Why Use Custom Resources

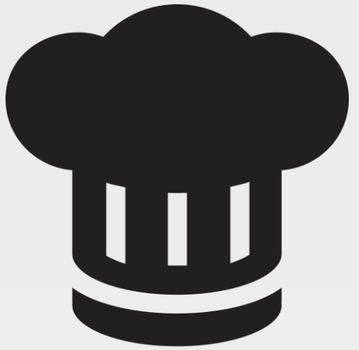
CHEF

Objectives

After completing this module, you should be able to:

- Determine when a Custom Resource would be beneficial for clarity and reusability

EXERCISE



Evaluation Before Pursuit

Just because I can does not mean I should. It is important to implement solutions that are arguably better software design.

Objective:

- ❑ Define the judgment criteria
- ❑ Evaluate a code sample

CHEF

CONCEPT

Software Quality Standards

When defining resources within our recipes we are writing software. Software has a number of quality characteristics that have already been defined. ISO/IEC 9126 is an international standard for evaluation of software quality.

CONCEPT

Software Quality Standards

- Functionality
- Reliability
- Usability
- Efficiency
- Maintainability
- Portability

CHEF

CONCEPT

Software Quality Standards

- **Functionality**
- Reliability
- Usability
- Efficiency
- Maintainability
- Portability

Does the code accomplish what it is designed to accomplish?

CHEF

CONCEPT

Software Quality Standards

- Functionality
- **Reliability**
- Usability
- Efficiency
- Maintainability
- Portability

Is the solution able to withstand fault and recover from a failure?

CHEF

CONCEPT

Software Quality Standards

- Functionality
- Reliability
- **Usability**
- Efficiency
- Maintainability
- Portability

Is the code easy to understand?
Is it easy to learn?

CHEF

CONCEPT

Software Quality Standards

- Functionality
- Reliability
- Usability
- **Efficiency**
- Maintainability
- Portability

Does the code consume too many physical resources when it executes (e.g. CPU, memory)?

CHEF

CONCEPT

Software Quality Standards

- Functionality
- Reliability
- Usability
- Efficiency
- **Maintainability**
- Portability

Are you able to easily adapt the solution? Is it testable?

CHEF

CONCEPT

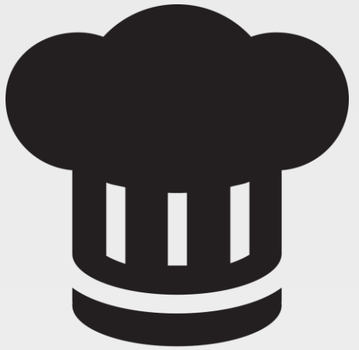
Software Quality Standards

- Functionality
- Reliability
- Usability
- Efficiency
- Maintainability
- **Portability**

Can the software adapt to changes in its environment? Or changes to its requirements?

CHEF

EXERCISE



Examine the Code Sample

With the criteria defined we can now examine code samples...

Objective:

- ✓ Define the judgment criteria
- ❑ Evaluate a code sample

CHEF

Resource Implementation v Custom Resource

```
directory '/srv/apache/admins/html' do
  recursive true
  mode '0755'
end

template '/etc/httpd/conf.d/admins.conf' do
  source 'conf.erb'
  mode '0644'

variables(document_root:'/srv/apache/admins/ht
ml',
port: 8080)
  notifies :restart, 'service[httpd]'
end

file '/srv/apache/admins/html/index.html' do
  content '<h1>Welcome admins!</h1>'
end
```

```
apache_vhost 'admins' do
  site_port 8080
end
```

Functionality | Reliability | Usability | Efficiency | Maintainability | Portability

Does the code accomplish what it is designed to accomplish?

