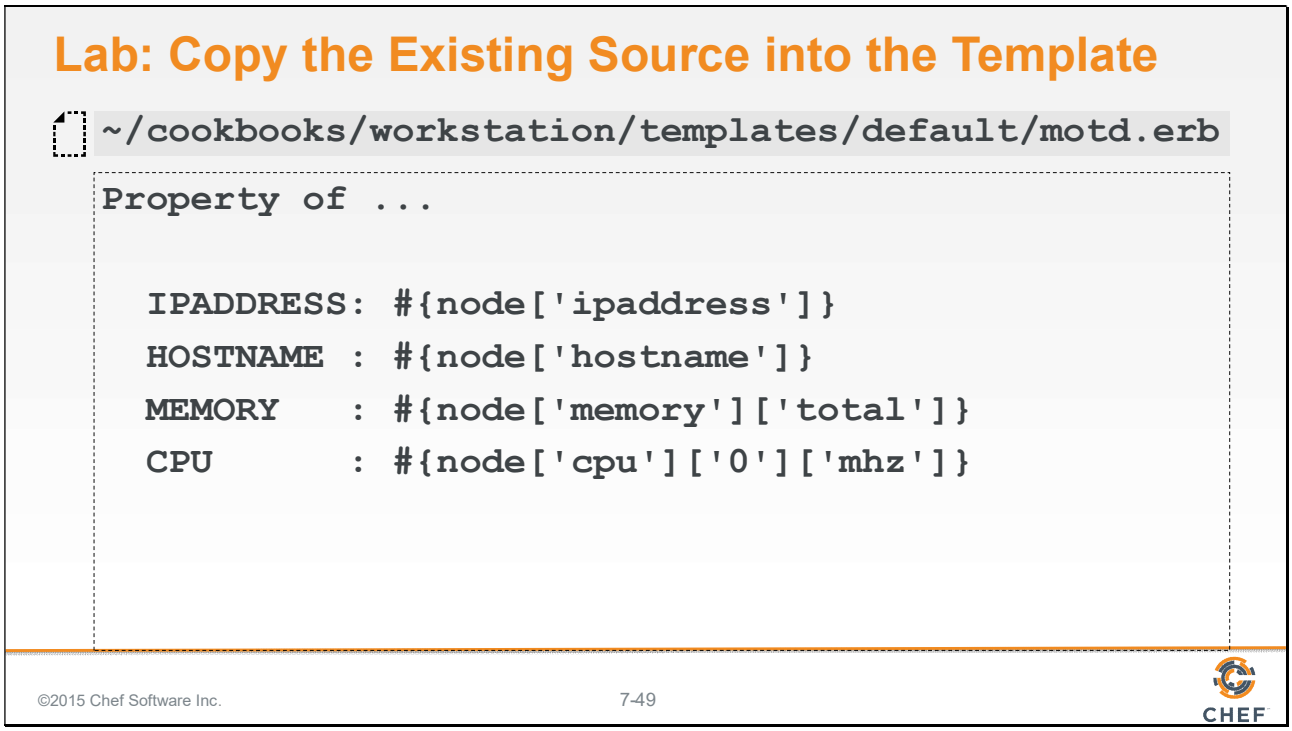
Lab: Return Home and Generate the Template

```
$ 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
```


Return to the home directory. Run the command to generate the template named 'motd' in the workstation cookbook.

Slide 49



We can start by copying and pasting the existing content for the Message of the Day file into the template file.

