Rubyの世界の継続的デリバリ

Continuous Delivery in Ruby World

柴田 博志

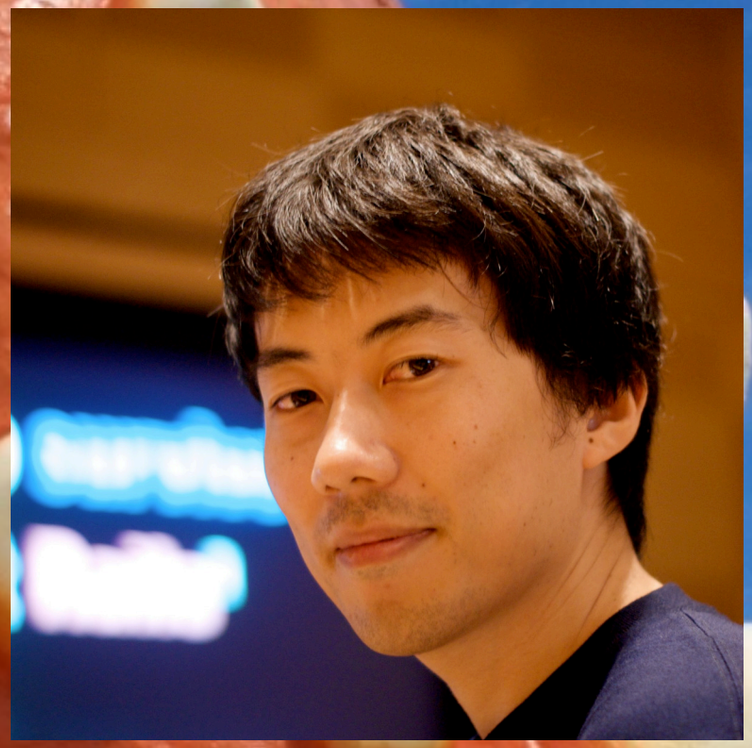
SHIBATA Hiroshi

paperboy&co.
asakusa.rb

paperboy&co., Inc.

発表場所 Sapporo Ruby Kaigi 2012　2012-09-16(Sun)

self



SHIBATA Hiroshi



id: hsbt



asakusa.rb



tDiary
いいね！ 159人

「いいね！」しました　メッセージ

ソフトウェア
歴史と実績をもつWeb日記システム。公式サイトはこちら→
http://www.tdiary.org/

基本データ　写真　いいね！　イベント

ハイライト

投稿　写真・動画

何か書く...

24人 友達 がtDiaryについて「いいね！」と言っています

+15

tDiarySさんがリンクをシェアしました。
8月31日

tDiarySさんがリンクをシェアしました。

tDiary プラグインの更新チェッカー - まちゅダイアリー(2012-08-18)

- category_autocompleteプラグインを公開しまし

Facebook広告でページを宣伝

tDiary

歴史と実績をもつWeb日記システム。公式サイトはこちら→
http://www.tdiary.org/

いいね！ · 柴田 博志さんが「いいね！」

8月
2012年
2011年
設立

http://www.facebook.com/tDiary

paperboy&co.

【採用情報】Webデザイナーの募集を開始しました。

メディア公式
アカウント一覧

- 会社概要
  - VISION 企業理念
  - HISTORY 会社沿革
  - ACCESS アクセス
  - PRIVACY プライバシーポリシー

もっとブログを楽しもう!
JUGEM PLUS
月額315円でワンランク上のブログ!
■うれしいアクセス制限付き
■容量たっぷり!無制限!
■複数画像アップロード!1点10MB
■もちろん!広告非表示
■スパム対策もバッチリ!
■携帯・スマートフォン対応!
■アクセス解析機能

企業理念 | COMPANY VISION



企業理念「もっとおもしろくできる」

すべての企業活動において、他社よりもおもしろいものを目指します。

ミッション「より多くの人に情報発信する喜びを提供する」

当社は、便利でリーズナブルなサービスを運営し、より多くの人に情報発信する喜びを提供してまいります。また、ユーザーの自己表現を支援するために、表現者のクリエイティビティを最大に引き出せる、最高の環境を創造してまいります。

スローガン「ネットのおいしいところ」

当社は、他社よりもおもしろいサービスを提供するだけでなく、収益性の高い事業の展開をすすめ、数あるインターネットサービス群の中で、すべてのステークホルダーが「[illegible]」と言えるものを手がけます。そのために、クリエイティビ[illegible]し、高い付加価値をもったサービスを手がけてまいりま

paperboy&co.

昼寝ができる!
ユニーク制度で効率UP

眠いのを我慢して仕事しても
全然効率的でない

技術基盤整備エンジニア

2012-09-15

「非Rubyな会社で仕事にRubyを持ち込むための5つの方法」資料 #sprk2012

札幌Ruby会議2012参加のため、9/13から札幌にいます。大規模なRubyのイベントに参加するのは初めてなのですが、せっかくなのでなにかおしゃべりしようと思ってトークに応募したらLTは通ったので、「非Rubyな会社でRubyで仕事にRubyを持ち込むための5つの方法」ちょっと話してきました。



Profile



id:antipop PRO

A blog about anything that I can feel delirious from

+ Subscribe 108

Friend request

Search

Search

Links

Help

Hatena Blog

Notices


SQALE
ホーム
Sqale について
ご利用料金
サポート
ログイン
paperboy&co.
手っ取り早く。
やりたいことだけ。
Quick deploy & Flexible resources.
B! 265
Tweet 918
いいね! 801
いますぐはじめる
OR
Log in with Twitter
Log in with Facebook
Log in with GitHub
コードを書く時間を1秒でも長く。
Sqaleは、開発者のためのホスティングサービスです。
Ruby
Sinatra
RAILS
Ruby on Rails
Resources
Overload
deploy
SQALE
More code!
VPS
set up & wait.....
Ruby on Rails 対応。月額 940円から。
必要になったら、そのときに考えよう。
あなたのアプリケーションがいつヒットするか、事前に見極めるのは難しいことです。
Sqale はアプリケーションの成長にあわせて、いつでもサーバーリソースの追加や削除ができます。 アクセス増による負荷に悩むのは、後回しにしましょう。
コード!コード!もっとコード!
VPS のようにサーバーの構築や運用に時間を費やすのは、あなたがしたいことですか? あなたがコードを書く時間が少しでも増えるように、めんどうな部分は Sqale が引き受けます。 あなたはいつもの画面からアプリケーションをデプロイするだけです。
sqale.jp

Gosuke Miyashita

SEP 15TH, 2012

「Sqale の裏側」資料 at 札幌 Ruby 会議 2012

札幌 Ruby 会議 2012 、まだ会期中ですが、一足お先に帰ってきました。(大学の必修科目である、基礎数学のスクーリング授業が今日から3日間あるため。)

YAPC::Asia 2011 の個人スポンサーTシャツを着て参加したのですが、Rubyist の皆さんの心が広いためか、宮川神のご加護があったからか、殴られたりせずに済んで一安心。

ペパボ は徐々に Ruby にスイッチしてきており、Ruby が動く PaaS である **Sqale** もリリースされたことから、宣伝も兼ねて **Sqale の裏側** というタイトルで発表させてもらいました。

Sqale は、 Ruby が動く PaaS でもあるし、Sqale 自体の開発も、フロントもバックもバリバリ Ruby が使われているのですが、Ruby の話はほとんどしない、出てくるコードも Lua と C の方が Ruby よりも多い、というトーク内容でしたが、技術的にはなかなかおもしろい内容になってると思いますので、PaaS の裏側、特にインフラ寄りの部分ってどうなってるんだろう、と興味のある方は、ぜひ資料をご覧になってみてください。

発表の会場には、Sqale の競合である Heroku の中の人でもあり、Ruby のパパでもある Matz さんがいらっしゃったので、ノートPCに張った Heroku ステッカーを見せて、しっかりと媚びを売ることも忘れませんでした。

YAPC::Asia Tokyo 2012 の前夜祭でも、同じ内容でお話させていただきます。Perl の方は Ruby 以上に関係がなく、Perl の話はまったくしませんし、基本的には札幌 Ruby 会議で話した内容とほとんど同じですので、みなさん僕の話は聞かずに、Perl Mongers との交流を楽しむことをお勧めします。

30 days Album™
We host your memories

なくしたくない思い出。全部、お預かりします。

オンラインアルバム + フォトストレージサービス・30days Album™

これまでに 207,779 人のお客様の 132,837,520 枚の写真をお預かりしています。

30days Album™ をはじめる
無料ユーザー登録ですぐにご利用いただけます

30days Album™ について詳しく

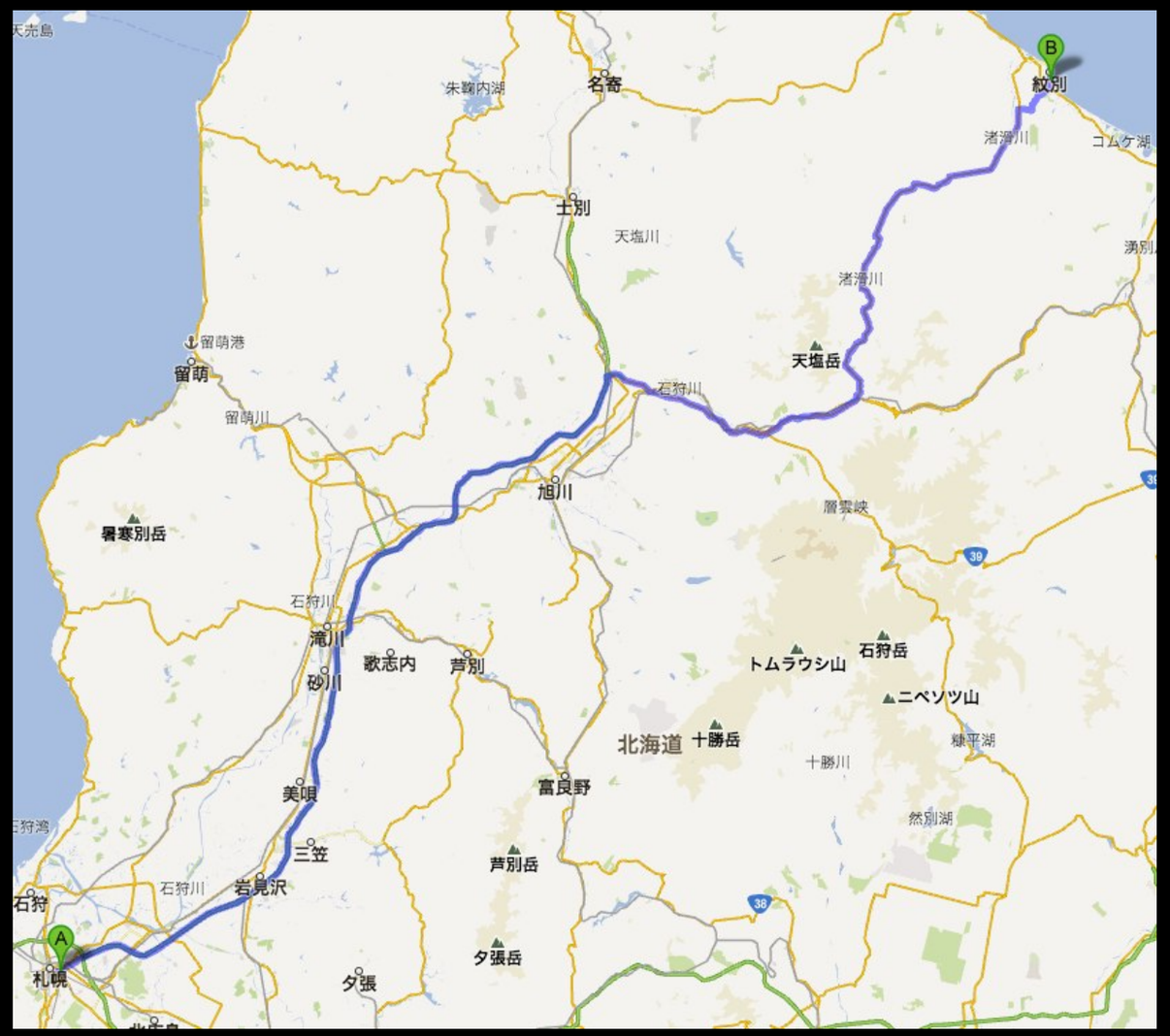


30d.jp

写真を見せる、写真を残す。30days Album™ で出来ること。

coming soon...





