

# Chef & DevOps

Naoya Ito



# 入門 Chef Solo

INFRASTRUCTURE AS CODE



伊藤 直也

## 登録情報

フォーマット : Kindle版

ファイルサイズ : 374 KB

並列的なデバイスの利用: 無制限

出版社: 伊藤直也; 1版 (2013/3/11)

販売 : Amazon Services International, Inc.

言語 日本語

ASIN: B00BSPH158

おすすめ度 : [カスタマーレビューを書きませんか?](#)

Amazon ベストセラー商品ランキング: Kindleストア 有料タイトル - 2位 ([Kindleストア 有料タイトルのベストセラーを見る](#))

1位 - [Kindleストア](#) > [Kindle本](#) > [科学・テクノロジー](#) > [工学](#)

1位 - [Kindleストア](#) > [Kindle本](#) > [コンピュータ・IT](#) > [OS](#)

1位 - [本](#) > [コンピュータ・IT](#) > [OS](#) > [Linux](#)

[画像に対するフィードバックを提供する、またはさらに安い価格について知らせる](#)

**1人10冊  
ご購入ください**

# Chef

---

「クラウドでインフラがソフトウェアになったなら、プロビジョニングもコードで書けばいいじゃない」

```
package "supervisor" do
  action :install
end

service "supervisord" do
  supports :status => true, :restart => true, :reload => true
  action [ :enable, :start ]
end

template "supervisord.conf" do
  path "/etc/supervisord.conf"
  source "supervisord.conf.erb"
  owner "root"
  group "root"
  mode 0644
  notifies :reload, 'service[supervisord]'
end
```

# “Infrastructure as Code”

- プログラマブルなインフラ
  - Vagrant、Puppet、Chef
  - Amazon Cloud Formation、Amazon OpsWorks

The image shows a screenshot of the Opscode website homepage. At the top, there is a navigation bar with the Opscode logo (an orange infinity symbol) and the tagline "OPSCODE RULE THE CLOUD". To the right of the logo are links for "Customer Login", "Sign Up", "Account Management", and "Recover Password", along with social media icons for Facebook and Twitter. Below the navigation bar is a main menu with links for "Products & Services", "Solutions", "Customers", "Support", "About", "Newsroom", and "Community".

The main content area features a large banner for ChefConf 2013. On the left side of the banner, it says "The best open-source automation platform" and "Built to handle the biggest infrastructure challenges on the planet". On the right side, it says "#ChefConf 2013 April 24 - 26, 2013 San Francisco, California" and "Your Road to Automation". Below this, it says "Talk IT automation with Facebook, Adobe, Nordstrom and other industry leaders." and a "Learn More" button.

Below the banner, there are two columns of text. The left column starts with "Let's get to work." and a quote: "This stuff rocks, so bring us your". The right column is titled "Chef Products" and says "Chef is an open-source system integration framework built specifically for automating the cloud."

# DevOps

---



Dev and Ops

1. Automated infrastructure
  2. Shared version control
  3. One step build and deploy
  4. Feature flags
  5. Shared metrics
  6. IRC and IM robots
- 
1. Respect
  2. Trust
  3. Healthy attitude about failure
  4. Avoiding Blame

# Chef によるオートメーション

```
🔗 sandbox.bloghackers.net [master] knife solo cook sandbox
Running Chef on sandbox...
Checking Chef version...
Syncing kitchen...
Adding patches...
Syncing solo config...
Running Chef...
Starting Chef Client, version 11.4.0
Compiling Cookbooks...
Converging 40 resources
Recipe: sandbox::zoneinfo
  * cookbook_file[/etc/localtime] action create (up to date)
  * cookbook_file[/etc/sysconfig/clock] action create (up to date)
Recipe: yum::epel
  * yum_key[RPM-GPG-KEY-EPEL-6] action add (up to date)
  * yum_repository[epel] action add (up to date)
  * yum_repository[epel] action update (up to date)
Recipe: build-essential::default
  * package[gcc] action install (up to date)
  * package[gcc-c++] action install (up to date)
  * package[kernel-devel] action install (up to date)
```