Slide 50

Lab: Update the motd.erb to Use ERB

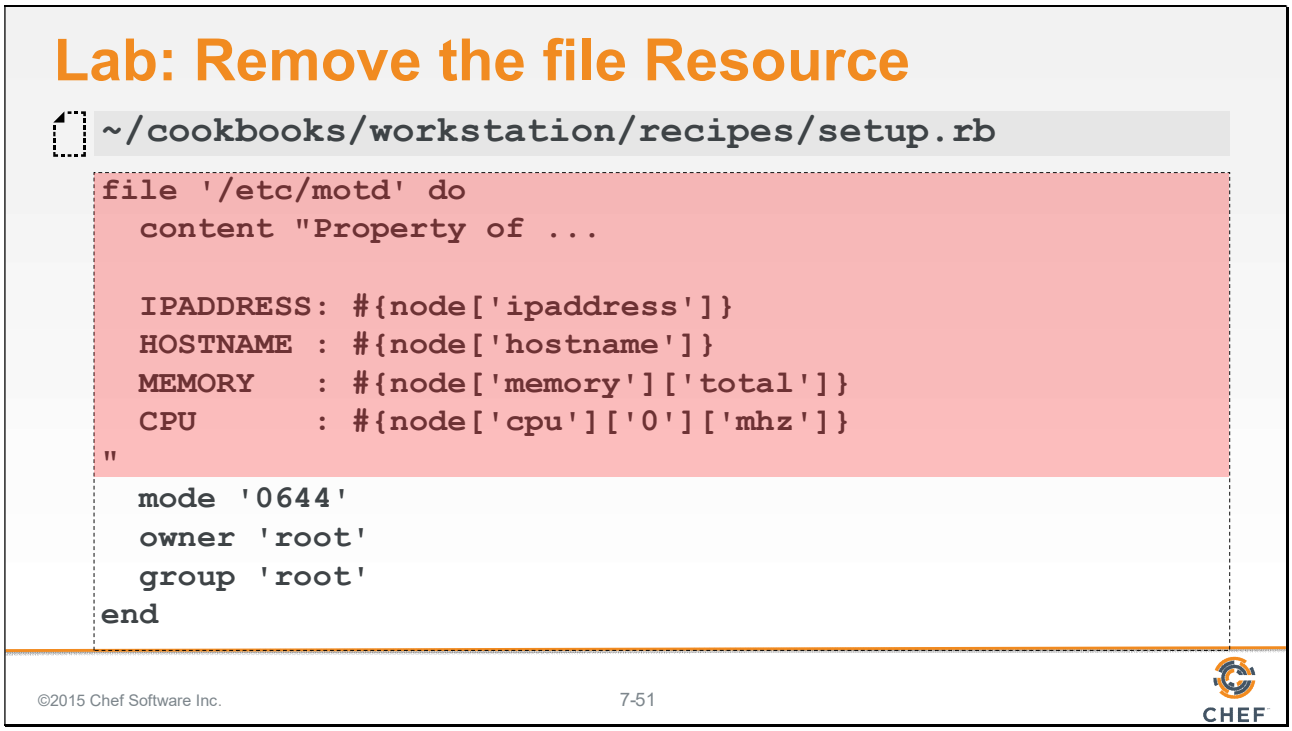
~/cookbooks/workstation/templates/default/motd.erb

```
Property of ...

  IPADDRESS: <%= node['ipaddress'] %>
  HOSTNAME : <%= node['hostname'] %>
  MEMORY   : <%= node['memory']['total'] %>
  CPU      : <%= node['cpu']['0']['mhz'] %>
```