紋別市
流氷/ガリンコ号
カニ

Rubyの世界の継続的デリバリ

Continuous Delivery in Ruby World

柴田 博志

SHIBATA Hiroshi

paperboy&co.
asakusa.rb

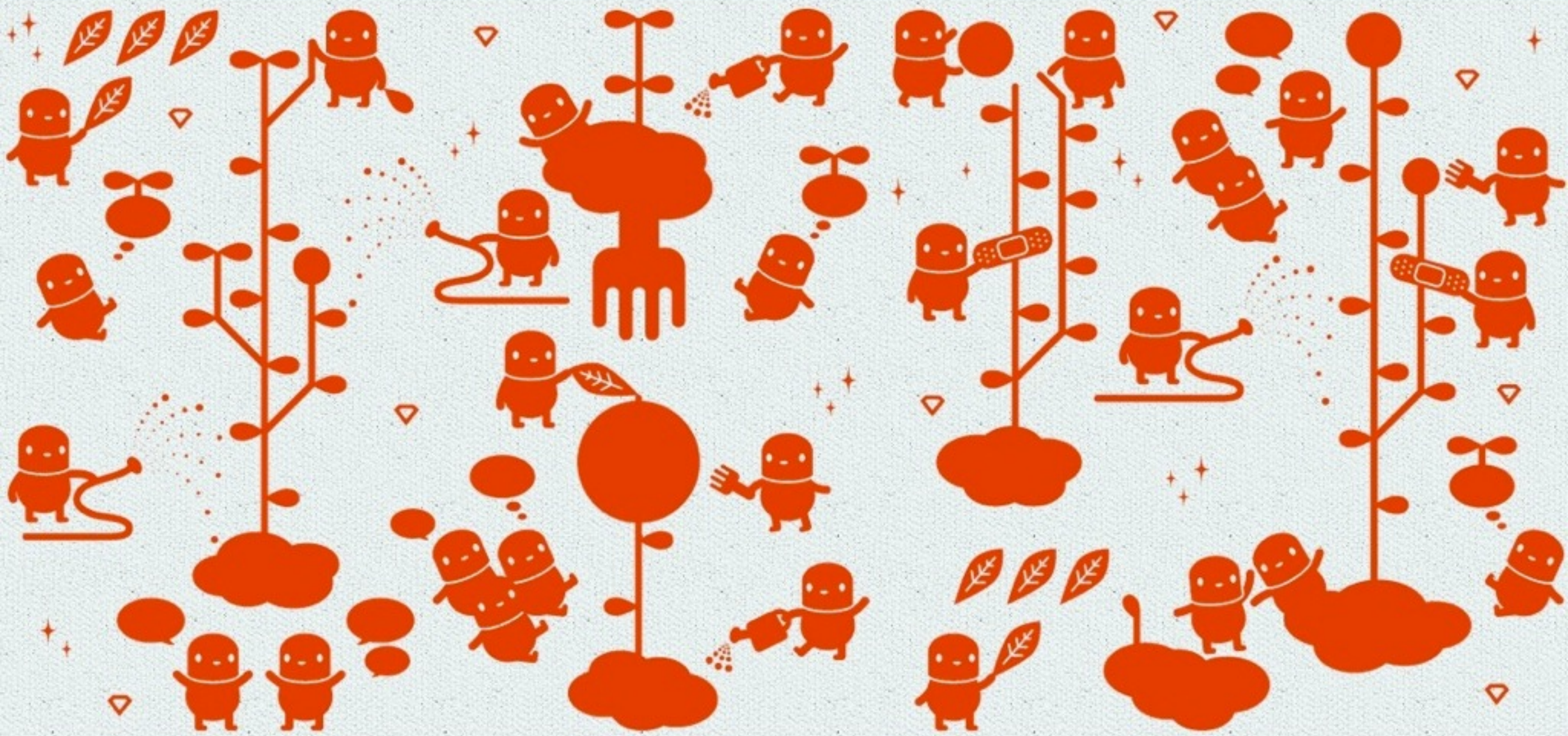
paperboy&co., Inc.

発表場所 Sapporo Ruby Kaigi 2012　2012-09-16(Sun)

Sapporo RubyKaigi 2012

We Code.

September 14-16th, 2012
at Sapporo Business Innovation Center



Keynote:

- Yukihiro Matsumoto
- Aaron Patterson

Show Skitch

日本語 | English

rence Registration

*3日間通し券

REGISTER

We Code

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Principles behind the Agile Manifesto

We follow these principles:

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Working software is the primary measure of progress.

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

The Addison-Wesley Signature Series

A Martin Fowler Signature Book

Continuous Delivery

Reliable Software Releases through Build, Test, and Deployment Automation

Jez Humble,
David Farley



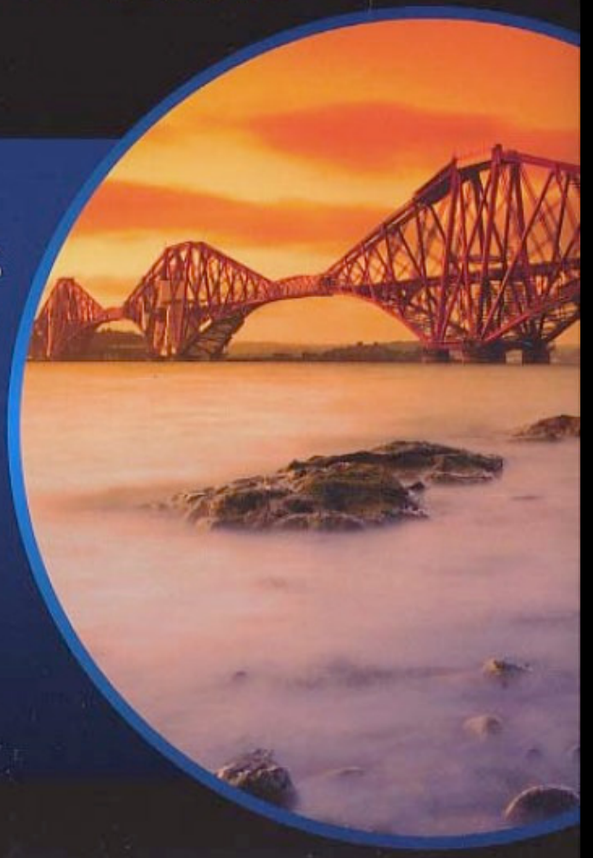
Continuous Delivery

継続的デリバリー

信頼できるソフトウェアリリースのための
ビルド・テスト・デプロイメントの自動化

Jez Humble, David Farley 著
和智右桂、高木正弘 訳

いつまで手動で
デプロイして
いるんですか?



ASCII



セコン
@hotchpotch

フォロー中

薄くて要点を絞った良書は、知識が深くかつ文章がうまい人じゃ無いと書けないので数少ないのが悲しい...。一時間ぐらいでぺろっと読み終わる良い技術書たくさん出て欲しいですね。

返信 リツイート お気に入りに登録

8 件のリツイート | 5 お気に入り



2012年8月21日 - 10:27 · このツイートをサイトに埋め込む

@hotchpotchさんへ返信



Taro Minowa Higepon @higepon 8月21日
@hotchpotch そして自分は分厚い方の本を読んで差別化ですね。分かります。
詳細



seki at druby.org @m_seki 8月21日
@hotchpotch 私の本は薄いよ
詳細

セコン @hotchpotch 8月21日
@higepon 自慢じゃ無いですけど厚い本読み終える確率10%ぐらいです！！！
詳細

セコン @hotchpotch 8月21日
@m_seki 咳さんのほんはぜんぶ読み終えてます!
詳細

継続○○が大事な二つの理由

12 日

意識が低くてもOKな仕組み

The RSpec Book

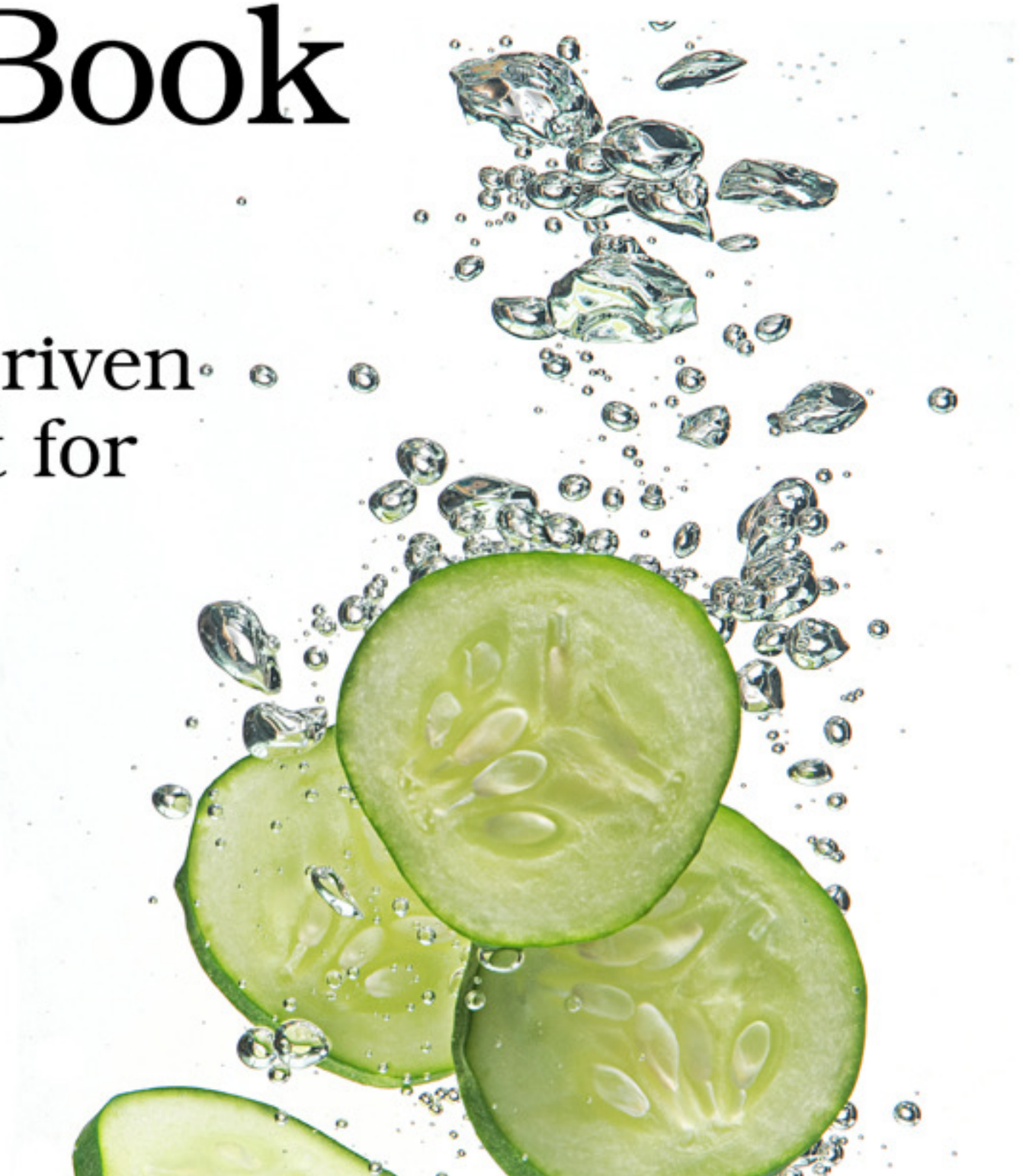
Behaviour-Driven Development with RSpec, Cucumber, and Friends

