



Chef Cookbook Testing

(Because not testing makes puppies cry)

Jon Hartman
<jon.hartman@rackspace.com>
<j@mailtrust.com>

Disclaimers

Disclaimers

- Very much TBD

Disclaimers

- Very much TBD
- Lots of alpha- and beta-quality tools

Disclaimers

- Very much TBD
- Lots of alpha- and beta-quality tools
- I only know what I'm talking about ~40% of the time

Why Test Cookbooks?

Why Test Cookbooks?

- Infrastructure as code

Why Test Cookbooks?

- Infrastructure as code
- Apply established dev concepts to infrastructure

Why Test Cookbooks?

- Infrastructure as code
- Apply established dev concepts to infrastructure
- TDD, BDD, RDD, OMGDD!

Methods of Testing

Methods of Testing

- Syntax testing

Methods of Testing

- Syntax testing
- Lint testing

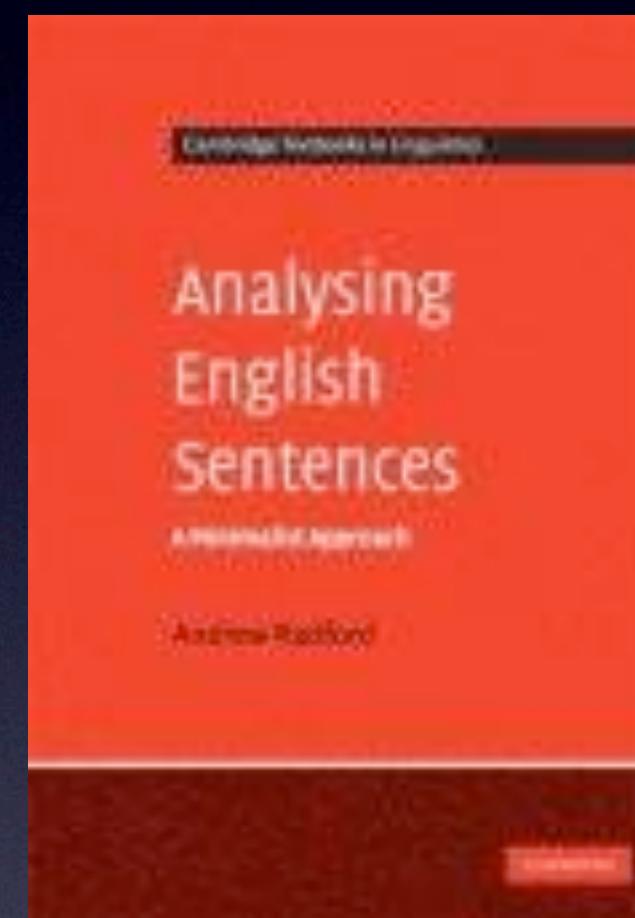
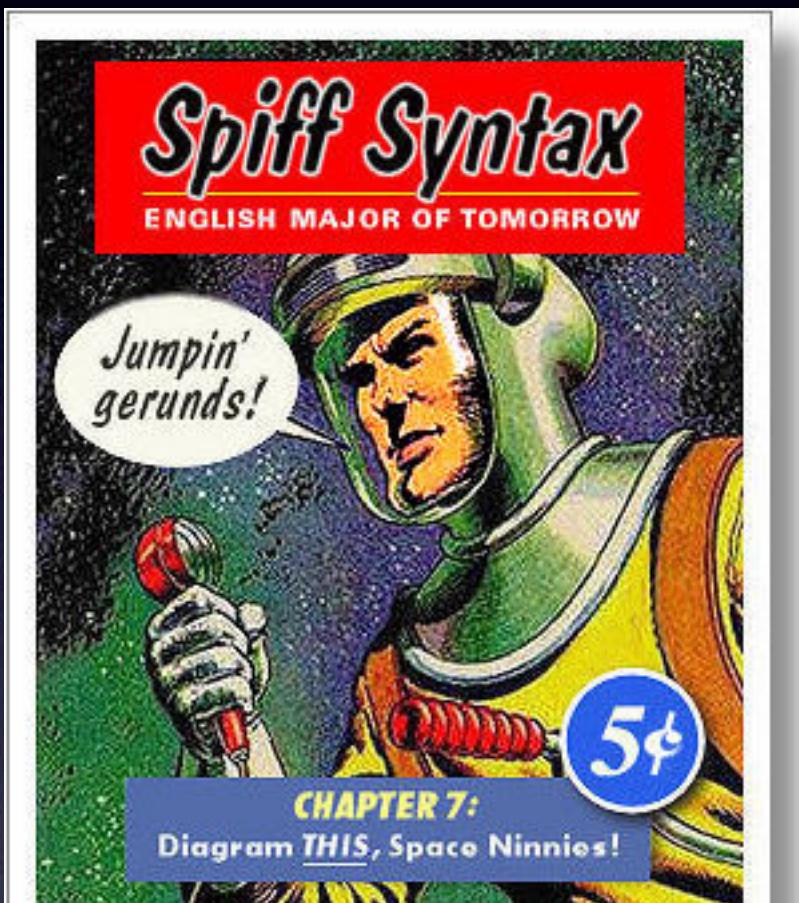
Methods of Testing

