Ohai and the Node Object

Finding and Displaying Information About Our System

CHEF

Objectives

After completing this module, you should be able to:

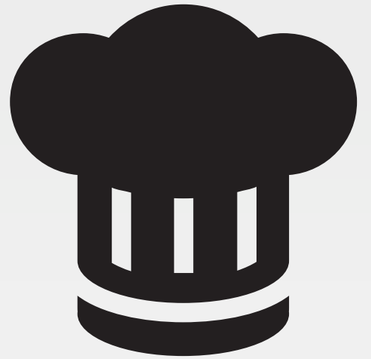
- Capture details about a system
- Use the node object within a recipe
- Use Ruby's string interpolation
- Update the version of a cookbook

Managing a Large Number of Servers

Have you ever had to manage a large number of servers that were almost identical?

How about a large number of identical servers except that each one had to have host-specific information in a configuration file?

CHEF™



Details About the System

Displaying system details in the default web page definitely sounds useful.

Objective:

- ❑ Find out various details about the system
- ❑ Update the web page file contents, in the "myiis" cookbook, to include system details
- ❑ Use chef-client to locally apply the "myiis" cookbook's default recipe

Some Useful System Data

- IP Address
- hostname
- memory
- CPU - MHz

GL: Finding the IP Address

```
> ipconfig
```

```
Windows IP Configuration


Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . : ec2.internal
   Link-local IPv6 Address . . . . . : fe80::2da8:4ba7:45e2:e863%21
   IPv4 Address. . . . . . . . . . . : 172.31.21.21
   Subnet Mask . . . . . . . . . . . : 255.255.240.0
   Default Gateway . . . . . . . . . : 172.31.16.1
```


CHEF

GL: Finding the Hostname

```
> hostname
```

```
WIN-KRQSVD3RFM7
```


CHEF

GL: Finding the Total Memory

```
> wmic ComputerSystem get TotalPhysicalMemory
```

```
TotalPhysicalMemory
8052654080
```

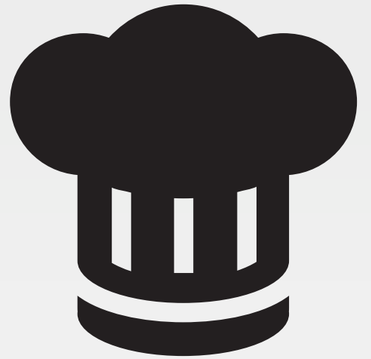

CHEF

GL: Finding the CPU MHz

```
> wmic cpu get name
```

```
Name
Intel(R) Xeon(R) CPU E5-2666 v3 @ 2.90GHz
```


CHEF



Details About the System

Displaying system details in the default web page definitely sounds useful.

Objective:

- ✓ Find out various details about the system
- ❑ Update the web page file contents, in the "myiis" cookbook, to include system details
- ❑ Use chef-client to locally apply the "myiis" cookbook's default recipe

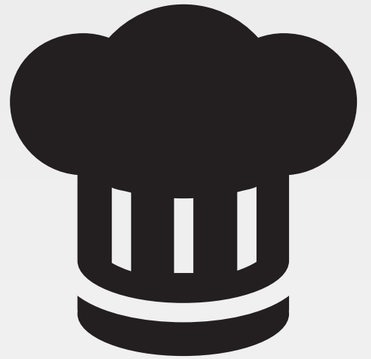
GL: Adding the CPU

~\cookbooks\myiis\recipe\server.rb

```
# ... POWERSHELL_SCRIPT RESOURCE ...

file 'C:\inetpub\wwwroot\Default.htm' do
  content '<h1>Hello, world!</h1>
<h2>ipaddress: 172.31.21.21</h2>
<h2>hostname: WIN-KRQSVD3RFM7</h2>
<h2>total memory: 8052654080</h2>
<h2>CPU Mhz: 2.90GHz</h2>'
end

# ... SERVICE RESOURCE ...
```



Details About the Node

Displaying system details in the default web page definitely sounds useful.