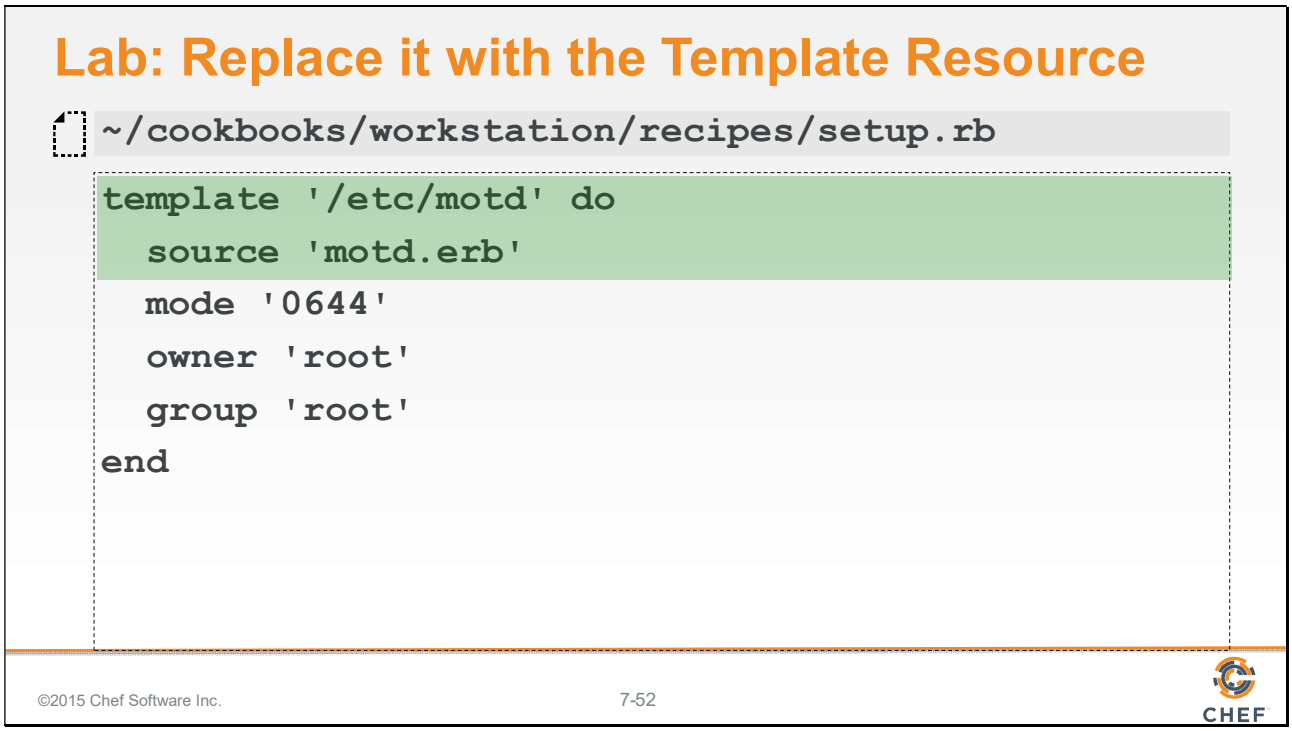

Replace all the string interpolation with ERB tags.

Slide 51



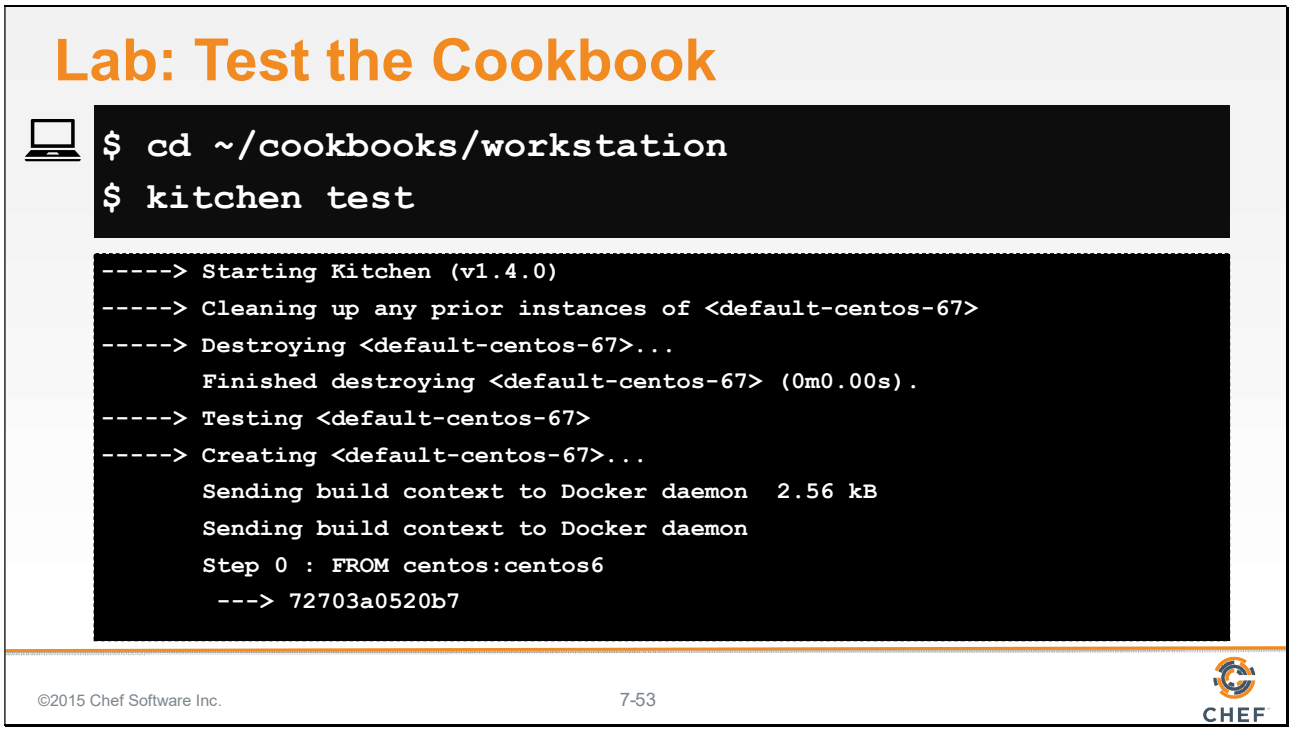
Remove the file resource from the setup recipe.