- Syntax testing
- Lint testing
- Unit testing

Methods of Testing

- Syntax testing
- Lint testing
- Unit testing
- Functional/Convergence testing

Syntax Testing



Chef Cookbook Test

Positives

Negatives

Chef Cookbook Test

Positives

- Integrates Ruby syntax with ERB syntax checks

Negatives

Chef Cookbook Test

Positives

- Integrates Ruby syntax with ERB syntax checks
- Helps catch simple errors early on

Negatives

Chef Cookbook Test

Positives

- Integrates Ruby syntax with ERB syntax checks
- Helps catch simple errors early on

Negatives

- Requires write access to a Chef cache dir

Chef Cookbook Test Demo

Chef Cookbook Test Demo

```
> knife cookbook test -c knife.rb -o cookbooks -a; rm -rf tmp
```

Chef Cookbook Test Demo

```
> knife cookbook test -c knife.rb -o cookbooks -a; rm -rf tmp
```

Running syntax check on rsapps-base

Validating ruby files

FATAL: Cookbook file recipes/sudoers.rb has a ruby syntax error:

FATAL: cookbooks/rsapps-base/recipes/sudoers.rb:28: syntax error, unexpected \$end

FATAL: Erb template templates/default/iptables/ssh_iptables.erb has a syntax error:

FATAL: -:2: syntax error, unexpected tCONSTANT, expecting keyword_do or '{' or '('