Objective:

- ✓ Find out various details about the system
- ✓ Update the web page file contents, in the "myiis" cookbook, to include system details
- ❑ Use chef-client to locally apply the "myiis" cookbook's default recipe

GL: Return Home and Apply myiis Cookbook

```
> cd ~
> chef-client --local-mode -r "recipe[myiis]"
```

```
Converging 3 resources
Recipe: myiis::server
  * powershell_script[Install IIS] action run
    - execute "C:\Windows\system32\WindowsPowerShell\v1.0\powershell.exe" -
NoLogo -NonInteractive -NoProfile -ExecutionPolicy Bypass -InputFormat Non
e -File "C:/tmp/chef-script20180919-996-rjuiw4.ps1"
…
- update content in file C:\inetpub\wwwroot\Default.htm from 17d291 to f37fdd
    --- C:\inetpub\wwwroot\Default.htm  2018-09-18 20:41:33.000000000 +0000
    +++ C:\inetpub\wwwroot/chef-Default20180919-996-om7yot.htm  2018-09-19
17:25:01.000000000 +0000
    @@ -1,2 +1,6 @@
     <h1>Hello, world!</h1>
    +<h2>ipaddress: 172.31.21.21</h2>
    +<h2>hostname: WIN-KRQSVD3REM7</h2>
```

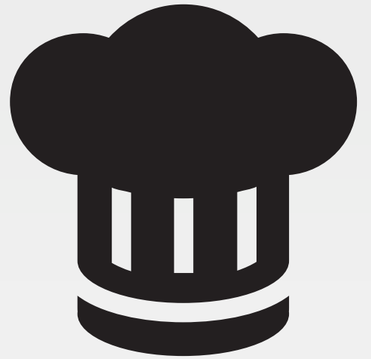

CHEF

GL: Verify the Default Page Returns the Details

```
> Invoke-WebRequest localhost
```

```
StatusCode        : 200
StatusDescription : OK
Content           : <h1>Hello, world!</h1>
                    <h2>ipaddress: 172.31.21.21</h2>
                    <h2>hostname: WIN-KRQSVD3RFM7</h2>
                    <h2>total memory: 8052654080</h2>
                    <h2>CPU Mhz: 2.90GHz</h2>
RawContent        : HTTP/1.1 200 OK
                    Accept-Ranges: bytes
                    Content-Length: 150
                    Content-Type: text/html
                    Date: Tue, 19 Sep 2018 17:27:19 GMT
```


CHEF



Details About the System

Displaying system details in the default web page definitely sounds useful.

Objective:

- ✓ Find out various details about the system
- ✓ Update the web page file contents, in the "myiis" cookbook, to include system details
- ✓ Use chef-client to locally apply the "myiis" cookbook's default recipe

DISCUSSION

Capturing System Data

What are the limitations of the way we captured this data?

How accurate will our recipe be when we deploy it on other systems?

CHEF

Hard Coded Values

The values that we have derived at this moment may not be the correct values when we deploy this recipe again even on the same system!

DISCUSSION

Data In Real Time

How could we capture this data in real-time?

CHEF

CONCEPT

Ohai!

Ohai is a tool that already captures all the data that we similarly demonstrated finding.

http://docs.chef.io/ohai.html

CHEF

GL: Running Ohai!

```
> ohai
```


```
"kernel": {
    "os_info": {
      "boot_device": "\\Device\\HarddiskVolume1",
      "build_number": "9600",
      "build_type": "Multiprocessor Free",
      "caption": "Microsoft Windows Server 2012 R2 Standard",
      "code_set": "1252",
      "country_code": "1",
      "creation_class_name": "Win32_OperatingSystem",
      "cs_creation_class_name": "Win32_ComputerSystem",
      "csd_version": null,
      "cs_name": "WIN-VBVUUIACREJ",
…
```


CHEF

CONCEPT

All About The System

Ohai queries the operating system with a number of commands, similar to the ones demonstrated.