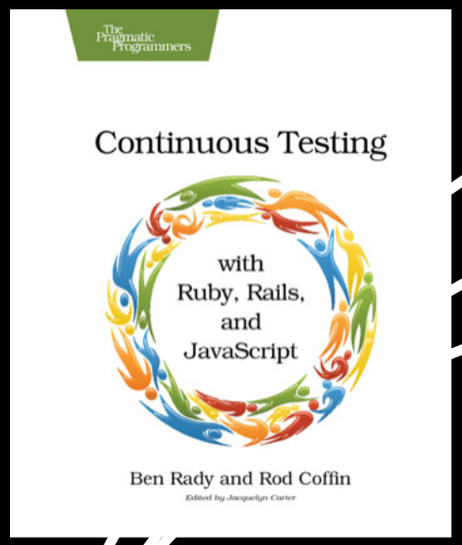


David Chelimsky
with Dave Astels, Zach Dennis, Aslak Hellesøy, Bryan Helmkamp, and Dan North

The Cucumber Book

Behaviour-Driven Development for Testers and Developers





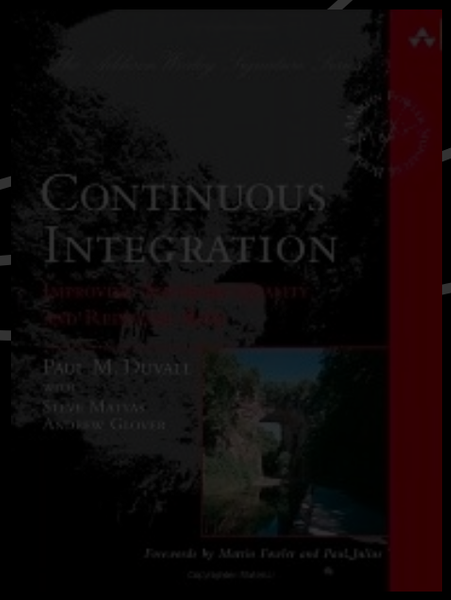
開発者

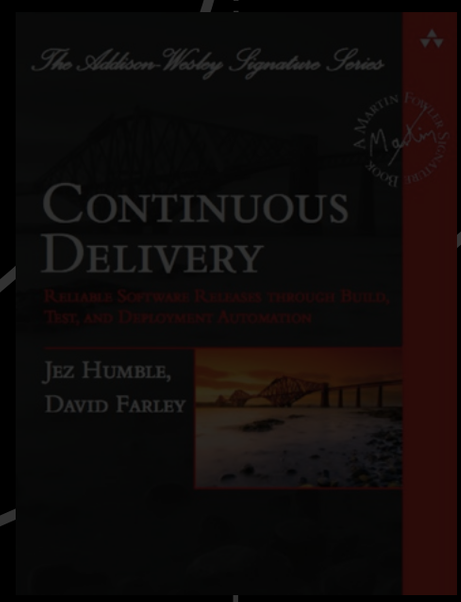
運用

QA

PL 他

ユーザー





Automation

Jenkins

検索

Jenkins

開発者

ビルド履歴

ビルドキュー

なし

ビルド実行状態

#	状態	
1	待機中	
2	待機中	

All

S	W	名前 ↓	最新の成功ビルド	最新の失敗ビルド	ビルド所要時間
		tdiary-contrib-1.8.7	2 日 23 時間 前 (#46)	–	17 秒
		tdiary-contrib-1.9.2	2 日 23 時間 前 (#46)	–	13 秒
		tdiary-contrib-1.9.3	2 日 23 時間 前 (#45)	–	19 秒
		tdiary-contrib-trunk	27 日 前 (#46)	2 日 23 時間 前 (#47)	16 秒
		tdiary-core-1.8.7	9 日 0 時間 前 (#96)	–	2 分 22 秒
		tdiary-core-1.9.2	9 日 0 時間 前 (#93)	–	1 分 56 秒
		tdiary-core-1.9.2-selenium	15 日 前 (#76)	9 日 0 時間 前 (#78)	5 分 50 秒
		tdiary-core-1.9.3	9 日 0 時間 前 (#96)	–	1 分 47 秒
		tdiary-core-1.9.3-selenium	15 日 前 (#74)	5 日 21 時間 前 (#77)	5 分 44 秒
		tdiary-core-trunk	1 ヶ月 5 日 前 (#93)	6 日 0 時間 前 (#100)	2 分 5 秒
		tdiary-core-trunk-selenium	2 ヶ月 9 日 前 (#65)	1 ヶ月 5 日 前 (#77)	6 分 37 秒

アイコン: S M L

凡例 RSS 全ビルド RSS 失敗ビルド RSS

Help us localize this page

更新: 2012/07/29 12:11:49

Jenkins

最近

ekmett/lens #394
処理時間: 1 min 55 sec, 終了時刻: -

ent-io/melang #17
処理時間: 6 min 50 sec, 終了時刻: 3 minutes ago

kleberpinel/kleberetalita #10
処理時間: 1 min 18 sec, 終了時刻: 4 minutes ago

weavejester/ns-tracker #3
処理時間: 1 min 43 sec, 終了時刻: 5 minutes ago

marionettejs/backbone.marionette #8
処理時間: 1 min 24 sec, 終了時刻: 6 minutes ago

ruby/ruby #2460
処理時間: 4 min 30 sec, 終了時刻: 4 minutes ago

docunext/svxbox #17
処理時間: 27 sec, 終了時刻: 9 minutes ago

LeventErkok/sbv #82
処理時間: 2 min 34 sec, 終了時刻: 11 minutes ago

dblock/jna #26
処理時間: 52 sec, 終了時刻: 18 minutes ago

ekmett/lens

Lenses, Folds, and Traversals

最新 | ビルド履歴 | Pull Requests | ブランチまとめ

ビルド	394	コミット	8b4c82a (master)
終了時刻	-	比較	49492a3bb8ee...8b4c82a106ae
処理時間	1 min 55 sec	制作者	Edward Kmett
		コミット者	Edward Kmett
メッセージ	hunit cleanup		
設定	-		

```
Using worker: jvm-otp2.worker.travis-ci.org:jvm-5

$ cd ~/builds
$ git clone --depth=100 --quiet git://github.com/ekmett/lens.git ekmett/lens
$ cd ekmett/lens
$ git checkout -qf 8b4c82a106ae478921fc1961c78c1adbb063fb22
$ ghc --version
The Glorious Glasgow Haskell Compilation System, version 7.4.1
$ cabal --version
cabal-install version 0.14.0
using version 1.14.0 of the Cabal library
$ cabal update
Downloading the latest package list from hackage.haskell.org
$ cabal install --only-dependencies --enable-tests --enable-benchmarks --force-reinstall
Resolving dependencies...
Warning: The following packages are likely to be broken by the reinstalls:
regex-posix-0.95.1
regex-compat-0.95.1
Continuing even though the plan contains dangerous reinstalls.
Downloading abstract-deque-0.1.6...
Configuring abstract-deque-0.1.6...
Building abstract-deque-0.1.6...
Preprocessing library abstract-deque-0.1.6...
[1 of 4] Compiling Data.Concurrent.Deque.Class ( Data/Concurrent/Deque/Class.hs,
dist/build/Data/Concurrent/Deque/Class.o )
[2 of 4] Compiling Data.Concurrent.Deque.Reference ( Data/Concurrent/Deque/Reference.hs,
dist/build/Data/Concurrent/Deque/Reference.o )
[3 of 4] Compiling Data.Concurrent.Deque.Tests ( Data/Concurrent/Deque/Tests.hs,
dist/build/Data/Concurrent/Deque/Tests.o )
[4 of 4] Compiling Data.Concurrent.Deque.Reference.DequeInstance (
Data/Concurrent/Deque/Reference/DequeInstance.hs, dist/build/Data/Concurrent/Deque/Reference/DequeInstance.o )
```