# Chefの守備範囲

---

- インスタンス立ち上げ後、アプリケーションデプロイ前
  - × インスタンスを立ち上げる
  - ○ ネットワークを設定する
  - ○ パッケージ入れる
  - ○ ユーザー作る
  - ○ サービス設定して起動する
  - ○ t1.micro にスワップ作る
  - △ Railsアプリケーションをデプロイする

# Chef is not only 自動化ツール

---

- Chef = システムを「あるべき状態」に収束させるフレームワーク
  - サーバー構築を自動化する = 目的の状態にする
    - ↑こればかりに着目しがち
  - 中途半端な状態のサーバーをあるべき状態にする
  - 過去のAMIスナップショットから作ったインスタンスを、今現在の状態に追いつかせる
  - 障害対応で一時的に設定を変えた各種状態をもとに戻す
  - 共有している個人開発用VMに、他の開発者が変更した状態を適用する

# Chefのワークフロー

---

- 「レシピ」を Ruby で書く
- knife-solo などで行  
  - rysnc でレシピを転送
  - chef-solo がそのレシピを実行
  - サーバー構成が定義した状態に収束
- **git commit**

# Chef + Github

「プロビジョニングをコードで書いたならバージョン管理して Social Coding  
すればいいじゃない」

naoya / [sandbox.bloghackers.net](#) Pull Request

Code Network Pull Requests 0 Issues 0

branch: master Files Commits Branches 1

[sandbox.bloghackers.net](#) / [site-cookbooks](#) / [sandbox](#) / [recipes](#) / [atig.rb](#)

naoya 24 days ago install atig

1 contributor

file | 24 lines (20 sloc) | 0.399 kb

[sandbox.bloghackers.net](#) / Commit History Keyboard shortcuts available

**Apr 21, 2013**

**Fix bloghackers.net 404** 9c64c7b89e  
naoya authored 4 days ago [Browse code](#)

**Apr 19, 2013**

**Update nginx.conf for hosting bloghackers.net** 88938ca88a  
naoya authored 5 days ago [Browse code](#)



# インフラを バージョン管理



# インフラを Social Coding

# serverspec

---

「コードで書いたんだったらテストも書けばいいじゃない」

```
require 'spec_helper'

describe 'nginx' do
  it { should be_installed }
  it { should be_enabled }
  it { should be_running }
end

describe 'port 80' do
  it { should be_listening }
end

describe '/etc/nginx/nginx.conf' do
  it { should be_file }
  it { should contain "server_name md2inao.bloghackers.net" }
end
```