The data is presented in JSON (JavaScript Object Notation).

http://docs.chef.io/ohai.html

CHEF

CONCEPT

ohai + chef-client = <3

chef-client and chef-apply automatically executes ohai and stores the data about the node in an object we can use within the recipes named node.

http://docs.chef.io/ohai.html

CHEF

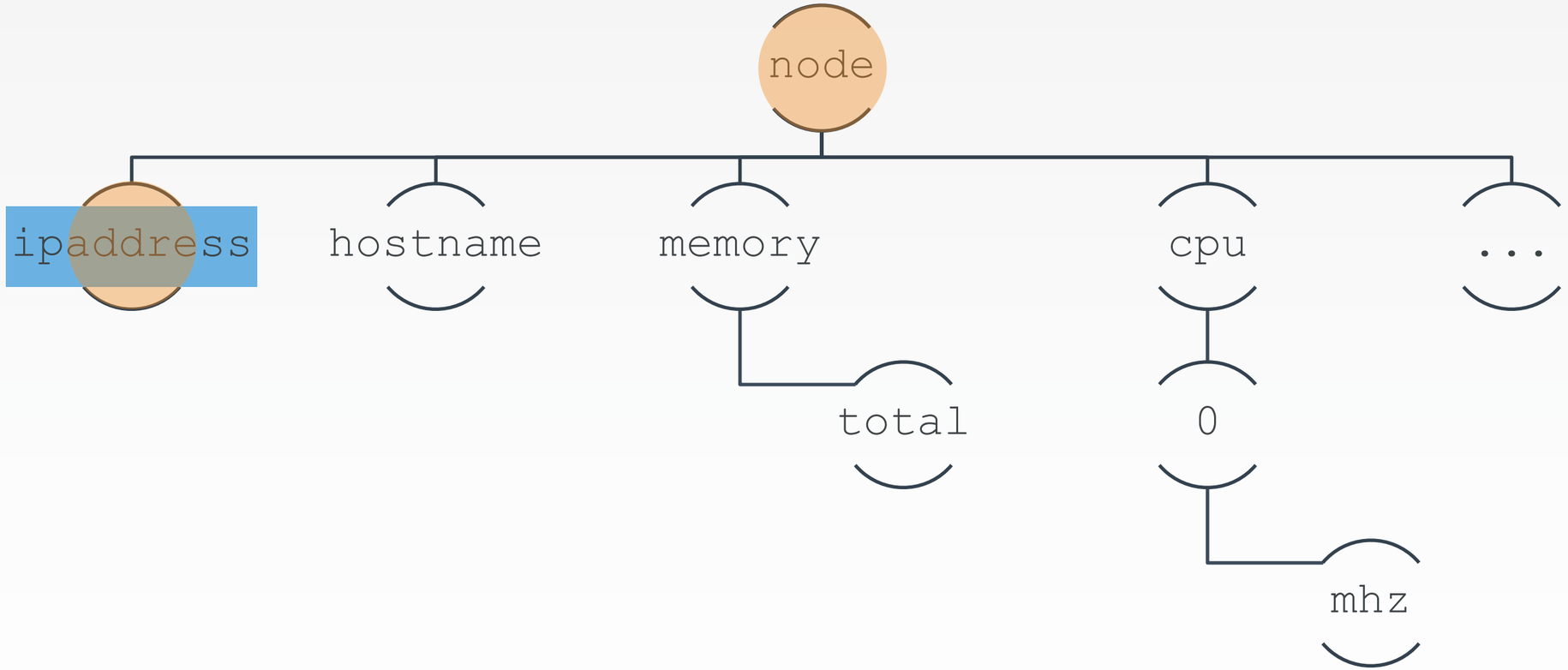
CONCEPT

The Node Object

The node object is a representation of our system.
It stores all the attributes found about the system.