FATAL: -A FWR -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT

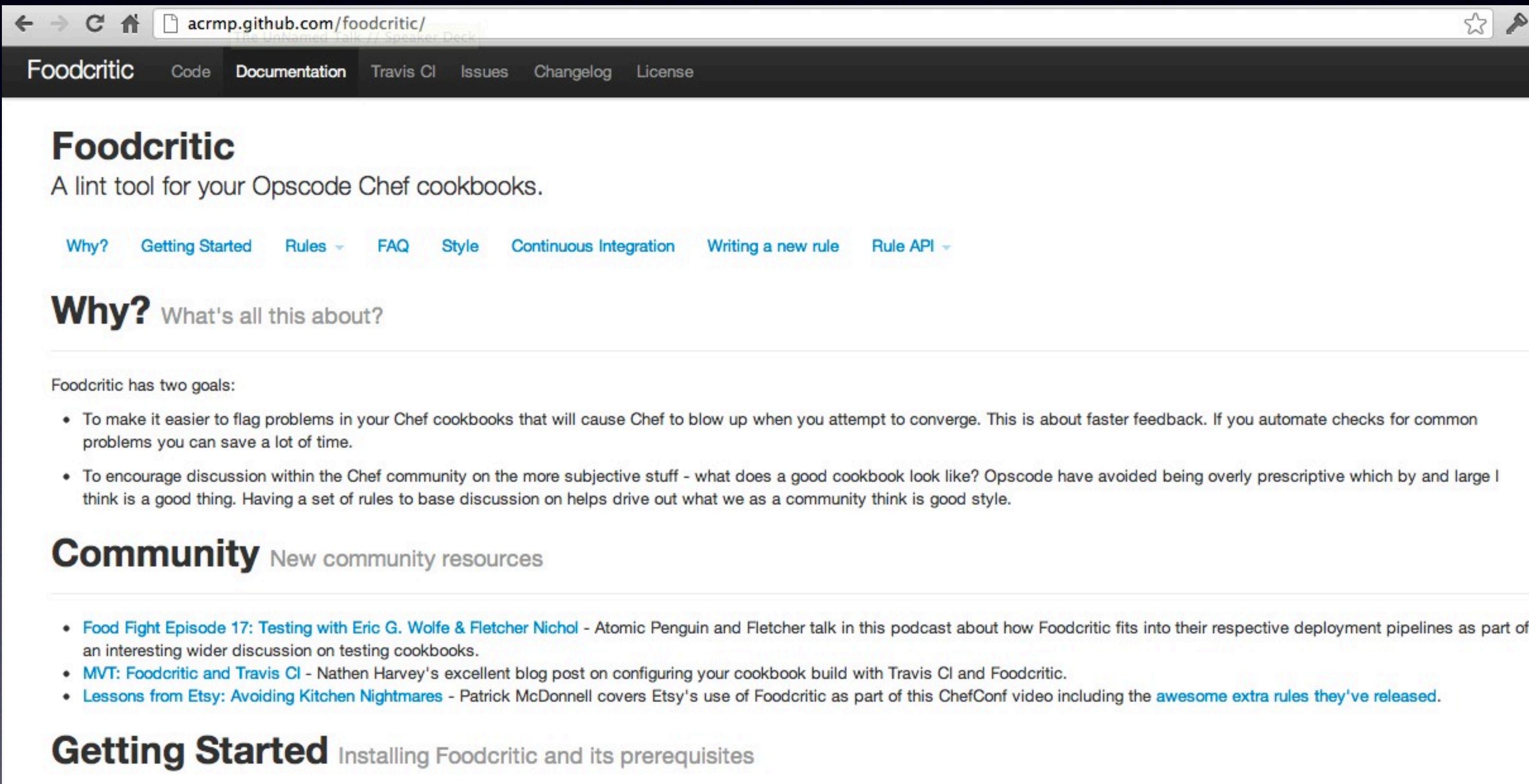
FATAL: ^



Lint Testing



Foodcritic



The screenshot shows a web browser displaying the Foodcritic project page on GitHub. The URL in the address bar is `acrmr.github.com/foodcritic/`. The page has a dark header with the GitHub logo and a search bar. Below the header is a navigation bar with links: Foodcritic (selected), Code, Documentation (highlighted in blue), Travis CI, Issues, Changelog, and License. The main content area has a dark background with white text. It features a large heading "Foodcritic" and a sub-heading "A lint tool for your Opscode Chef cookbooks.". Below this are several navigation links: Why?, Getting Started, Rules, FAQ, Style, Continuous Integration, Writing a new rule, and Rule API. A section titled "Why?" with the subtitle "What's all this about?" explains the tool's goals: to make it easier to flag problems in Chef cookbooks and encourage discussion within the Chef community. Another section titled "Community" lists "New community resources" including links to a podcast episode, a blog post, and a ChefConf video. A final section titled "Getting Started" provides information on installing Foodcritic and its prerequisites.

Foodcritic

A lint tool for your Opscode Chef cookbooks.

Why? Getting Started Rules FAQ Style Continuous Integration Writing a new rule Rule API

Why?

What's all this about?

Foodcritic has two goals:

- To make it easier to flag problems in your Chef cookbooks that will cause Chef to blow up when you attempt to converge. This is about faster feedback. If you automate checks for common problems you can save a lot of time.
- To encourage discussion within the Chef community on the more subjective stuff - what does a good cookbook look like? Opscode have avoided being overly prescriptive which by and large I think is a good thing. Having a set of rules to base discussion on helps drive out what we as a community think is good style.

Community

New community resources

- [Food Fight Episode 17: Testing with Eric G. Wolfe & Fletcher Nichol](#) - Atomic Penguin and Fletcher talk in this podcast about how Foodcritic fits into their respective deployment pipelines as part of an interesting wider discussion on testing cookbooks.
- [MVT: Foodcritic and Travis CI](#) - Nathon Harvey's excellent blog post on configuring your cookbook build with Travis CI and Foodcritic.
- [Lessons from Etsy: Avoiding Kitchen Nightmares](#) - Patrick McDonnell covers Etsy's use of Foodcritic as part of this ChefConf video including the awesome extra rules they've released.