Travis

何をつくる

インセプション
デッキ

ストーリー
収集

スプリント0

完成

リリース
スプリント

スプリント

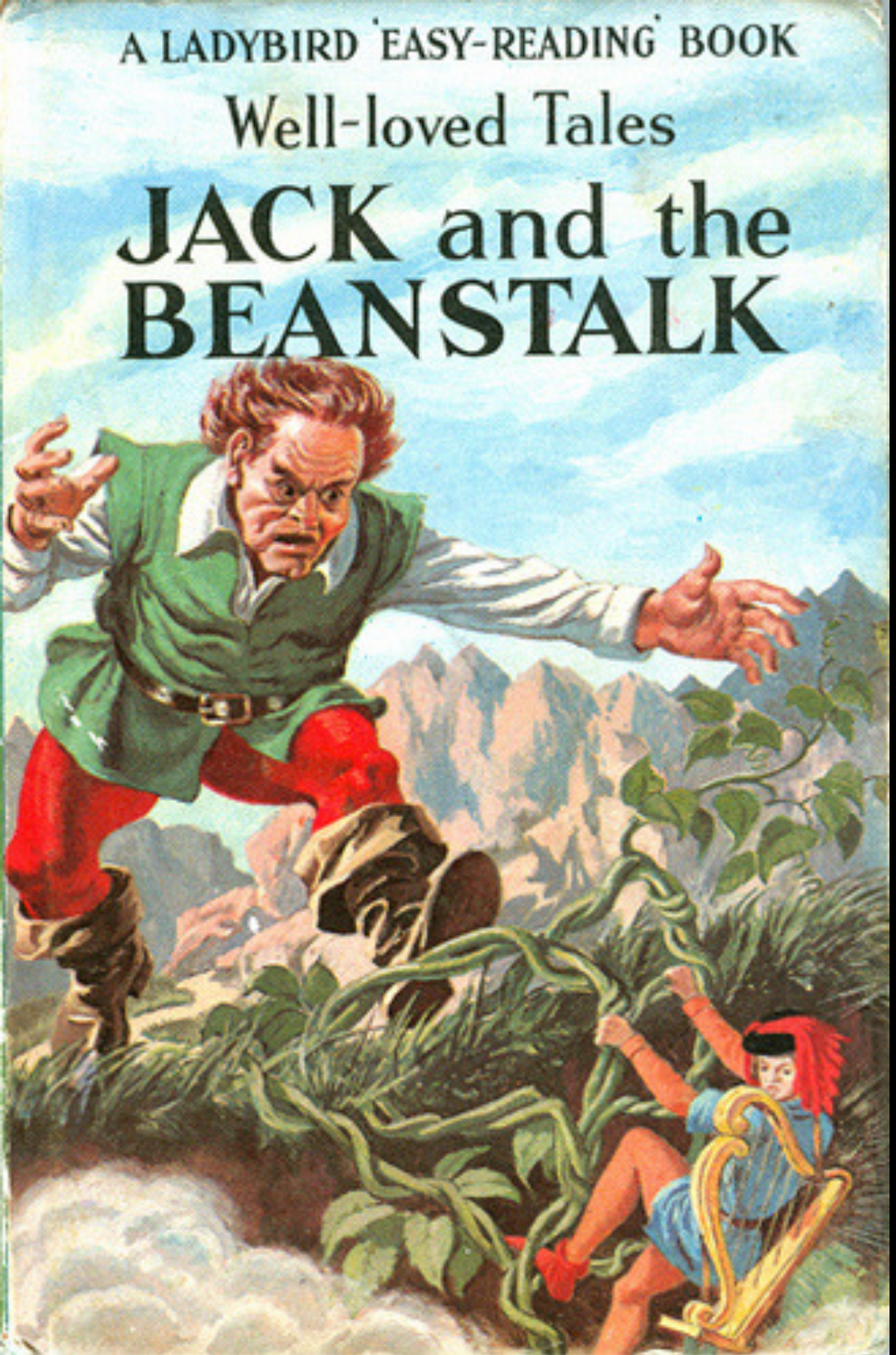
いつまでに
出荷するか

https://speakerdeck.com/u/nawoto/p/basic-of-basics-of-agile-development

rintakumpu - fl

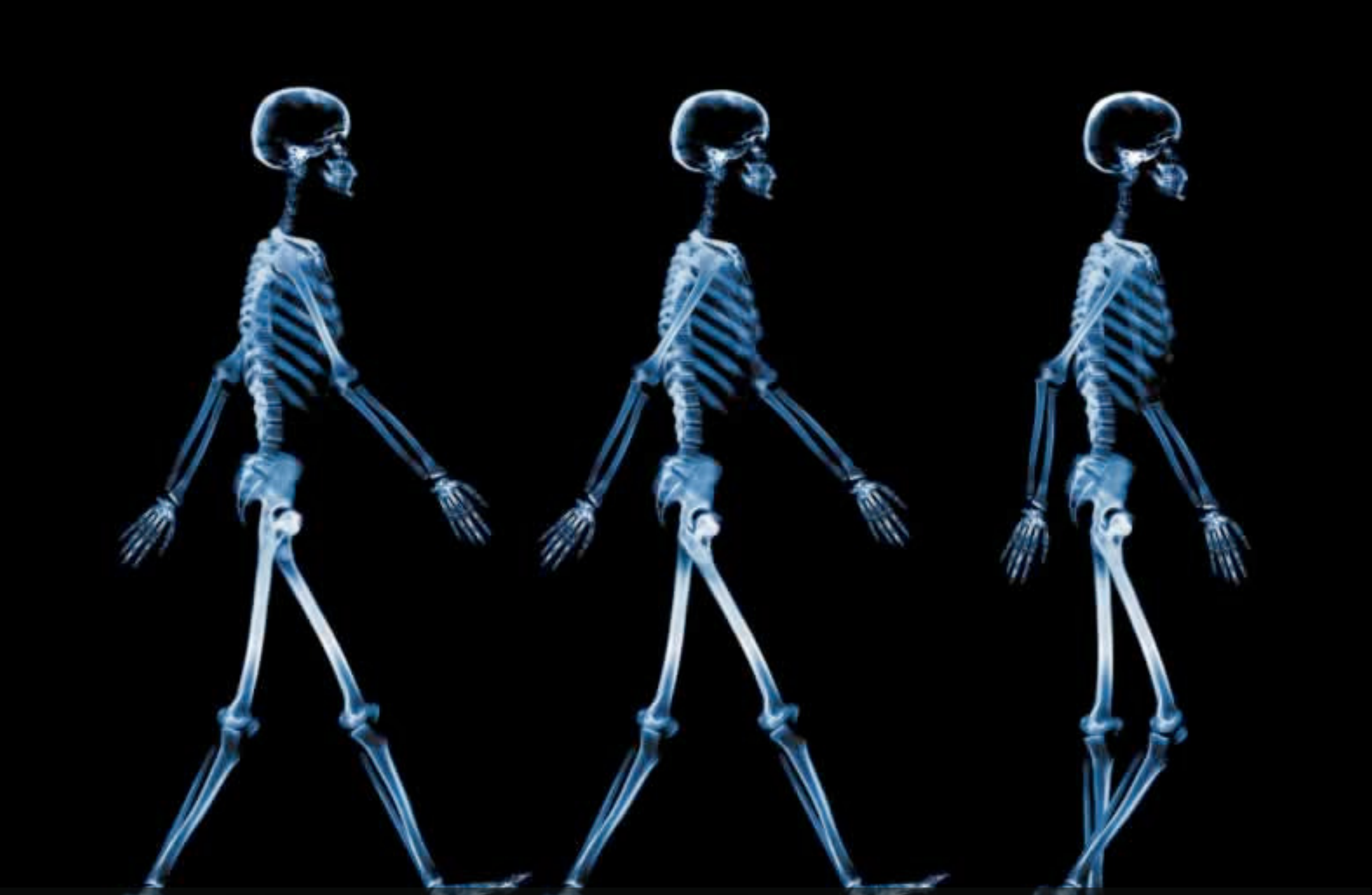
品質管理
よくある
誤解

結合テストで品質を上げるとか言う人がいる



ジャックと豆の木のように何かが起きて品質を上がるわけではない

芽を幹にするように



walking skelton

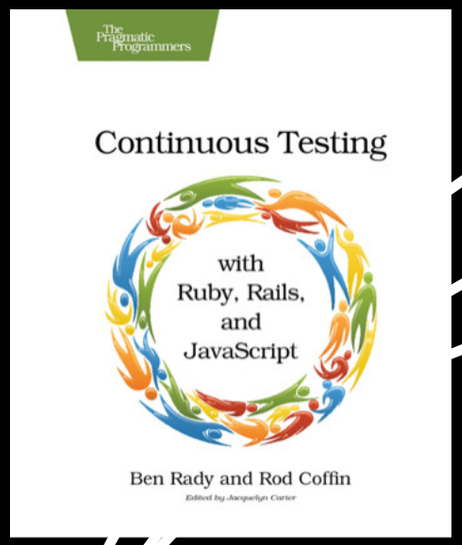
意識が低くてもOKな仕組み



八雲町
ハーベスター・八雲

2 3 E

確実なフィードバックと改善を行える仕組み



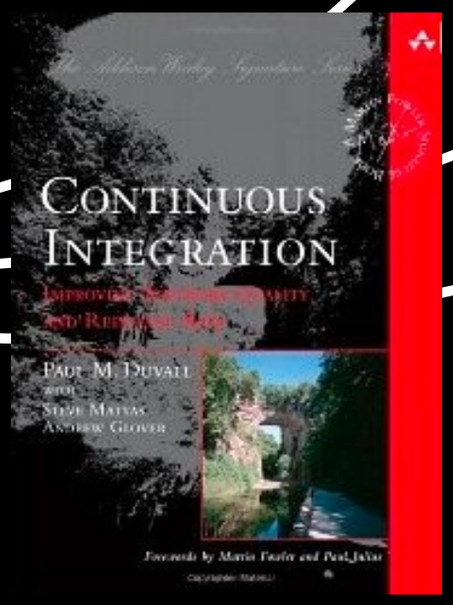
開発者

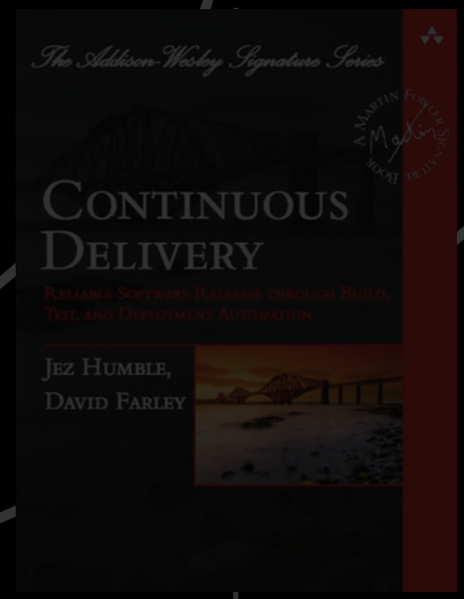
運用

QA

PL 他

ユーザー







Photograph by Melissa Brandts, Your Shot

© COPYRIGHT NATIONAL GEOGRAPHIC SOCIETY. ALL RIGHTS RESERVED.

単体テストの実行
受入テストの実行
プロビジョニング
デプロイメント
非機能テストの実行


build
pipeline

テスト以外
こそ機械に
やらせよう

```
<IfDefine DEFAULT_VHOST>
# see bug #178966 why this is in here

# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 8080

# Use name-based virtual hosting.
NameVirtualHost *:8080

# When virtual hosts are enabled, the main host defined in the default
# httpd.conf configuration will go away. We redefine it here so that it is
# still available.
#
# If you disable this vhost by removing -D DEFAULT_VHOST from
# /etc/conf.d/apache2, the first defined virtual host elsewhere will be
# the default.
<VirtualHost *:8080>
        ServerName www.hsbt.org
        Include /etc/apache2/vhosts.d/default_vhost.include

        <IfModule mpm_peruser_module>
                ServerEnvironment apache apache
        </IfModule>
</VirtualHost>
</IfDefine>
```

設定ファイルの誤り

Chef

Reality-Based automation for the Cloud.

Chef is an open-source systems integration framework built specifically for automating the cloud. No matter how complex the realities of your business, Chef makes it easy to deploy servers and scale applications throughout your entire infrastructure. Because it combines the fundamental elements of configuration management and service oriented architectures with the full power of Ruby, Chef makes it easy to create an elegant, fully automated infrastructure.

Rule The Cloud with Chef



Watch and Learn How

How Chef Works

You write "recipes" that describe how you want a part of your server (such as Apache, MySQL, or Hadoop) to be configured. These recipes describe a series of resources that should be in a particular state - p kages that should be stalled, services ho un f t should be written. hef k ac esource is properly gu , a gi a s e, flexible, easily-repeatable mechanism for making sure your

Why Chef Works

Most infrastructure solutions force you to adapt to their "one-size-fits-all" approaches, even though every infrastructure challenge is different. That's crazy-talk. And it's precisely the reason why Chef is built directly on top of Ruby, a dynamic, open-source programming language, integrated with well-known cloud platforms like Rackspace and EC2, and built to seamlessly integrate into your existing environment. We designed Chef from the ground up

Presenting the Priva
Chef 60-Day Free Tri
Fully automated infrastruc
Fully supported.