http://docs.chef.io/nodes.html#attributes

The Node

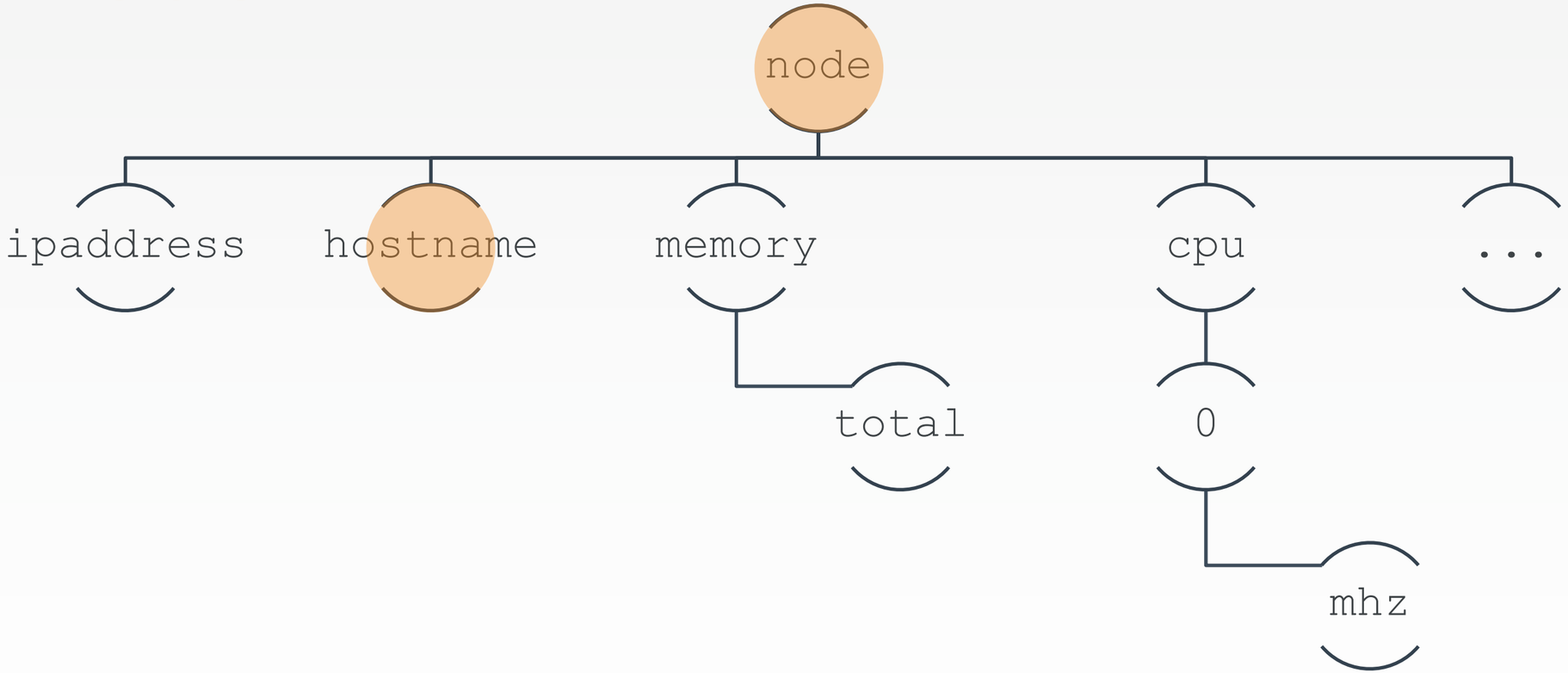



```
IPADDRESS: 172.31.21.21
```

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


CHEF

The Node

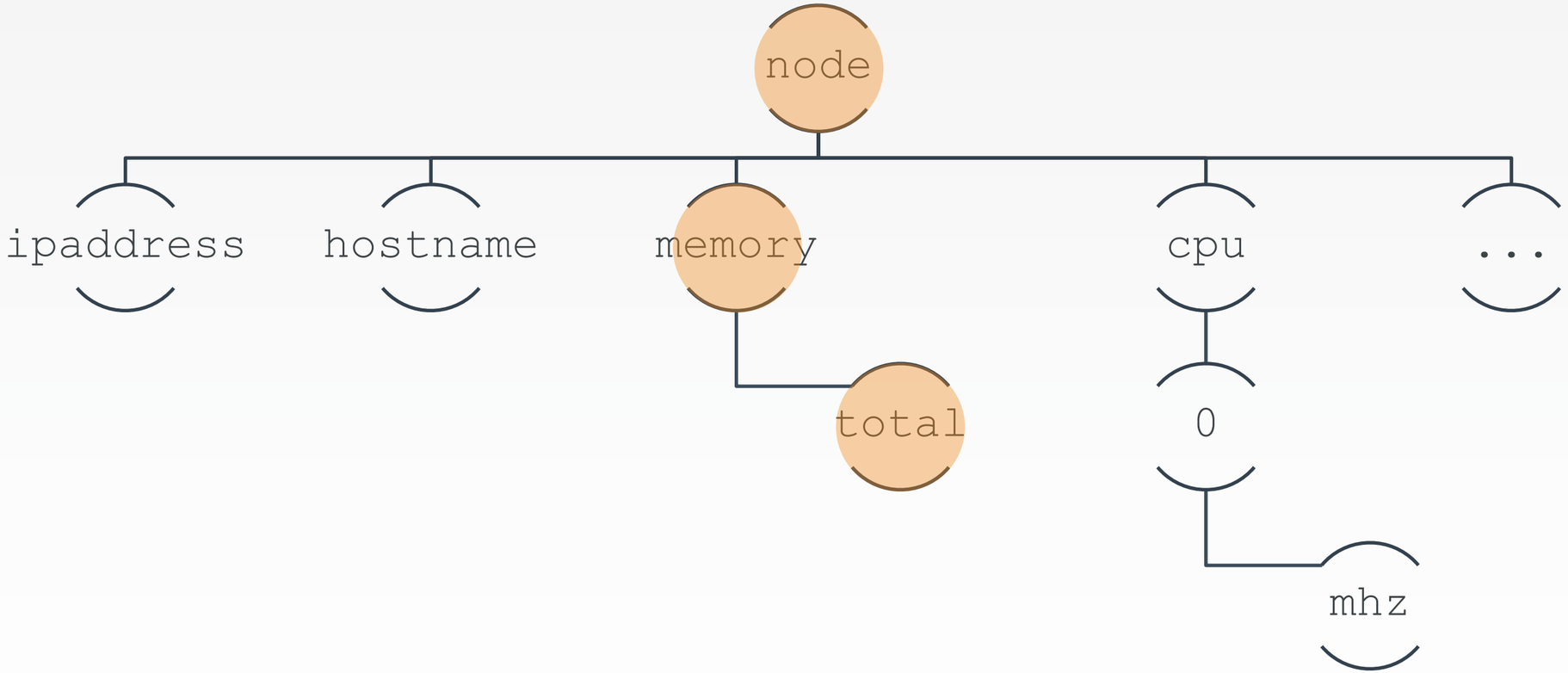