Getting Started

Installing Foodcritic and its prerequisites

Foodcritic

Positives

Negatives

Foodcritic

Positives

- Easy to use

Negatives

Foodcritic

Positives

- Easy to use
- Extensible tests

Negatives

Foodcritic

Positives

- Easy to use
- Extensible tests
- Quickly gaining community adoption

Negatives

Foodcritic

Positives

- Easy to use
- Extensible tests
- Quickly gaining community adoption
- Best practices!

Negatives

Foodcritic

Positives

- Easy to use
- Extensible tests
- Quickly gaining community adoption
- Best practices!

Negatives

- Requires Ruby 1.9.2

Foodcritic

Positives

- Easy to use
- Extensible tests
- Quickly gaining community adoption
- Best practices!

Negatives

- Requires Ruby 1.9.2
- Some tests are too fragile

Foodcritic

Positives

- Easy to use
- Extensible tests
- Quickly gaining community adoption
- Best practices!

Negatives

- Requires Ruby 1.9.2
- Some tests are too fragile
- No built-in output extensibility

Foodcritic Demo

Foodcritic Demo

> gem install foodcritic

Foodcritic Demo

```
> gem install foodcritic  
> git clone git://github.com/etsy/foodcritic-rules.git
```

Foodcritic Demo

```
> gem install foodcritic  
> git clone git://github.com/etsy/foodcritic-rules.git  
> foodcritic -f any -I foodcritic-rules/rules.rb cookbooks/
```

Foodcritic Demo

```
> gem install foodcritic  
> git clone git://github.com/etsy/foodcritic-rules.git  
> foodcritic -f any -I foodcritic-rules/rules.rb cookbooks/
```

FC001: Use strings in preference to symbols to access node attributes: cookbooks/rsapps-base/attributes/default.rb:9

FC001: Use strings in preference to symbols to access node attributes: cookbooks/rsapps-base/attributes/default.rb:12