Slide 52



...and replace it with the Template resource. The source attribute specifies the file path 'motd.erb' - the new template file that was created.

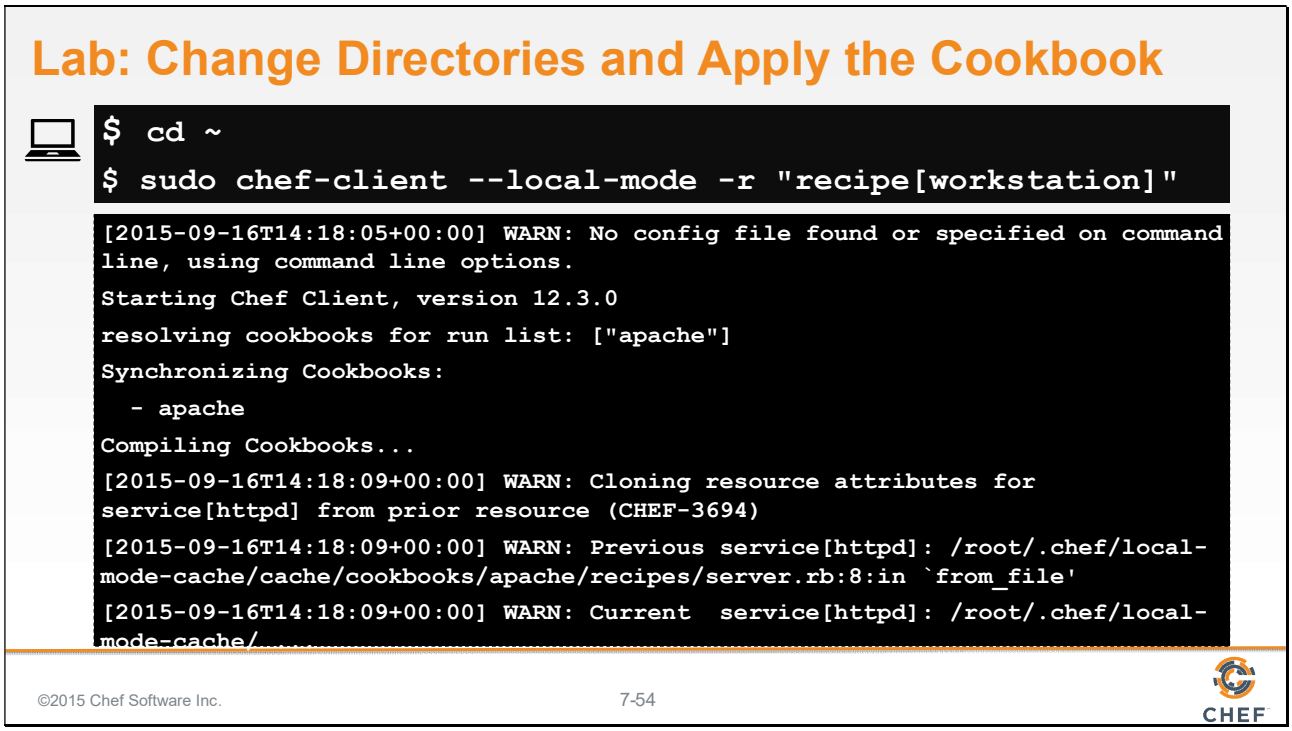
Slide 53



Since kitchen is a cookbook testing tool, you need to move into the cookbook's directory.

Then run the 'kitchen test' command, addressing any issues if they show up.

Slide 54



When all the tests pass, return to the home directory, so you can execute 'chef-client'.