```
HOSTNAME: WIN-KRQSVD3RFM7
```

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


CHEF

The Node

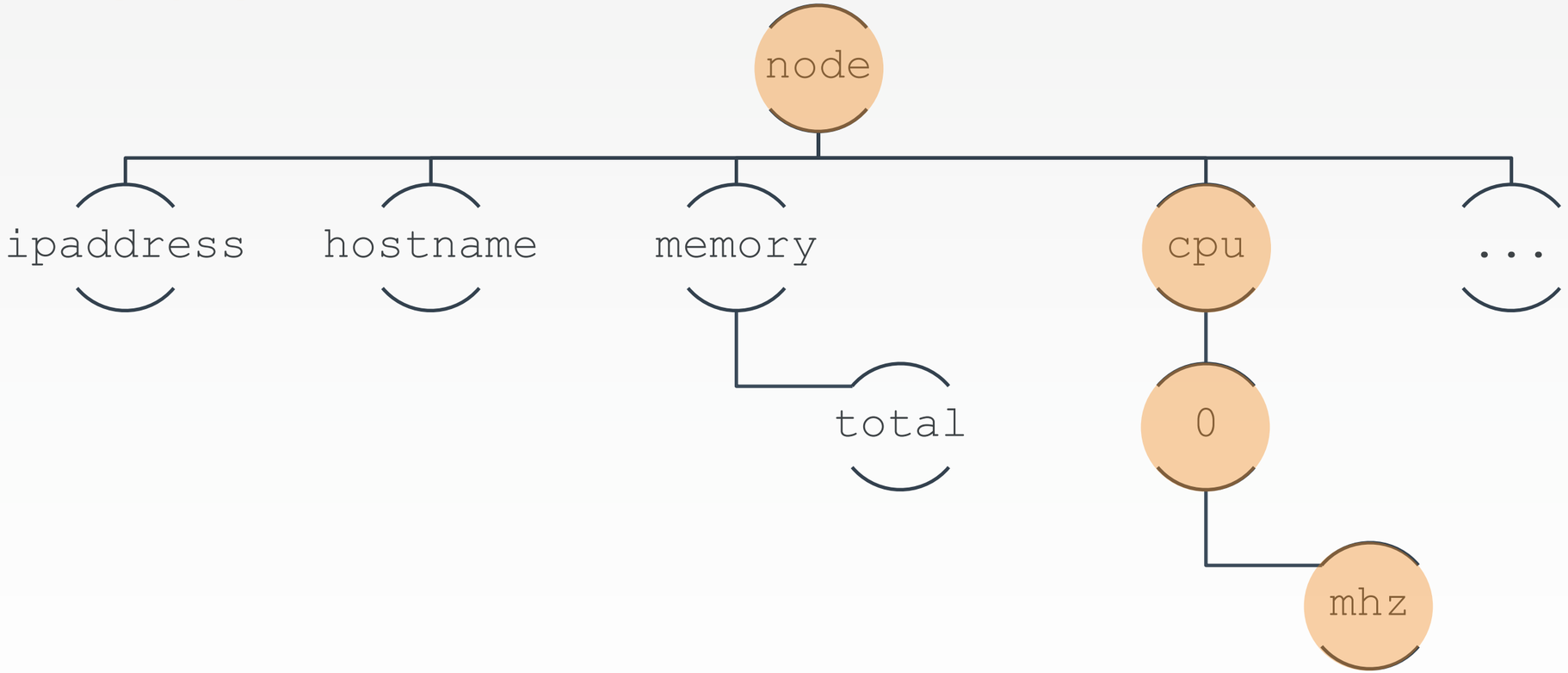



```
MEMORY: 7863920kB
```

```
"Memory: #{node['memory']['total']}"
```