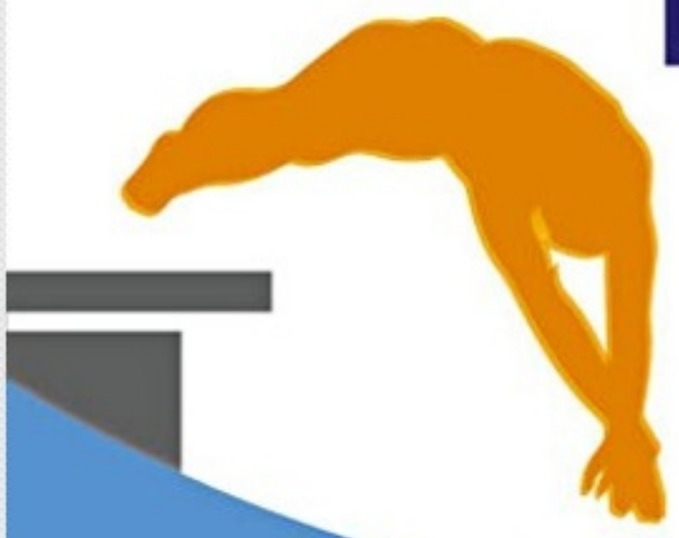
Chef

OPSC

The power and speed of Opscode. The freedom and flexibility of Chef. Completely customized for your busines behind your firewall. No mat how complex the realities of business, Private Chef make easy to deploy servers and applications throughout you entire infrastructure.

Learn more
Start your free trial

chef



Dive into automation and bring home the GOLD

Download Puppet Enterprise

Learn more »

Puppet Enterprise Webinar

Watch our weekly live demo with Q&A.



Learning Puppet VM

Learn Puppet in a safe, virtual environment



Register for PuppetConf

Hear these great speakers and more!

Our customers manage millions of nodes with Puppet

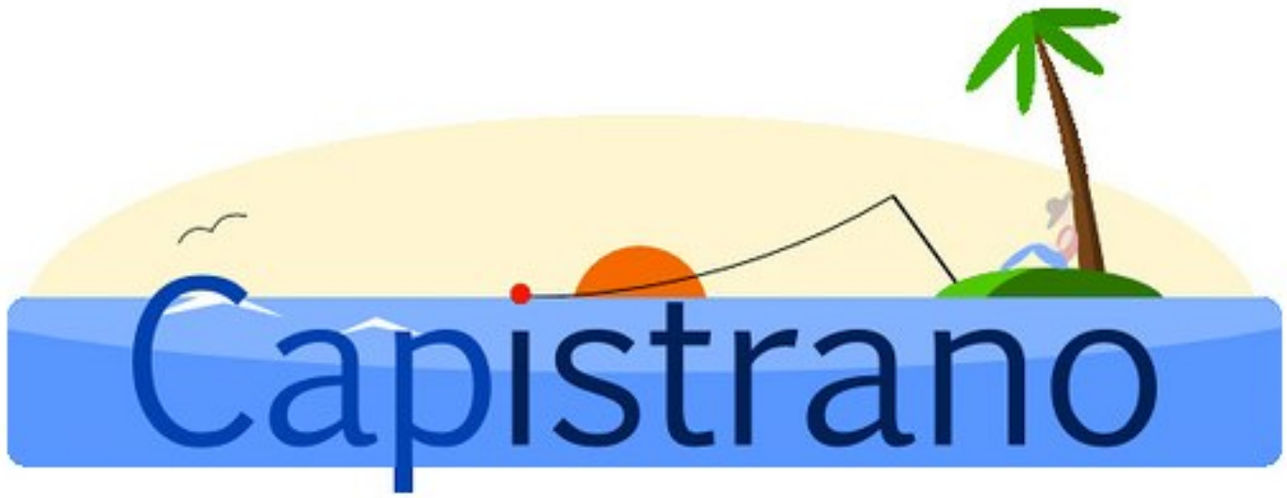
puppet

Constant Contact

"Puppet is fantastic at configuration management and everyone is really excited about it. We selected Puppet for

Capistrano.rb

Source Code Issues Wiki



General Resources & Beginner Guides

If you are new to Capistrano 2.x please read Getting Started, once you are familiar with the basics you may want to read From the beginning which is a much more detailed tutorial.

You may also want to read, and or contribute to the Capistrano handbook - an immature document that holds some really useful tips, but needs a lot more work!

- Github Guide to Deploying with Capistrano

Frameworks & Tools

- Using Capistrano 2.x with Rails 2.x
- Using Capistrano 2.x with Bundler
- Using Capistrano 2.x with RVM

Intermediate Resources

- Default Deployment Behaviour
- Managing `database.yml` Securely
- Packaging Recipes

Reference Resources

- API Documentation
- DSL Documentation
  - Action Module (`get()` , `run()` , `stream()` , `capture()` , etc)
  - Configuration Module (`set()` , `fetch()` , etc)
- Configuration Variables
- Capistrano

capistrano

30 days Album™
We deploy our changes

Capistrano deployment the easy way

Projects

- 30days-front
- 30days-puppet
- 30days-yum

Hosts

app001.30d.jp
app001.30d.local
app002.30d.jp
app002.30d.local
job001.30d.local
job002.30d.local
manage001.30d.jp
manage001.30d.local
manage201.30d.jp
mysql2.30d.jp
storage001.30d.local
storage002.30d.local
storage003.30d.local

Recipes

9 recipes

12 users

Welcome to Webistrano

Introduction and help

Welcome to Webistrano. Your first steps should be to create a project with a stage. This stage can then be deployed once you made at least one host known to Webistrano.



For help and more documentation, see

- Wiki
- Configuration Parameter
- FAQ

Webistrano is Open Source. Paid support is available by Peritor Consulting.

Recent deployments

Project: 30days-front User: mayoto Started: 2012-09-10 18:07
Stage: development Rev: 1f2bf6a... Duration: half a minute
Comment: issue #206 の時刻制限 Task: deploy
Show

Project: 30days-front User: mayoto Started: 2012-09-10 15:06
Stage: development Rev: 3da0773... Duration: half a minute
Comment: issue #206 の指定時刻にする機能 Task: deploy
Show

User: admin Started: 2012-09-10 11:18
Rev: 9432b00... Duration: less than a minute
Jenkins Task: deploy
Show

webistrano

Migrations

Migrations are a convenient way for you to alter your database in a structured and organized manner. You could edit fragments of SQL by hand but you would then be responsible for telling other developers that they need to go and run them. You'd also have to keep track of which changes need to be run against the production machines next time you deploy.

Active Record tracks which migrations have already been run so all you have to do is update your source and run `rake db:migrate`. Active Record will work out which migrations should be run. It will also update your `db/schema.rb` file to match the structure of your database.

Migrations also allow you to describe these transformations using Ruby. The great thing about this is that (like most of Active Record's functionality) it is database independent: you don't need to worry about the precise syntax of `CREATE TABLE` any more than you worry about variations on `SELECT *` (you can drop down to raw SQL for database specific features). For example you could use SQLite3 in development, but MySQL in production.

In this guide, you'll learn all about migrations including:

- **The generators you can use to create them**
- **The methods Active Record provides to manipulate your database**
- **The Rake tasks that manipulate them**
- **How they relate to `schema.rb`**

Chapters

1. Anatomy of a Migration
   - Migrations are Classes
   - What's in a Name
   - Changing Migrations
   - Supported Types
2. Creating a Migration
   - Creating a Model
   - Creating a Standalone Migration
3. Writing a Migration
   - Creating a Table
   - Changing Tables
   - Special Helpers
   - Using the change Method
   - Using the up/down Methods
4. Running Migrations
   - Rolling Back
   - Resetting the database
   - Running specific migrations
   - Changing the output of running migrations
5. Using Models in Your Migrations
6. Schema Dumping and You
   - What are Schema Files for?

Rails Migrations

Deploying Rails

Automate, Deploy,
Scale, Maintain,
and Sleep at Night



Build Pipeline View: user-app

Configure View

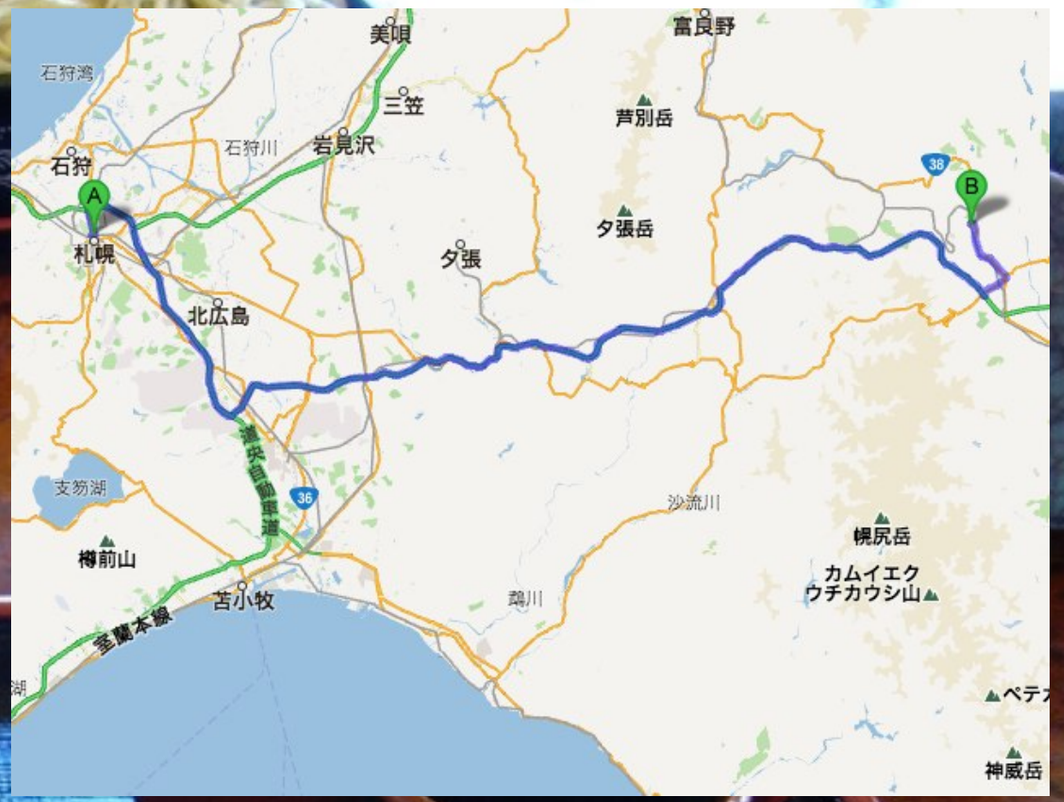
Start Build of Pipeline for petit-app-setup