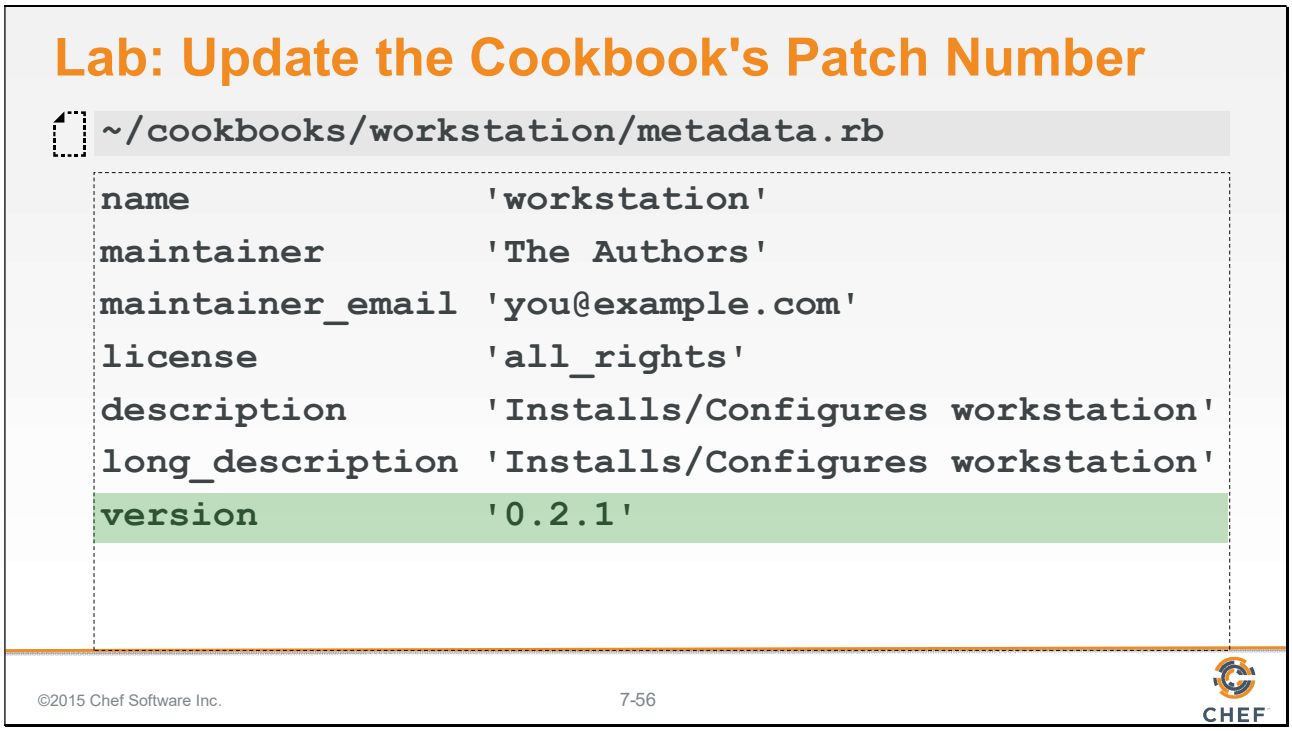
And then apply the workstation cookbook's default recipe to the local system.

Slide 55

With everything working it is time to update the patch version and commit the changes.