CHEF

The Node




```
CPU: 2900
```

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


CHEF

CONCEPT

String Interpolation

```
I have 4 apples
```

```
apple_count = 4
puts "I have #{apple_count} apples"
```

http://en.wikipedia.org/wiki/String_interpolation#Ruby

CHEF

CONCEPT

String Interpolation

```
I have 4 apples
```

```
apple_count = 4
puts "I have #{apple_count} apples"
```

http://en.wikipedia.org/wiki/String_interpolation#Ruby

CHEF

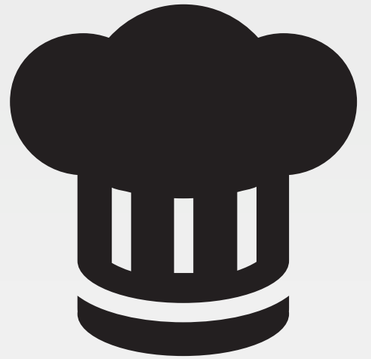
CONCEPT

String Interpolation

```
I have 4 apples
```

```
apple_count = 4
puts "I have #{apple_count} apples"
```


CHEF



Using Node Attributes

Hard-coding the values was a start, but a better approach would be to replace with the dynamic values found from Ohai.

Objective:

- ❑ Update the web page file contents, in the "myiis" cookbook, to include system details from node attributes.

GL: Using the Node's Attributes

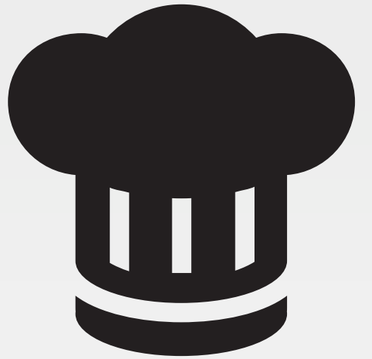
~\cookbooks\myiis\recipe\server.rb

```
# ... POWERSHELL_SCRIPT RESOURCE ...

file 'c:\inetpub\wwwroot\Default.htm' do
  content "<h1>Hello, world!</h1>
<h2>ipaddress: #{node['ipaddress']}</h2>
<h2>hostname: #{node['hostname']}</h2>
<h2>total memory: #{node['memory']['total']}</h2>
<h2>CPU Mhz: #{node['cpu']['0']['mhz']}</h2>"
end

# ... SERVICE RESOURCE ...
```


CHEF



Using Node Attributes

That feels much better!

Objective:

- ✓ Update the web page file contents, in the "myiis" cookbook, to include system details from node attributes.

CHEF

Lab: Verify the Changes

- Change directory into the home directory
- Run chef-client locally to verify the "myiis" cookbook's default recipe.