View/Hide Build Pipeline Icon Legend

	petit-app-setup	petit-app-1.9.3-p194	petit-app-deploy
Git: c905d66ce57823d7b...	petit-app-setup #144 Sep 12, 2012 9:01:10 AM JST Duration: 46 秒	petit-app-1.9.3-p194 #262 Sep 12, 2012 9:02:02 AM JST Duration: 3 分 50 秒	petit-app-deploy #172 Sep 12, 2012 9:05:58 AM JST Duration: 4 分 0 秒
Git: c905d66ce57823d7b...	petit-app-setup #143 Sep 11, 2012 6:06:20 PM JST Duration: 33 秒	petit-app-1.9.3-p194 #261 Sep 11, 2012 6:06:58 PM JST Duration: 3 分 57 秒	petit-app-deploy #171 Sep 11, 2012 6:11:00 PM JST Duration: 3 分 59 秒
Git: 81af18ff189900b08...	petit-app-setup #142 Sep 11, 2012 9:01:11 AM JST Duration: 52 秒	petit-app-1.9.3-p194 #260 Sep 11, 2012 9:02:08 AM JST Duration: 3 分 58 秒	petit-app-deploy #170 Sep 11, 2012 9:06:12 AM JST Duration: 3 分 6 秒
Git: 81af18ff189900b08...	petit-app-setup #141 Sep 10, 2012 8:26:06 PM JST Duration: 31 秒	petit-app-1.9.3-p194 #259 Sep 10, 2012 8:26:43 PM JST Duration: 4 分 22 秒	petit-app-deploy #169 Sep 10, 2012 8:31:11 PM JST Duration: 2 分 45 秒
Git: 2fcd24b...23c5...559...	petit-app-setup #140 Sep 10, 2012 7:01:07 PM JST ...26 秒	petit-app-1.9.3-p194 #258 Sep 10, 2012 7:01:39 PM JST Duration: 3 分 43 秒	petit-app-deploy #168 Sep 10, 2012 7:05:29 PM JST Duration: 3 分 12 秒

Build Pipeline Plugin

確実なフィードバックと改善を行える仕組み



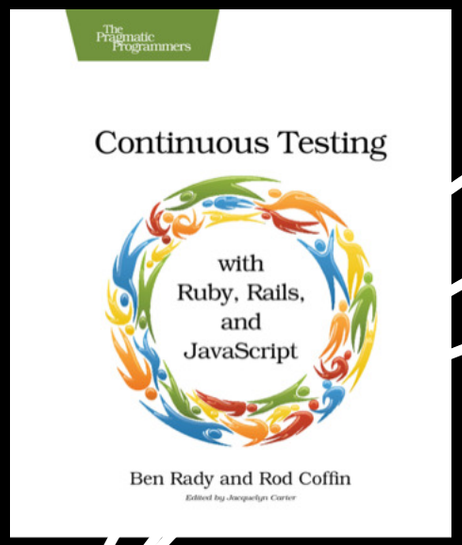
そばレストラン

玄穣

石臼挽き手打ちそば

3 2 E

サイクルタイムの短縮と継続した価値の提供



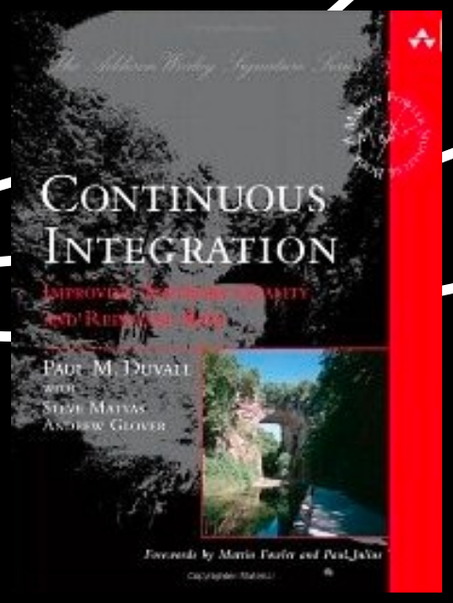
ユーザー

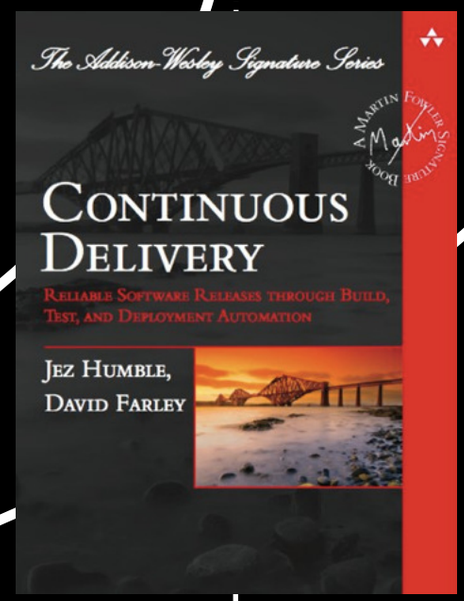
PL 他

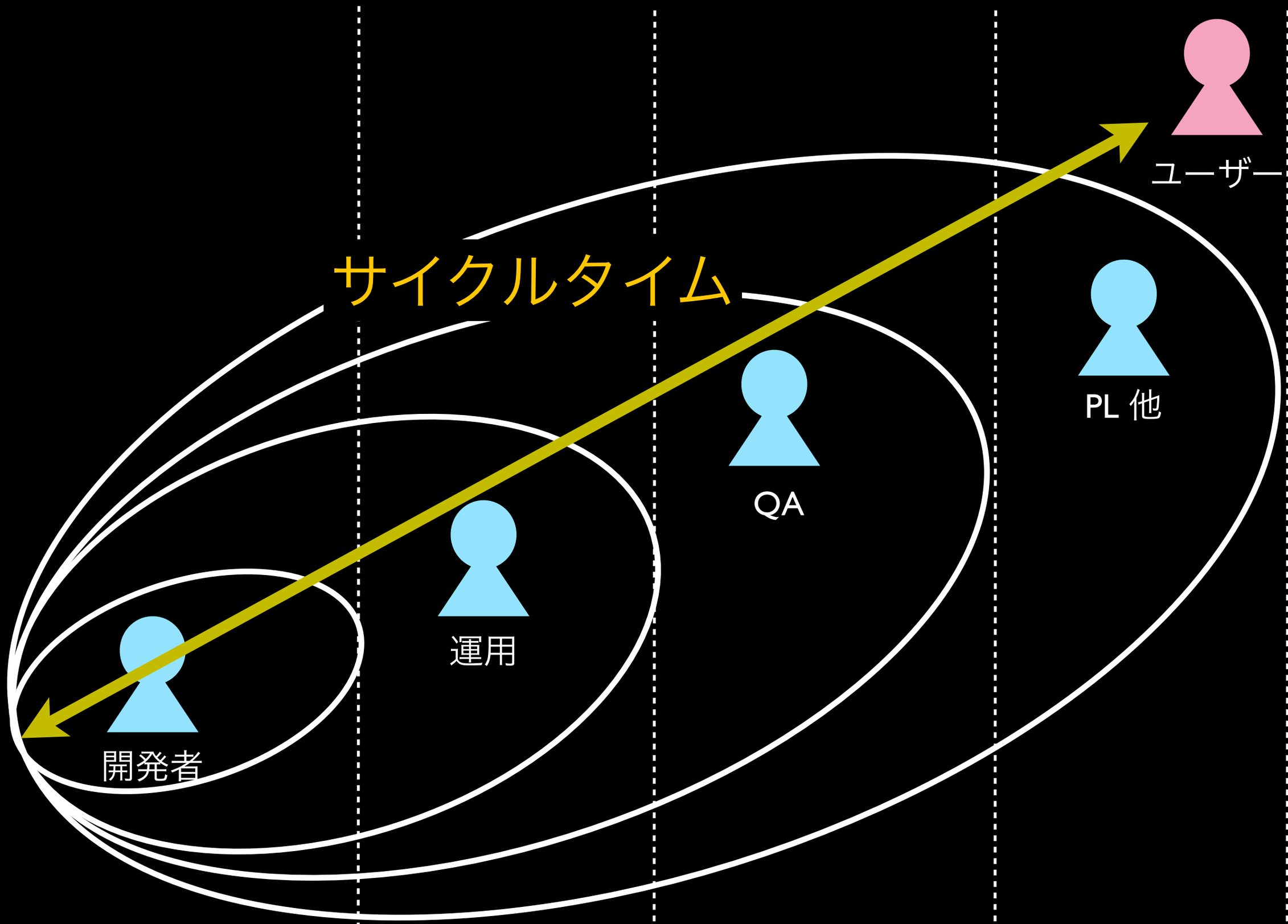
QA

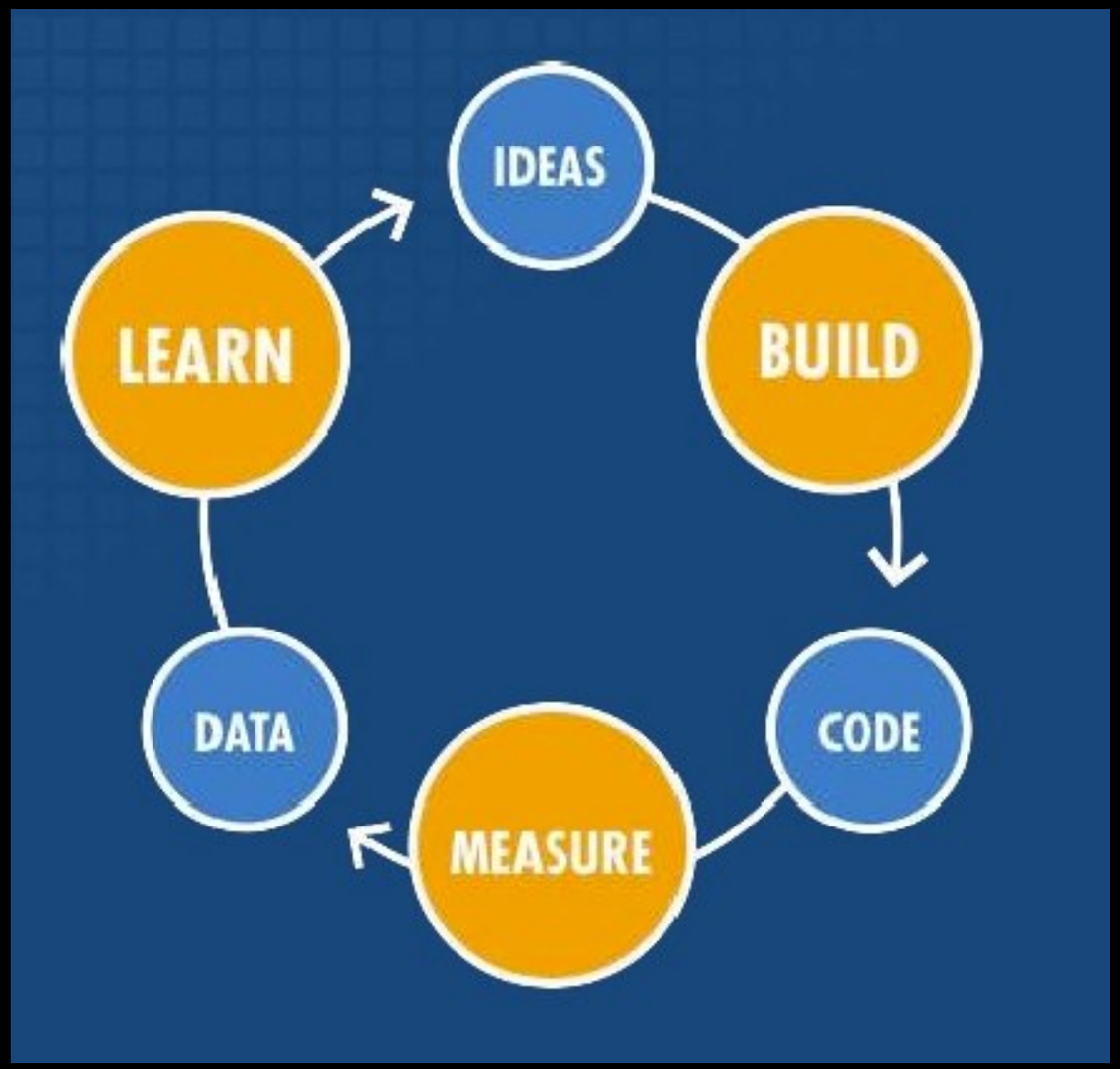
運用

開発者






サイクルタイム
ユーザー
PL 他
QA
運用
開発者



変更すると決めてからユーザーが使えるようになるまでの時間を短くする

tdiary-core / Commit History

Keyboard shortcuts available

Sep 08, 2012

update dependency
hsbt authored 6 hours ago
9408355ff9
Browse code

Aug 31, 2012

Merge pull request #199 from tdiary/js-testing ···
ago
843937291d
ode

add javascri ec
machu author ys ago
8 7406
wse code

setup jasmi
ys ago
Brow

add jasmine for testing javascript
machu authored 9 days ago
c6f7d067d9
Browse code