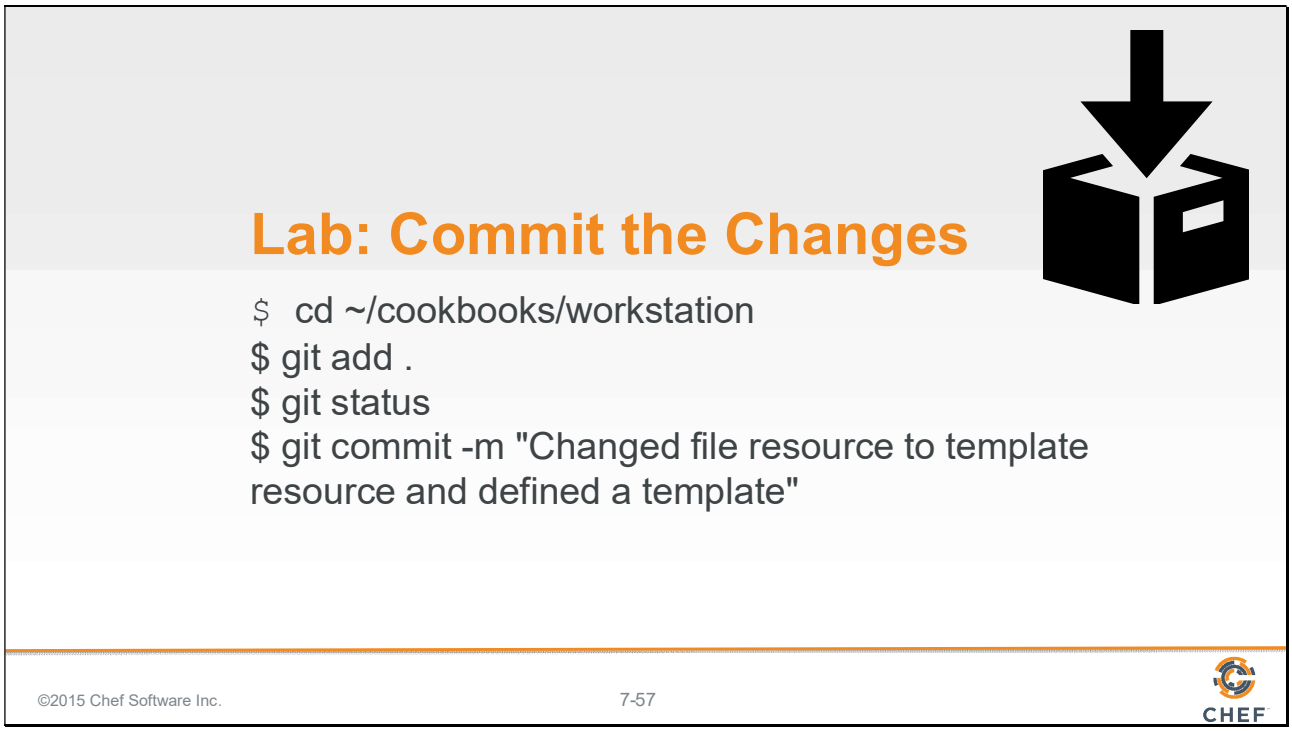
Instructor Note: Allow 5 minutes to complete this exercise.

Slide 56



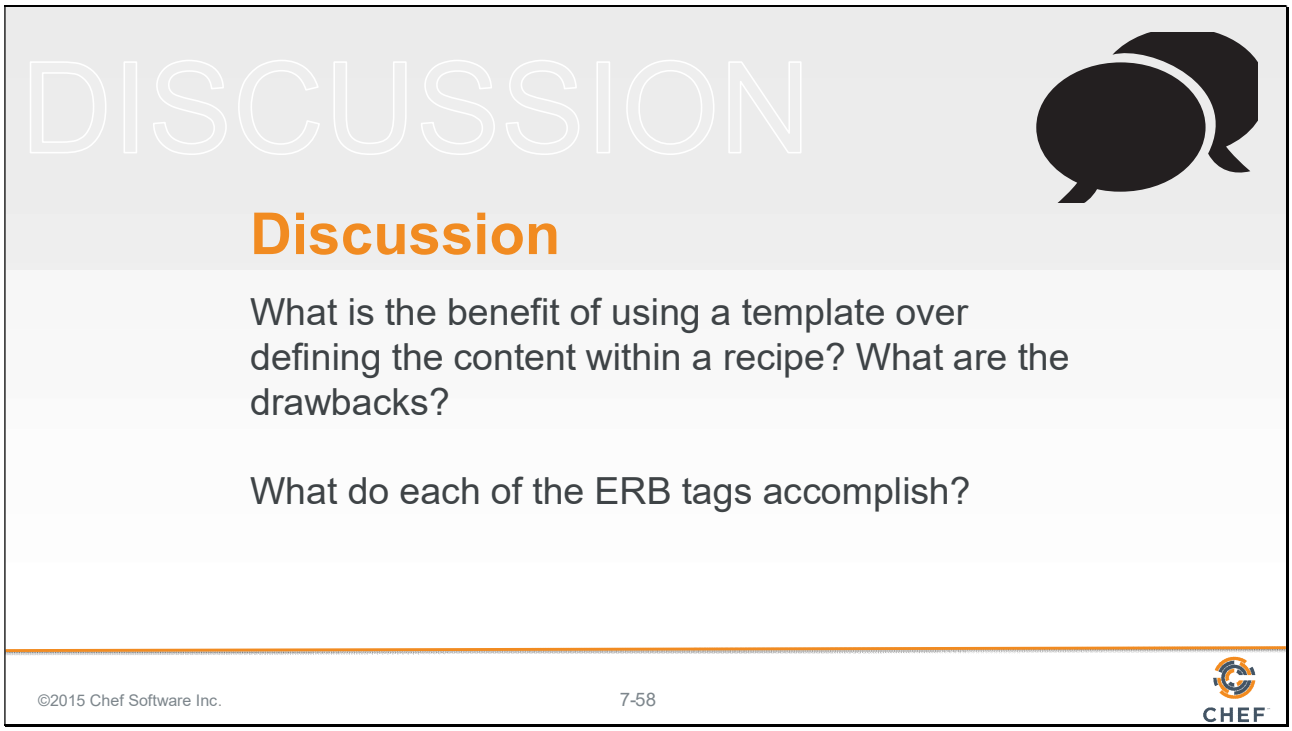
Update the patch version number for the workstation cookbook.

