

Crafting Reusable Cookbooks

Ian Henry

Jennifer Davis (remote)

Network Name: ChefConf

Access Code: ChefConf2016

Instructor

→ Ian Henry

TA's

- Robb Kidd
- Matt Stratton
- David Echols
- Stephen Lauck
- Bill Meyer

Communication

- Jennifer Davis Twitter: @sigje
- Ian Henry Twitter: @Eeyun_____

Hashtag: #chefconf

Feedback

- Constructive feedback
 - What did you find helpful?
 - What would you like to see more/less of?
 - Was there anything you found unclear?

Why are we here?

Expectations

- Code of Conduct
- Safe space to share experiences, learn from each other.
- Learn effective patterns for developing reusable cookbooks.

Agenda

Collaboration

Individuals working together with shared interactions and input, building towards a common goal.

Collaboration

- Multiple types of collaboration
- Monitoring for single points of failure
- Monitoring burnout

Smarter Teams

build better value

Tools

Accelerators of culture that if used effectively can enhance and support a culture of collaboration and affinity.

Team

- Common purpose
- Defined beliefs
- Empowered

Diversity in Teams

- Professional
- Personal
- Goals
- Cognitive Styles

Critical Habits for Teams

- Code Review
- Pairing

Code Review

- Max 90 minutes in one setting

Pairing

- Agile software development
- 2 people work together on 1 workstation
- Driver - writes code
- Observer - reviews each line
- Roles switch frequently

Types of Pairing

- Expert-expert
- Expert-novice
- Novice-novice

Team Activity 1

- Meet your team!
- What are motivations?
- What are objectives?
- Skills? Gaps in skills?
- git, chef, docker, continuous integration, continuous delivery

Local Development Environment (LDE)

- Consistent set of tools across the team
- Ability to quickly onboard new engineers

Team Activity 2

- Share with your team:
 - Do you have a team common local development environment?
 - What current tools are you using in chef development?

Time: 10 minutes

Provisioned Node - LDE

- AWS instance node
- Chef DK
 - Test Kitchen
 - Ruby
 - ChefSpec, ServerSpec
- Git

Introduction to Lab 1

- Log into node and configure git
- Create GitHub identity if you don't have one already.
- Set up GitHub keys.
- Share team tools from Activity 2.

Lab 1

Time: 30 minutes



Supermarket

- Community site with a number of cookbooks

Class Supermarket: <https://supermarket.reusablechef.com>

Chef DK

- Chef development kit
- Includes a number of utilities and software to facilitate cookbook creation
- Free download off of the website

Berkshelf

- Dependency management
- Included with Chef DK

Test Kitchen

- Included with Chef DK
- Test harness
- Sandbox automation

.kitchen.yml

- driver
- Provisioner
- platforms
- suites

.kitchen.yml driver

→ virtualization or cloud provider

Example: vagrant, docker

.kitchen.yml provisioner

- application to configure the node

Example: chef_zero

.kitchen.yml platforms

- target operating systems

Example: centos-6.5

.kitchen.yml suites

→ target configurations

Example:

```
name: default
  run_list:
    - recipe[apache::default]
  attributes:
```

Introduction to Lab 2

→ Create cookbook

Lab 2

Lab 2

Time: 20 minutes



Runbook to Recipes

Introducing MongoDB

MongoDB terminology

- document - basic unit of data for MongoDB, ordered set of keys with associated values
- collection - group of documents. analogous to table.
- instance - MongoDB can host multiple independent databases, each of which can have its own collections.

MongoDB Shell

→ mongo

full featured JavaScript interpreter

Introduction to Lab 3

Translate a runbook for installing MongoDB into chef.

[Lab 3]https://github.com/ienhnae/crc-training/blob/master/labs/lab_3.md)

Lab 3

Time: 20 minutes



Quality Cookbooks

Iterative

Issue Management

- Should be clear where to file bugs, request features.
- Should be regularly groomed

Clear and Specific Licensing

For reusable cookbooks across the community

- Apache 2.0
- GNU Public License 2.0
- GNU Public License 3.0
- MIT

Multiple Collaborators

- More than 1 person with the ability to release new cookbooks
- More than 1 person with ability to merge pull requests

Test Everything

- Use integration tests
 - Assert the state of the converged node to ensure it's in the desired state

Foodcritic and Rubocop

- Chef and Ruby linters pass cleanly

Multiple Platforms

- As applicable, supports multiple platforms

Documentation

- Clear and specific
- Includes contributing guidelines

No binaries

- Binaries should be stored in signed artifact repositories separate from the cookbook

Existence of a test cookbook

No duplication of code

Valid kitchen configuration

- Ability for others to replicate and test cookbook easily

Custom Resources over Attributes

- Attribute driven
- Resources

Custom Matchers when using Custom Resources

Example: [https://github.com/chef-cookbooks/tomcat/
blob/master/libraries/matchers.rb](https://github.com/chef-cookbooks/tomcat/blob/master/libraries/matchers.rb)

Build Cookbooks for Wrapping

Support the 95% use case

History of Customizing Chef

- Attributes
- Definitions
- LWRP and HWRP
- Custom Resources

Custom Resources

- As of 12.5
- `compat_resource` to bring custom resources to 12+

Introduction to Lab 4

Translate our MongoDB cookbook from recipes into resources.