Aug 25, 2012

update dependency
hsbt authored 15 days ago
d346d5a156
Browse code

Aug 24, 2012

bump vertion t 3.1.4.20120824
machu authored 16 days ago
18c851d5f8
Browse code

Merge pull request #198 from machu/default_data_path ···
machu authored 16 days ago
1f82e64576
Browse code

not ignore .htaccess in @data_path
machu authored 16 days ago
c2656a0ddb
Browse code

コミットスナージ



ユーザーストーリー

INVEST

independent

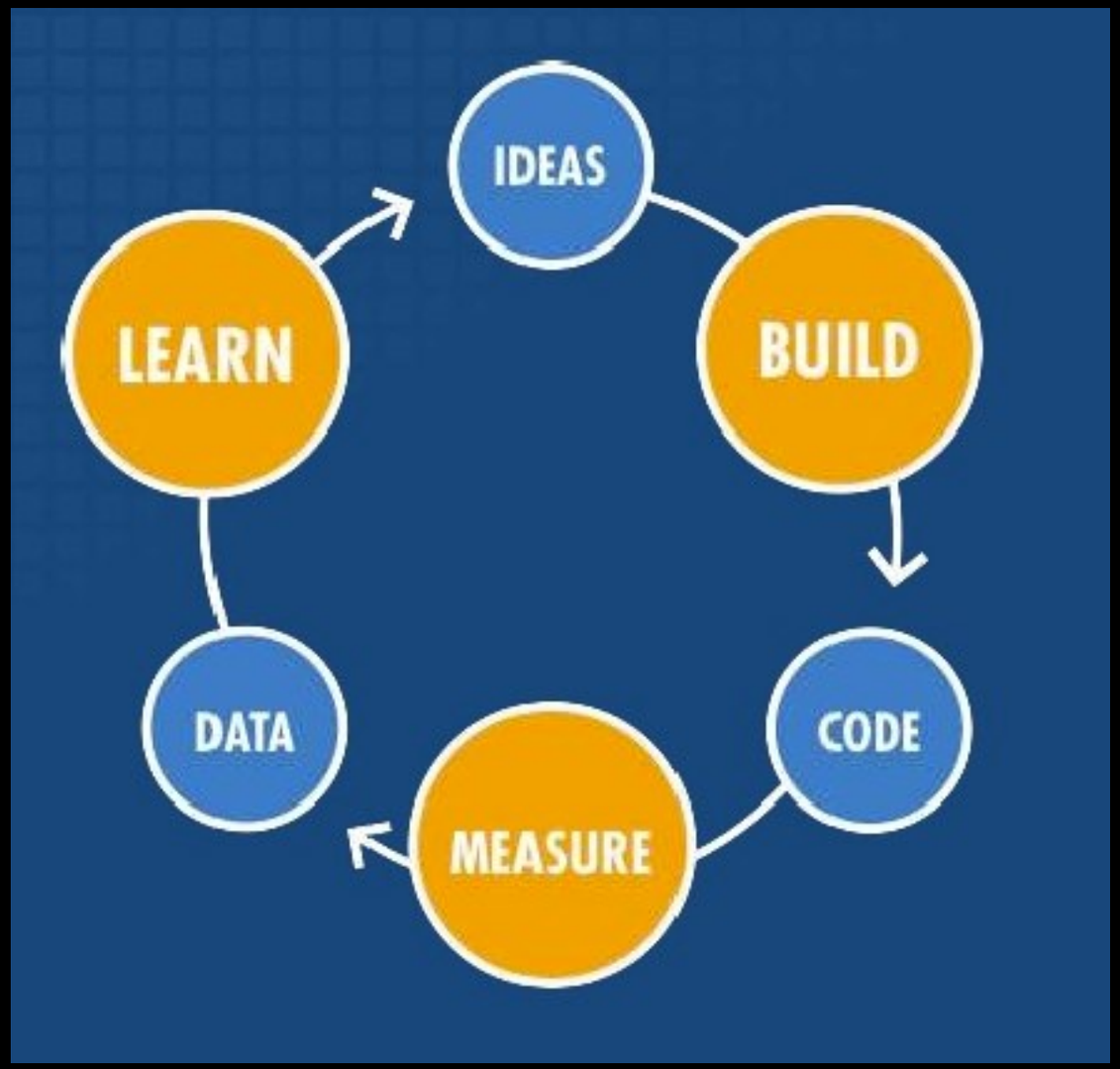
negotiable

valuable

estimatable

sized appropriately

testable

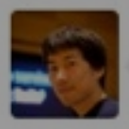


変更すると決めてからユーザーが使えるようになるまでの時間を短くする



kyanny added some commits — 10 days ago

kyanny	Remove past contest files	b30bc92
kyanny	Remove past contest routing	a016a26



hsbt referenced this pull request — 10 days ago

Pull Request #187: **Fix 187 attributes** — Closed



hsbt commented — 10 days ago

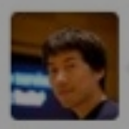
album_expo と third_anniversary, voice を grep してみると他にも結構ありますね。ぽちぽち消していきます。



kyanny commented — 10 days ago

@hsb
の手 remo st-c st-fi EAD) では a _expo ird_a rs 一個 ット いの が、
残っ るか てく い。
と a iews/ 以 ある イルはコンテ のアー ブページ用で も残 く ページ で れ
にマ は無 よい。

- http://30d.jp/contest/albumexpo
- http://30d.jp/contest/3rd_anniversary
- http://30d.jp/tour/voice



hsbt commented — 10 days ago

https://github.com/paperboy-30days/30days-front/blob/remove-past-contest-files/app/controllers/albums_controller.rb#L72
この辺は必要ですか?



kyanny commented — 10 days ago

> https://github.com/paperboy-30days/30days-front/blob/remove-past-contest-files/app/controllers/albums_controller.rb#L72
> この辺は必要ですか?

pull request

探索テスト

パフォーマンステスト

Jenkins Promoted builds

Promoted Builds Plugin

5 Added by Kohsuke Kawaguchi, last edited by ohtake_tomohiro - on Aug 06, 2012 (view change)

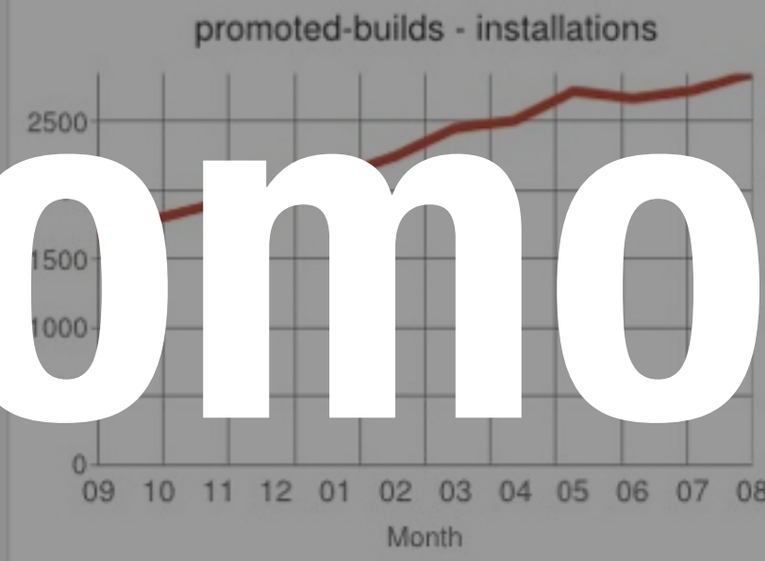
Jenkins

- Home
- Mailing lists
- Source code
- Bugtracker
- Security Advisories
- Donation
- Commercial Support
- Wiki Site Map

Documents

- Meet Jenkins
- Use Jenkins
- Extend Jenkins
- Plugins
- Servlet Container Notes

Plugin In[illegible]mat[illegible]