# Vagrant + Chef + serverspec ⇨ TDD

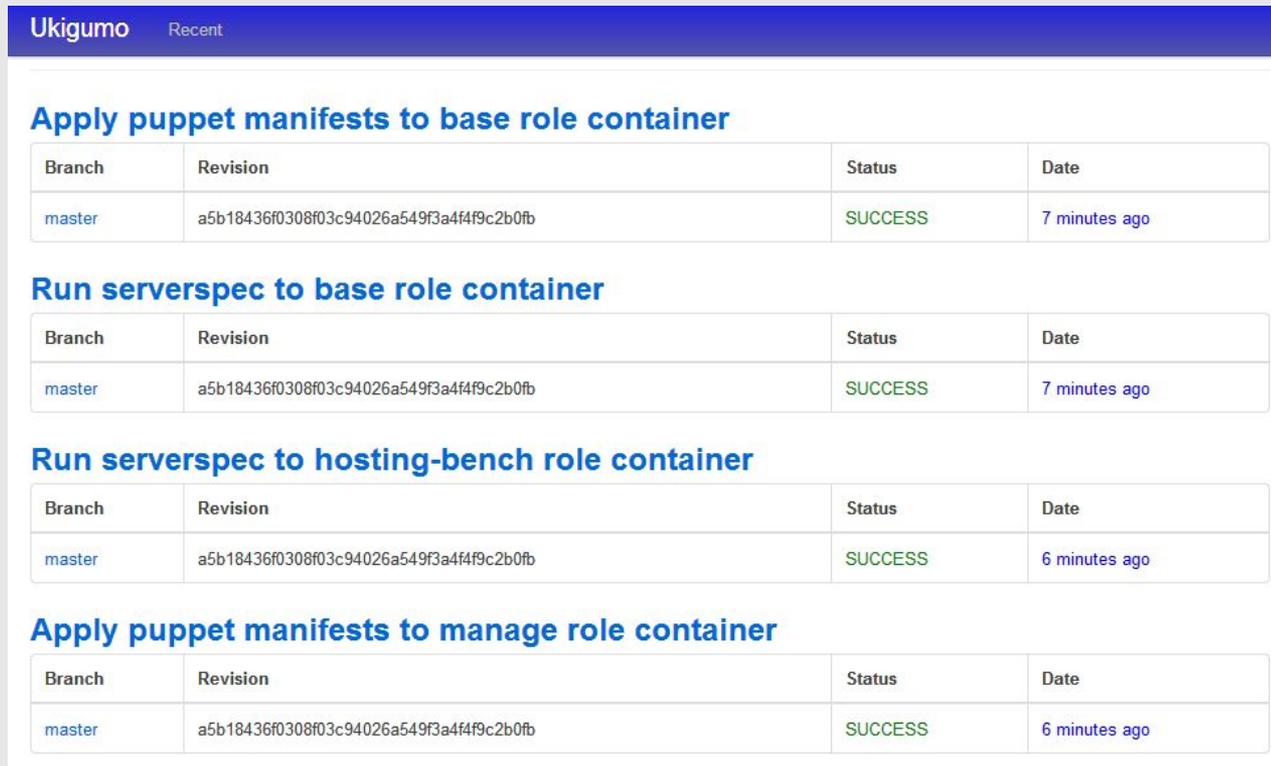
---

```
🌩 sandbox.bloghackers.net [master] rake spec
/Users/naoya/.rbenv/versions/1.9.3-p362/bin/ruby -S rspec spec
spec/sandbox/lig_spec.rb spec/sandbox/md2inao_spec.rb spec/san
ec/sandbox/suervisord_spec.rb spec/sandbox/swap_spec.rb spec/s
--options spec/spec.opts
.....

Finished in 6.22 seconds
25 examples, 0 failures
```

# インフラのCI

「テスト書いたんだったら継続的インテグレーションすればいいじゃない」



The screenshot displays the Ukigumo CI interface with a blue header bar containing the name 'Ukigumo' and the word 'Recent'. Below the header, there are four job entries, each with a title and a corresponding table of results. All jobs show a 'SUCCESS' status.

| Ukigumo Recent   |  |         |               |
|--|--|---------|---------------|
| <b>Apply puppet manifests to base role container</b>   |  |         |               |
| Branch   | Revision                                 | Status  | Date          |
| master   | a5b18436f0308f03c94026a549f3a4f4f9c2b0fb | SUCCESS | 7 minutes ago |
| <b>Run serverspec to base role container</b>           |  |         |               |
| Branch   | Revision                                 | Status  | Date          |
| master   | a5b18436f0308f03c94026a549f3a4f4f9c2b0fb | SUCCESS | 7 minutes ago |
| <b>Run serverspec to hosting-bench role container</b>  |  |         |               |
| Branch   | Revision                                 | Status  | Date          |
| master   | a5b18436f0308f03c94026a549f3a4f4f9c2b0fb | SUCCESS | 6 minutes ago |
| <b>Apply puppet manifests to manage role container</b> |  |         |               |
| Branch   | Revision                                 | Status  | Date          |
| master   | a5b18436f0308f03c94026a549f3a4f4f9c2b0fb | SUCCESS | 6 minutes ago |

Ukigumo + serverspec + Puppet による継続的インテグレーション

via <http://mizzy.org/blog/2013/03/27/1/>

# DevOpsのいま

- 「インフラも Social Coding」
  - Infrastructure as Code
  - Github
  - TDDと継続的インテグレーション



インフラの構築も Pull Request で、Social Coding で。  
「各個人の手順書」という暗黙知から  
「共有されたソースコード」という形式知へ