FC011: Missing README in markdown format: cookbooks/rsapps-base/README.md:1

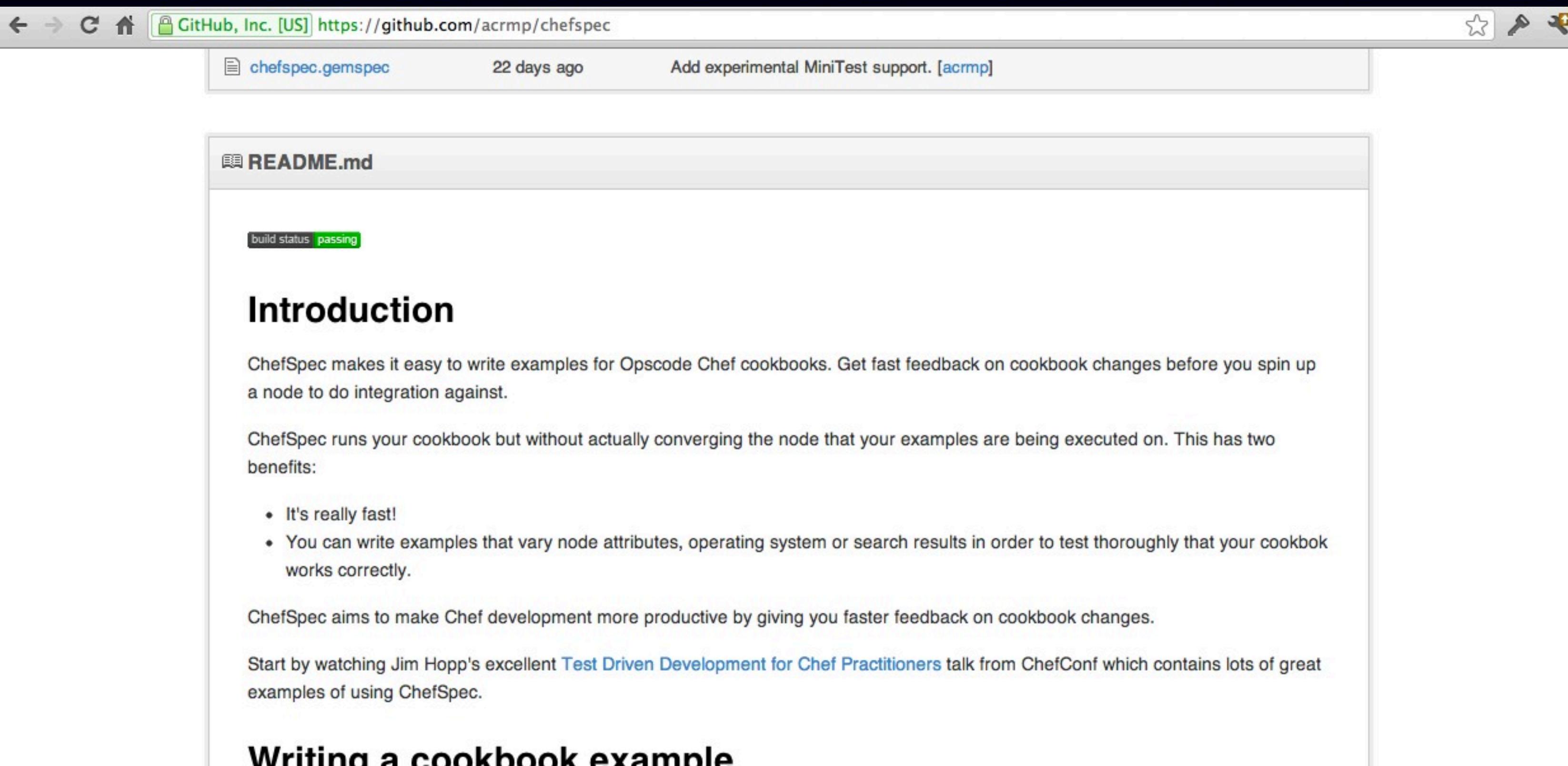
FC019: Access node attributes in a consistent manner: cookbooks/rsapps-base/attributes/default.rb:4



Unit Testing



ChefSpec



The screenshot shows a GitHub browser interface. The address bar indicates the URL is <https://github.com/acrmp/chefspec>. The main content area displays the README.md file of the repository. At the top of the README, there is a commit card for a file named `chefspec.gemspec`, which was committed 22 days ago by user `[acrmp]` with the message "Add experimental MiniTest support." Below the commit card, the `README.md` file is shown. It features a "build status" badge labeled "passing". The first section of the README is titled "Introduction". The text in this section states: "ChefSpec makes it easy to write examples for Opscode Chef cookbooks. Get fast feedback on cookbook changes before you spin up a node to do integration against." It then explains that ChefSpec runs your cookbook without converging the node, listing two benefits: "It's really fast!" and "You can write examples that vary node attributes, operating system or search results in order to test thoroughly that your cookbook works correctly." The text concludes by stating that ChefSpec aims to make Chef development more productive through faster feedback. A link to Jim Hopp's talk at ChefConf is provided for further learning.

Introduction

ChefSpec makes it easy to write examples for Opscode Chef cookbooks. Get fast feedback on cookbook changes before you spin up a node to do integration against.

ChefSpec runs your cookbook but without actually converging the node that your examples are being executed on. This has two benefits:

- It's really fast!
- You can write examples that vary node attributes, operating system or search results in order to test thoroughly that your cookbook works correctly.

ChefSpec aims to make Chef development more productive by giving you faster feedback on cookbook changes.

Start by watching Jim Hopp's excellent [Test Driven Development for Chef Practitioners](#) talk from ChefConf which contains lots of great examples of using ChefSpec.