CHEF

Resource Implementation v Custom Resource

```
directory '/srv/apache/admins/html' do
  recursive true
  mode '0755'
end

template '/etc/httpd/conf.d/admins.conf' do
  source 'conf.erb'
  mode '0644'

variables(document_root:'/srv/apache/admins/ht
ml',
port: 8080)
  notifies :restart, 'service[httpd]'
end

file '/srv/apache/admins/html/index.html' do
  content '<h1>Welcome admins!</h1>'
end
```

```
apache_vhost 'admins' do
  site_port 8080
end
```

Functionality | **Reliability** | Usability | Efficiency | Maintainability | Portability

Is the solution able to withstand fault and recover from a failure?

CHEF

Resource Implementation v Custom Resource

```
directory '/srv/apache/admins/html' do
  recursive true
  mode '0755'
end

template '/etc/httpd/conf.d/admins.conf' do
  source 'conf.erb'
  mode '0644'

variables(document_root:'/srv/apache/admins/ht
ml',
port: 8080)
  notifies :restart, 'service[httpd]'
end

file '/srv/apache/admins/html/index.html' do
  content '<h1>Welcome admins!</h1>'
end
```

```
apache_vhost 'admins' do
  site_port 8080
end
```

Functionality | Reliability | **Usability** | Efficiency | Maintainability | Portability

Is the code easy to understand? Is it easy to learn?

CHEF

Resource Implementation v Custom Resource

```
directory '/srv/apache/admins/html' do
  recursive true
  mode '0755'
end

template '/etc/httpd/conf.d/admins.conf' do
  source 'conf.erb'
  mode '0644'

variables(document_root:'/srv/apache/admins/ht
ml',
port: 8080)
  notifies :restart, 'service[httpd]'
end

file '/srv/apache/admins/html/index.html' do
  content '<h1>Welcome admins!</h1>'
end
```

```
apache_vhost 'admins' do
  site_port 8080
end
```

Functionality | Reliability | Usability | **Efficiency** | Maintainability | Portability

Does the code consume too many physical resources when it executes (e.g. CPU, memory)?

CHEF

Resource Implementation v Custom Resource

```
directory '/srv/apache/admins/html' do
  recursive true
  mode '0755'
end

template '/etc/httpd/conf.d/admins.conf' do
  source 'conf.erb'
  mode '0644'

variables(document_root:'/srv/apache/admins/ht
ml',
port: 8080)
  notifies :restart, 'service[httpd]'
end

file '/srv/apache/admins/html/index.html' do
  content '<h1>Welcome admins!</h1>'
end
```

```
apache_vhost 'admins' do
  site_port 8080
end
```

Functionality | Reliability | Usability | Efficiency | **Maintainability** | Portability

Are you able to easily adapt the solution? Is it testable?

CHEF

Resource Implementation v Custom Resource

```
directory '/srv/apache/admins/html' do
  recursive true
  mode '0755'
end

template '/etc/httpd/conf.d/admins.conf' do
  source 'conf.erb'
  mode '0644'

variables(document_root:'/srv/apache/admins/ht
ml',
port: 8080)
  notifies :restart, 'service[httpd]'
end

file '/srv/apache/admins/html/index.html' do
  content '<h1>Welcome admins!</h1>'
end
```

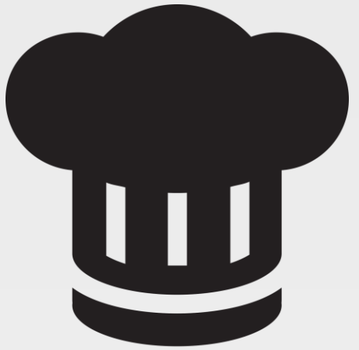
```
apache_vhost 'admins' do
  site_port 8080
end
```

Functionality | Reliability | Usability | Efficiency | Maintainability | **Portability**

Can the software adapt to changes in its environment? Or changes to its requirements?

CHEF

EXERCISE



Evaluation Before Pursuit

There are many ways to critically evaluate code ... if these do not suit your or your team find the ones that do; talk about them and share them.

Objective:

- ✓ Define the judgment criteria
- ✓ Evaluate a code sample

Discussion

What value does reviewing code for functionality, reliability, usability, efficiency, maintainability, portability bring?

CHEF

DISCUSSION

Q&A

What questions can we answer for you?

CHEF

CHEF™