Lab 4

Lab 4

Time: 30 minutes



Managing Risk

- Test
- Small frequent releases

Linting

- Ensure code adheres to styles and conventions
- Weave expectations into development
- Encourages collaboration

Testing

- Documenting objectives and intent
- Measuring “done”

Code Correctness

- foodcritic
- rubocop

Integration Tests

- ServerSpec
- InSpec

Rubocop

- Ruby linter
- Analyzes source code to verify the syntax and structure of our cookbooks.
- Ruby style guide
- Included with ChefDK

Why

- identify simple errors early
- improve the review process by eliminating stylistic issues

Rubocop Example

```
$ rubocop cookbooks/COOKBOOK1 cookbooks/COOKBOOK2 cookbooks/COOKBOOK4
```

Reading Rubocop Output

```
Inspecting 8 files
CWCWCCCC
```

- . means that the file contains no issues
- C means a issue with convention
- W means a warning
- E means an error
- F means an fatal error

Rubocop Cops

→ Lint

→ Rails

→ Style

Example: rubocop --lint.

Disabling Rubocop cops

Any configuration in `.rubocop.yml` is disabled.

To disable string literals:

`StringLiterals:`

`Enabled: false`

Foodcritic

- Chef linter
- Chef style guide
- Included with ChefDK

Foodcritic Example

```
$ foodcritic cookbooks/setup
```

Reading Foodcritic Output

FC008: Generated cookbook metadata needs updating: ./metadata.rb:2

ServerSpec

- Tests to verify servers functionality
- Resource types
 - Package, service, user, and many others
- Integrates with Test Kitchen
- <http://serverspec.org>

ServerSpec Generic Form

```
describe "<subject>" do
  it "<description>" do
    expect(thing).to eq result
  end
end
```

ServerSpec Potential Tests

- Is the service running?
- Is the port accessible?
- Is the expected content being served?

ServerSpec Example

```
describe 'apache' do
  it "is installed" do
    expect(package 'httpd').to be_installed
  end
  it "is running" do
    expect(service 'httpd').to be_running
  end
end
```

Reading ServerSpec Output

```
app::default  
  httpd service is running
```

```
Finished in 0.26429 seconds (files took 0.7166 seconds to load)  
1 example, 0 failures
```

InSpec

- infrastructure specification
- open-source testing framework for infrastructure
- human and machine readable
- compliance, security, and policy requirements
- Based off of ServerSpec

InSpec Example

```
describe service 'ssh-agent' do
  it { should be_running }
end
```

InSpec Example Run

```
$ inspec exec test.rb
```

.

```
Finished in 0.00901 seconds (files took 0.98501 seconds to load)
1 example, 0 failures
```

InSpec Remote Run

```
$ inspec exec test.rb -i ~/.aws/YOUR.pem -t ssh://ec2-user@54.152.7.203
```

InSpec Remote Run on Windows

```
$ inspec exec test.rb -t winrm://Admin@192.168.1.2 --password super
```

InSpec Docker Containers

```
$ inspec exec test.rb -t docker://3dda08e75838
```

More info on InSpec

[https://github.com/chef/inspec/blob/master/docs/
ctl_inspec.rst](https://github.com/chef/inspec/blob/master/docs/ctl_inspec.rst)

InSpec Audit Rules

- lots of resources to facilitate auditing nodes

InSpec host resource example

```
describe host('example.com', port: 80, proto: 'tcp') do
  it { should be_reachable }
end
```

InSpec login_defs resource example

```
describe login_defs do
  its('PASS_MAX_DAYS') { should eq '180' }
  its('PASS_MIN_DAYS') { should eq '1' }
  its('PASS_MIN_LEN') { should eq '15' }
  its('PASS_WARN_AGE') { should eq '30' }
end
```

Introduction to Lab 5

Add Ubuntu platform to the mongodb cookbook.