Writing a cookbook example

ChefSpec

Positives

Negatives

ChefSpec

Positives

- Extends RSpec with Chef constructs

Negatives

ChefSpec

Positives

- Extends RSpec with Chef constructs
- Tests each recipe independently

Negatives

ChefSpec

Positives

- Extends RSpec with Chef constructs
- Tests each recipe independently
- Tests get packaged with your cookbook

Negatives

ChefSpec

Positives

- Extends RSpec with Chef constructs
- Tests each recipe independently
- Tests get packaged with your cookbook
- Pair with Fauxhai for extra awesome

Negatives

ChefSpec

Positives

- Extends RSpec with Chef constructs
- Tests each recipe independently
- Tests get packaged with your cookbook
- Pair with Fauxhai for extra awesome

Negatives

- Early stage of development

ChefSpec

Positives

- Extends RSpec with Chef constructs
- Tests each recipe independently
- Tests get packaged with your cookbook
- Pair with Fauxhai for extra awesome

Negatives

- Early stage of development
- RSpec's learning curve

ChefSpec

Positives

- Extends RSpec with Chef constructs
- Tests each recipe independently
- Tests get packaged with your cookbook
- Pair with Fauxhai for extra awesome

Negatives

- Early stage of development
- RSpec's learning curve
- Missing some desired Chef concepts

ChefSpec

Positives

- Extends RSpec with Chef constructs
- Tests each recipe independently
- Tests get packaged with your cookbook
- Pair with Fauxhai for extra awesome

Negatives

- Early stage of development
- RSpec's learning curve
- Missing some desired Chef concepts
- No built-in JUnit XML output in RSpec

ChefSpec Demo

ChefSpec Demo

```
> gem install chefspec
```

ChefSpec Demo

```
> gem install chefspec  
> knife cookbook create_specs -o cookbooks/ rsapps-base
```

ChefSpec Demo

```
> gem install chefspec  
> knife cookbook create_specs -o cookbooks/ rsapps-base  
> vim cookbooks/rsapps-base/spec/<recipes>_spec.rb
```

ChefSpec Demo

```
> gem install chefspec  
> knife cookbook create_specs -o cookbooks/ rsapps-base  
> vim cookbooks/rsapps-base/spec/<recipes>_spec.rb  
> cat cookbooks/rsapps-base/spec/clamav_spec.rb
```

ChefSpec Demo

```
> gem install chefspec  
> knife cookbook create_specs -o cookbooks/ rsapps-base  
> vim cookbooks/rsapps-base/spec/<recipes>_spec.rb  
> cat cookbooks/rsapps-base/spec/clamav_spec.rb
```

```
require "chefspec"

describe "rsapps-base::clamav" do
  let (:chef_run) { ChefSpec::ChefRunner.new.converge \
    "rsapps-base::clamav" }
  it "should install a specific version of rsapps-smtp-clamav" do
    chef_run.should install_package_at_version "rsapps-smtp-clamav",
      "0.97.2-1"
  end
end
```

Cont'd

Cont'd

```
> rspec cookbooks/rsapps-base
```

Cont'd

```
> rspec cookbooks/rsapps-base
```

```
.FF...F
```

Failures:

1) rsapps-base::clamav should install a specific version of rsapps-smtp-clamav

Failure/Error: chef_run.should install_package_at_version "rsapps-smtp-clamav",
expected chef_run: recipe[rsapps-base::clamav] to install package at version
"rsapps-smtp-clamav" and "0.97.2-2"

...

2) rsapps-base::default should include the external and internal recipes

Failure/Error: Unable to find matching line from backtrace

(#<Chef::Recipe:0x007f95d1434d38>).include_recipe("rsapps-base::bad_recipe")