Lab: Apply the 'myiis' Cookbook's Default Recipe

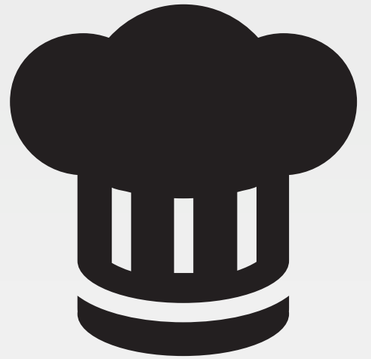
```
> cd ~
> chef-client --local-mode -r "recipe[myiis]"
```

```
Converging 3 resources
Recipe: myiis::server
  * powershell_script[Install IIS] action run
    - execute "C:\Windows\system32\WindowsPowerShell\v1.0\powershell.exe" -
NoLogo -NonInteractive -NoProfile -ExecutionPolicy Bypass -InputFormat Non
e -File "C:/tmp/chef-script20180919-3724-1wk70vt.ps1"
  * file[C:\inetpub\wwwroot\Default.htm] action create
    - update content in file C:\inetpub\wwwroot\Default.htm from f37fdd to
38be49
    --- C:\inetpub\wwwroot\Default.htm  2018-09-19 17:25:01.000000000 +0000
    +++ C:\inetpub\wwwroot/chef-Default20180919-3724-1mczbah.htm       2018-
09-19 17:34:44.000000000 +0000
```


CHEF

Lab: Verify the Changes

- ✓ Change directory into the home directory
- ✓ Run chef-client locally to verify the "myiis" cookbook's default recipe.



Change Means a New Version

Let's bump the cookbook's version number

Objective:

- ❑ Determine the new version number
- ❑ Update the version of the "myiis" cookbook

CONCEPT

Cookbook Versions

A cookbook version represents a set of functionality that is different from the cookbook on which it is based.

https://docs.chef.io/cookbook_versions.html

CHEF

CONCEPT

Semantic Versions

Given a version number **MAJOR**.**MINOR**.**PATCH**, increment the:

- **MAJOR** version when you make incompatible API changes
- **MINOR** version when you add functionality in a backwards-compatible manner
- **PATCH** version when you make backwards-compatible bug fixes

http://semver.org

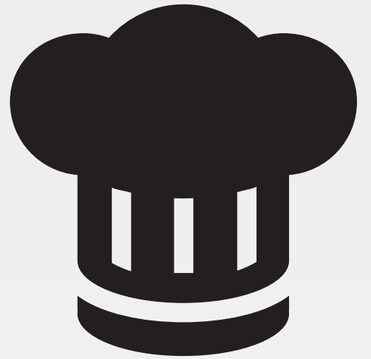
CHEF

DISCUSSION

Major, Minor, or Patch?

What kind of changes did you make to the cookbook?

CHEF



Change Means a New Version

Let's bump the version number

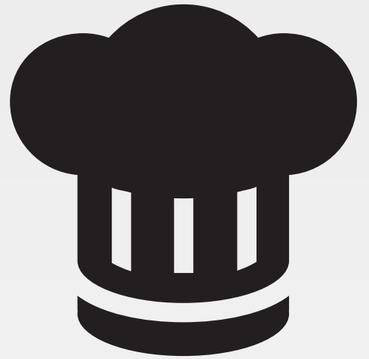
Objective:

- ✓ Determine the new version number
- ❑ Update the version of the "myiis" cookbook

GL: Update the Cookbook Version

~\cookbooks\myiis\metadata.rb

```
name                'myiis'
maintainer          'The Authors'
maintainer_email    'you@example.com'
license             'all_rights'
description         'Installs/Configures myiis'
long_description    'Installs/Configures myiis'
version             '0.2.0'
```



Change Means a New Version

Lets bump the version number

Objective:

- ✓ Determine the new version number
- ✓ Update the version of the "myiis" cookbook

DISCUSSION

Discussion

What is the major difference between a single-quoted string and a double-quoted string?

How are the details about the system available within a recipe?

How does the version number help convey information about the state of the cookbook?

CHEF

DISCUSSION

Q&A

What questions can we help you answer?

- Ohai
- Node Object
- Node Attributes
- String Interpolation
- Semantic Versions

CHEF

CHEF™