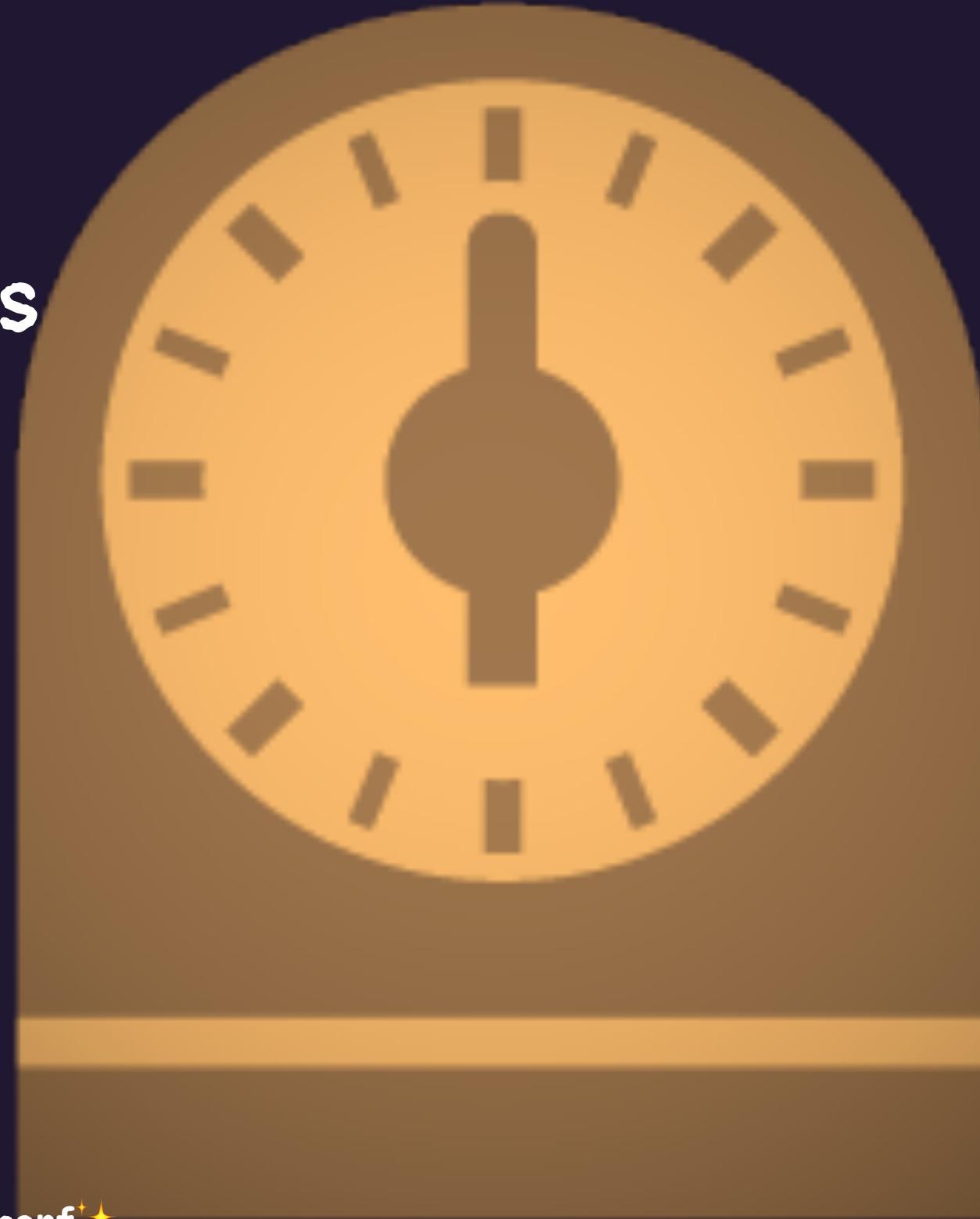
Clean any rubocop and foodcritic issues reported.

Add tests to MongoDB cookbook.

Lab 5

Lab 5

Time: 30 minutes



Community Cookbooks

Community driven.

- * Some managed by Chef. <https://github.com/chef-cookbooks>
- * Some partner managed.

CI/CD

Travis

- free (for Open Source projects)
- build files available

Chef Managed Community Cookbooks

CI

- cookstyle (included with chef dk) <https://github.com/chef/cookstyle>
- allows for consistent rubocop testing across cookbooks
- use travis

Travis Sample

- Tomcat Travis configuration
 - sudo: required (required when using docker with travis)
 - limits branch to master
 - matrix set of platforms to test
 - script tests to be run

Introduction to Lab 6

Extending Existing Cookbooks

Lab 6

Lab 6

Time: 1 hour



Introduction to Lab 7

Lab 7



Lab 7

Time: 1 hour



Review