Slide 57



Add and then commit the changes to the workstation cookbook.

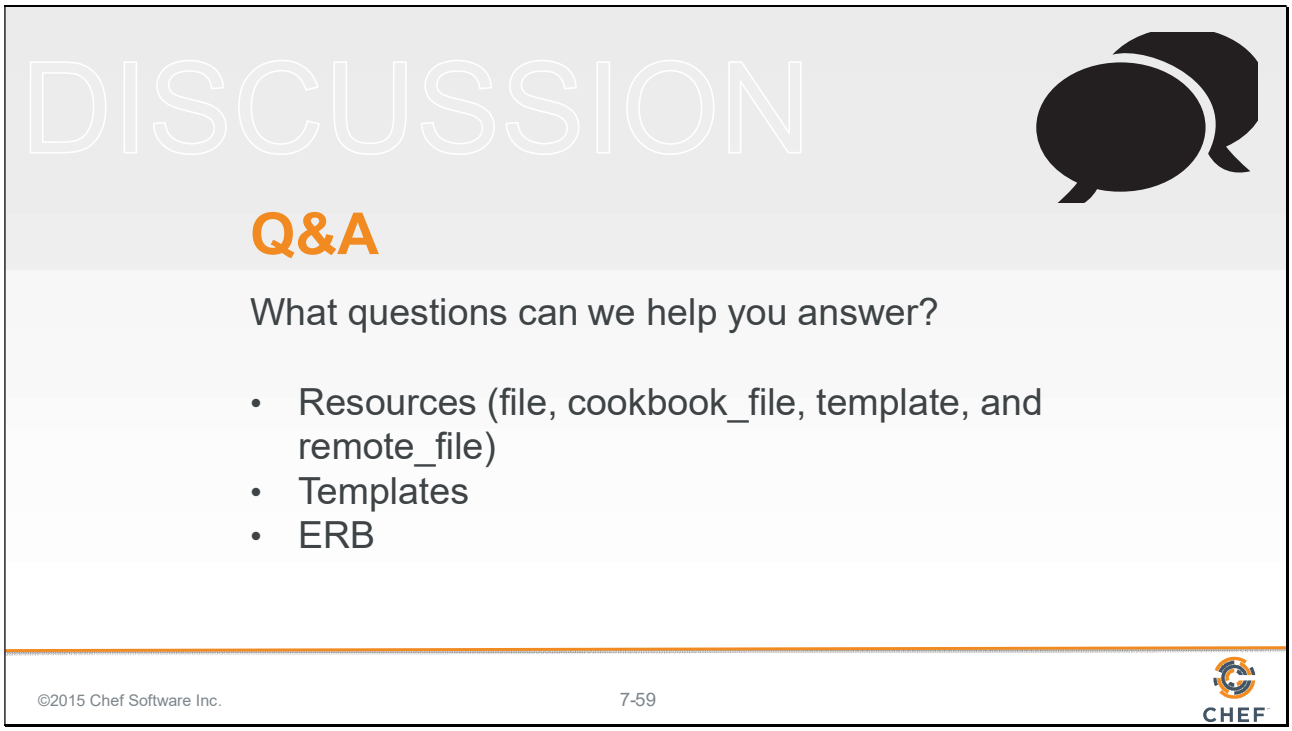
Slide 58



Answer these questions.

With your answers, turn to another person and alternate asking each other these questions and sharing your answers.

Slide 59



What questions can we help you answer?

Generally or specifically about resources, templates, and ERB.

Slide 60