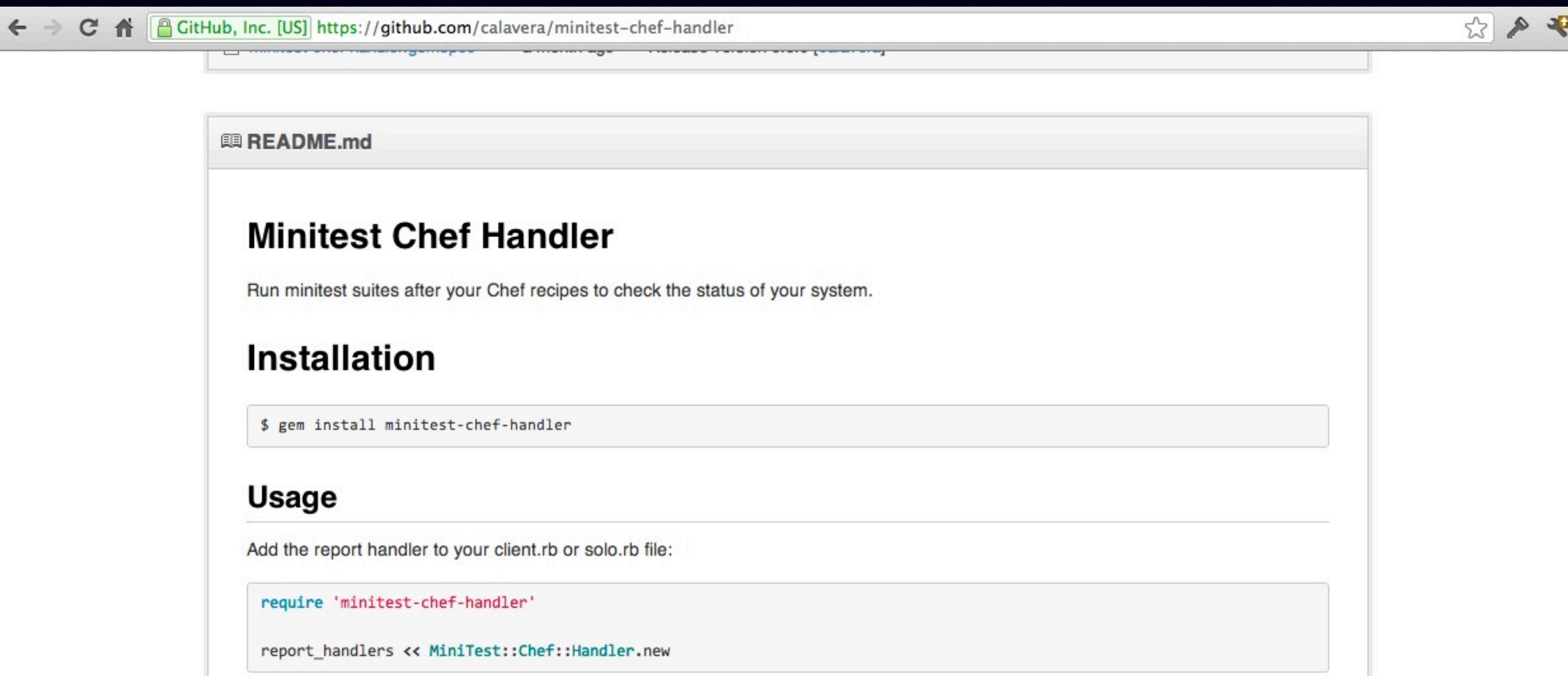
expected: 1 time

received: 0 times

...

Convergence Testing

Minitest-Chef-Handler



The screenshot shows a web browser displaying the GitHub README.md page for the `minitest-chef-handler` repository. The page contains the following content:

- Section Headers:** `README.md`, **Minitest Chef Handler**, **Installation**, **Usage**.
- Text:** Run minitest suites after your Chef recipes to check the status of your system.
- Code Block:** `$ gem install minitest-chef-handler`
- Text:** Add the report handler to your client.rb or solo.rb file:
- Code Block:** `require 'minitest-chef-handler'`
`report_handlers << MiniTest::Chef::Handler.new`

Minitest-Chef-Handler

Positives

Negatives

Minitest-Chef-Handler

Positives

Negatives

- Extends Minitest with Chef constructs

Minitest-Chef-Handler

Positives

- Extends Minitest with Chef constructs
- Tests recipe and cookbook interaction

Negatives

Minitest-Chef-Handler

Positives

- Extends Minitest with Chef constructs
- Tests recipe and cookbook interaction
- Tests the real results of a Chef run

Negatives

Minitest-Chef-Handler

Positives

- Extends Minitest with Chef constructs
- Tests recipe and cookbook interaction
- Tests the real results of a Chef run
- Tests get packaged with your cookbook

Negatives

Minitest-Chef-Handler

Positives

- Extends Minitest with Chef constructs
- Tests recipe and cookbook interaction
- Tests the real results of a Chef run
- Tests get packaged with your cookbook

Negatives

- Some overlap with ChefSpec

Minitest-Chef-Handler

Positives

- Extends Minitest with Chef constructs
- Tests recipe and cookbook interaction
- Tests the real results of a Chef run
- Tests get packaged with your cookbook

Negatives

- Some overlap with ChefSpec
- Pairing with Vagrant and automating still TBD

Minitest-Chef-Handler Demo

Minitest-Chef-Handler Demo

```
> gem install minitest-chef-handler
```

Minitest-Chef-Handler Demo

```
> gem install minitest-chef-handler  
> echo 'require "minitest-chef-handler"' >> solo.rb
```

Minitest-Chef-Handler Demo

```
> gem install minitest-chef-handler  
> echo 'require "minitest-chef-handler"' >> solo.rb  
> echo 'report_handlers << MiniTest::Chef::Handler.new' >> solo.rb
```

Minitest-Chef-Handler Demo

```
> gem install minitest-chef-handler  
> echo 'require "minitest-chef-handler"' >> solo.rb  
> echo 'report_handlers << MiniTest::Chef::Handler.new' >> solo.rb  
> vim cookbooks/rsapps-base/tests/<recipe>_test.rb
```

Minitest-Chef-Handler Demo

```
> gem install minitest-chef-handler  
> echo 'require "minitest-chef-handler"' >> solo.rb  
> echo 'report_handlers << MiniTest::Chef::Handler.new' >> solo.rb  
> vim cookbooks/rsapps-base/tests/<recipe>_test.rb  
> cat cookbooks/rsapps-base/tests/chef_key_cleanup_test.rb
```

Minitest-Chef-Handler Demo

```
> gem install minitest-chef-handler
> echo 'require "minitest-chef-handler"' >> solo.rb
> echo 'report_handlers << MiniTest::Chef::Handler.new' >> solo.rb
> vim cookbooks/rsapps-base/tests/<recipe>_test.rb
> cat cookbooks/rsapps-base/tests/chef_key_cleanup_test.rb
require "minitest-chef-handler"

class ChefKeyCleanupSpec < MiniTest::Chef::Spec
  include MiniTest::Chef::Assertions
  include MiniTest::Chef::Resources

  describe_recipe "rsapps-base::chef_key_cleanup" do
    it "deletes the Chef key file" do
      file("/etc/chef/validation.pem").wont_exist
      file("/home").wont_exist
    end
  end
end
```

Cont'd

Cont'd

> chef-solo

Cont'd

```
> chef-solo
```

```
[2012-08-15T13:53:34+00:00] INFO: Chef Run complete in 2.16697 seconds
```

```
[2012-08-15T13:53:34+00:00] INFO: Running report handlers
```

```
Run options: --seed 19184
```

```
# Running tests:
```

```
F.
```

```
Finished tests in 0.005494s, 364.0335 tests/s, 546.0502 assertions/s.
```

```
I) Failure:
```

```
test_0001_deletes the Chef key file(recipe::rsapps-base::chef_key_cleanup) [/tmp/vagrant-chef-1/chef-solo-1/cookbooks/rsapps-base/tests/chef_key_cleanup_test.rb:10]:  
Expected path '/home' not to exist
```

```
2 tests, 3 assertions, 1 failures, 0 errors, 0 skips
```

```
[2012-08-15T13:53:34+00:00] INFO: Report handlers complete
```

```
[2012-08-15T13:53:34+00:00] ERROR: Running exception handlers
```

```
[2012-08-15T13:53:34+00:00] ERROR: Exception handlers complete
```

```
[2012-08-15T13:53:34+00:00] FATAL: RuntimeError: MiniTest failed with 1 failure(s)
```

Other/Coming Soon

- TasteTest
- WhyRun mode
- ???

Helpful Links

Tools

- <http://acrmp.github.com/foodcritic/>
- <https://github.com/acrmp/chefspec>
- <https://github.com/calavera/minitest-chef-handler>

Tutorials

- <http://nathenharvey.com/blog/2012/05/29/mvt-foodcritic-and-travis-ci/>
- <http://hackers.mylookout.com/2012/04/cookout-at-lookout-with-chef/>

Internal Projects

- <https://github.rackspace.com/RSEmail/rsapps-base>

Other

- #chef on Freenode
- <https://twitter.com/bryanwb>
- <https://twitter.com/jtimberman>
- <http://foodfightshow.org/>