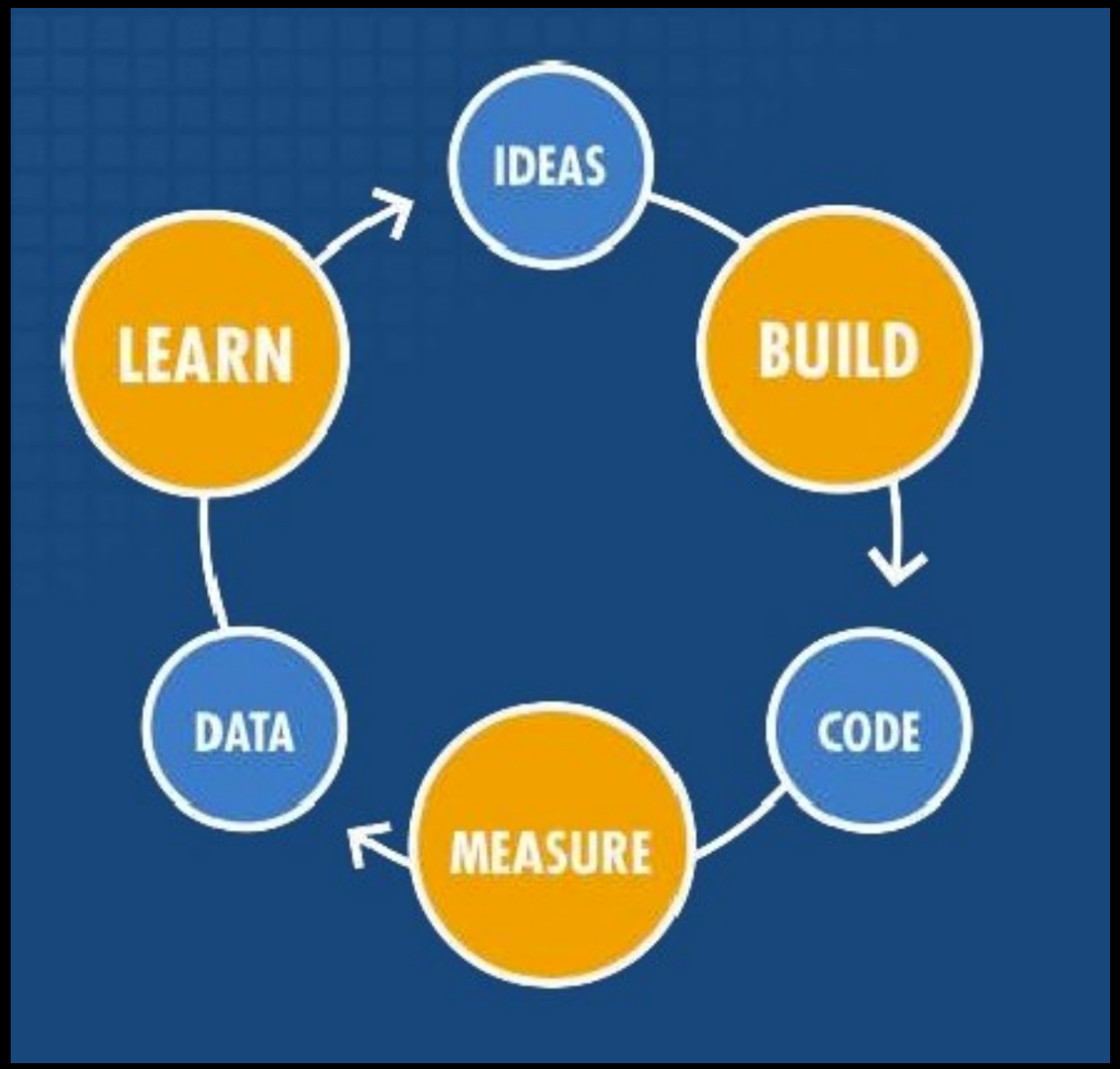
Plugin ID	[illegible]	**[illegible]ange[illegible]**	[illegible]atest Release [illegible]test Release
Lat[illegible]	[illegible]	**[illegible]urce[illegible]de**	[illegible]
Latest [illegible]ase Date	[illegible] 06, 20[illegible]	**[illegible]ssue Tracking**	[illegible] Issues
Required Core	1.448	**Maintainer(s)**	Kohsuke Kawaguchi (id: kohsuke)
Dependencies	maven-plugin (version:1.448) javadoc (version:1.0)		Peter Hayes (id: petehayes)
Usage		**Installations**	2011-Sep 1660 2011-Oct 1795 2011-Nov 1904 2011-Dec 1976 [illegible]12-Ja[illegible] [illegible]2-F[illegible]247 [illegible]2-M[illegible]2450 2012-A[illegible]505 [illegible]2-M[illegible]71[illegible] [illegible]12-Ju[illegible] 2012-Jul 2723 2012-Aug 2851

This plugin allows you to distinguish good builds from bad builds by introducing the notion of 'promotion'.Put simply, a promoted build is a succe[illegible] as more compre[illegible]ive tests that are set up as [illegible]stre[illegible]jobs.) The ty[illegible] situation in which you use promotion is where you have multiple 'build' job. You'll [illegible] configure the build job so tha[illegible] bu[illegible]ets promote[illegible]en all the test jobs passed successfully. This allows you to keep faster feedback [illegible]ld fail[illegible]nd y[illegible]an s[illegible]isti[illegible]h bui[illegible] good [illegible]ilds that compiled but had runtime problems.

Another variatio[illegible]his u[illegible]e is [illegible]manu[illegible] pro[illegible] bu[illegible] (bas[illegible]on in[illegible]t o[illegible]mething else that runs outside Jenkins.) Promoted builds w[illegible] be then picked u[illegible] othe[illegible]ams [illegible]ploy[illegible] th[illegible]gin[illegible]a, [illegible] as t[illegible] bu[illegible] passed additional quality criteria. In more complicate[illegible] promotions. This [illegible] nice [illegible] an[illegible]ironn[illegible] wh[illegible]here [illegible] mu[illegible]e sta[illegible] of testings [illegible] example, QA testing, acceptance testing, staging,

Promotion Action

When a build is promoted, you can have Jenkins perform some actions (such as running a shell script, triggering other jobs, etc. — or in Jenkins for example to copy the promoted build to somewhere else, deploy it to your QA server. You can also define it as a separate job and then have



変更すると決めてからユーザーが使えるようになるまでの時間を短くする

19:05:11 <hiroya> デプロイすると /home/sqale/current/log/production.log が吹っ飛んじゃうな

19:06:23 <hiroya> さてさて...

19:10:26 (ikachan) staging先輩、お疲れ様でした！！

19:11:05 <mizzy> あー

19:11:17 <mizzy> どうすんのがいいんだろう

19:11:45 <ikachan> [nagios_alert] PROBLEM Host: staging-queue001.sqale.lan Address: staging-queue001.sqale.lan State: DOWN Date/Time: 09-11-2012 19:01:56 - Info: (Host Check Timed Out)

19:12:04 <ikachan> [nagios_alert] PROBLEM Host: staging-proxydb001.sqale.jp Address: staging-proxydb001.sqale.jp State: DOWN Date/Time: 09-11-2012 19:02:16 - Info: (Host Check Timed Out)

19:12:24 <hiroya> これは悩みますね

19:12:24 <ikachan> [nagios_alert] PROBLEM Host: staging-app001.sqale.jp Address: staging-app001.sqale.jp State: DOWN Date/Time: 09-11-2012 19:02:36 - Info: (Host Check Timed Out)

19:12:26 <ikachan> [nagios_alert] PROBLEM Host: staging-app001.sqale.lan Address: staging-app001.sqale.lan State: DOWN Date/Time: 09-11-2012 19:02:36 - Info: (Host Check Timed Out)

19:12:34 <ikachan> [nagios_alert] PROBLEM Host: staging-proxydb001.sqale.lan Address: staging-proxydb001.sqale.lan State: DOWN Date/Time: 09-11-2012 19:02:46 - Info: CRITICAL - Host Unreachable (staging-proxydb001.sqale.lan)

19:12:54 <ikachan> [nagios_alert] PROBLEM Host: staging-queue001.sqale.jp Address: staging-queue001.sqale.jp State: DOWN Date/Time: 09-11-2012 19:03:06 - Info: (Host Check Timed Out)

19:14:44 <ikachan> [nagios_alert] PROBLEM Host: staging-queue001.sqale.lan Address: staging-queue001.sqale.lan State: DOWN Date/Time: 09-11-2012 19:04:56 - Info: CRITICAL - Host Unreachable (staging-queue001.sqale.lan)

19:41:35 (ikachan) Jenkins (sqale-chef-create_container-daily): SUCCESS - http://ci.sqale.pb/job/sqale-chef-create_container-daily/29/ - rcov: http://bit.ly/yblKPE - http://app001.sqale.pb/

19:46:57 (ikachan) hiboma pushed to develop at paperboy-sqale/sqale-chef September 11, 2012 * hiboma committed 5c1e3d5 add test: sample_app BBSを追加。

19:46:58 (ikachan) https://github.com/paperboy-sqale/sqale-chef/compare/91c0950baf...5c1e3d5abb

19:48:21 (ikachan) Deployment of [chef / dev:develop] has started by jenkins ...

19:48:23 (ikachan) Comment: ------- Auto deployment by jenkins ci.sqale.pb -------

19:49:49 (ikachan) Deployment of [chef / dev:develop] has finished by jenkins (90.37 sec) - http://bit.ly/JWJfKo - http://sqale.jp/

19:51:04 (ikachan) Chef status: applied role[rootfs_ruby] users001.sqale.pb [development] rev: 5c1e3d5abb226add695c815f94302dd2b7ab06d9

19:51:36 (ikachan) Chef status: applied role[rootfs_ruby] users900.sqale.pb [development] rev: 5c1e3d5abb226add695c815f94302dd2b7ab06d9

19:51:57 (ikachan) pplog commented on issue paperboy-sqale/sqale-app#933 明日、お願いします！ping @demi168

19:51:58 (ikachan) https://github.com/paperboy-sqale/sqale-app/issues/933#issuecomment-8455577

19:53:57 (ikachan) pplog opened issue 949 on paperboy-sqale/sqale-app September 11, 2012 トップページからデプロイマニュアルへ誘導

19:53:58 (ikachan) https://github.com/paperboy-sqale/sqale-app/issues/949

19:56:41 (ikachan) Jenkins (sqale-chef): SUCCESS - http://ci.sqale.pb/job/sqale-chef/399/ - rcov: http://bit.ly/yblKPE - http://app001.sqale.pb/

20:10:58 (ikachan) pplog commented on issue paperboy-sqale/sqale-app#945 http://192.168.25.37/gyazo/data/d8fc60e547c82dffb4cdc9ffef33fc6c.png ボタンを置くとしたらアプリ毎の詳細ページでしょうか、タグのやつ @[illegible]いした

[illegible]) https://[illegible]hub.com/p[illegible]/sqale-app/iss[illegible]945#issuec[illegible]nt-8455975

2[illegible]han) pplo[illegible]ened issue 9[illegible]rboy-sqale/sq[illegible] Septen[illegible]2012 クーポンコード短くしたい

[illegible]) https://g[illegible].com/pap[illegible]/sqale-[illegible]/issues/950

全てをフィードバック

継続的な改善

継続的デリバリー最後の砦

Migrations

Migrations are a convenient way for you to alter your database in a structured and organized manner. You could edit fragments of SQL by hand but you would then be responsible for telling other developers that they need to go and run them. You'd also have to keep track of which changes need to be run against the production machines next time you deploy.

Active Record tracks which migrations have already been run so all you have to do is update your source and run `rake db:migrate`. Active Record will work out which migrations should be run. It will also update your `db/schema.rb` file to match the structure of your database.

Migrations also allow you to describe these transformations using Ruby. The great thing about this is that (like most of Active Record's functionality) it is database independent: you don't need to worry about the precise syntax of `CREATE TABLE` any more than you worry about variations on `SELECT *` (you can drop down to raw SQL for database specific features). For example you could use SQLite3 in development, but MySQL in production.

In this guide, you'll learn all about migrations including:

- **The generators you can use to create them**
- **The methods Active Record provides to manipulate your database**
- **The Rake tasks that manipulate them**
- **How they relate to `schema.rb`**

Chapters

1. Anatomy of a Migration
   - Migrations are Classes
   - What's in a Name
   - Changing Migrations
   - Supported Types
2. Creating a Migration
   - Creating a Model
   - Creating a Standalone Migration
3. Writing a Migration
   - Creating a Table
   - Changing Tables
   - Special Helpers
   - Using the change Method
   - Using the up/down Methods
4. Running Migrations
   - Rolling Back
   - Resetting the database
   - Running specific migrations
   - Changing the output of running migrations
5. Using Models in Your Migrations
6. Schema Dumping and You
   - What are Schema Files for?

Rails Migrations

サイクルタイムの短縮と継続した価値の提供



小樽 なると

Wrapping Up

意識が低くてもOKな仕組み

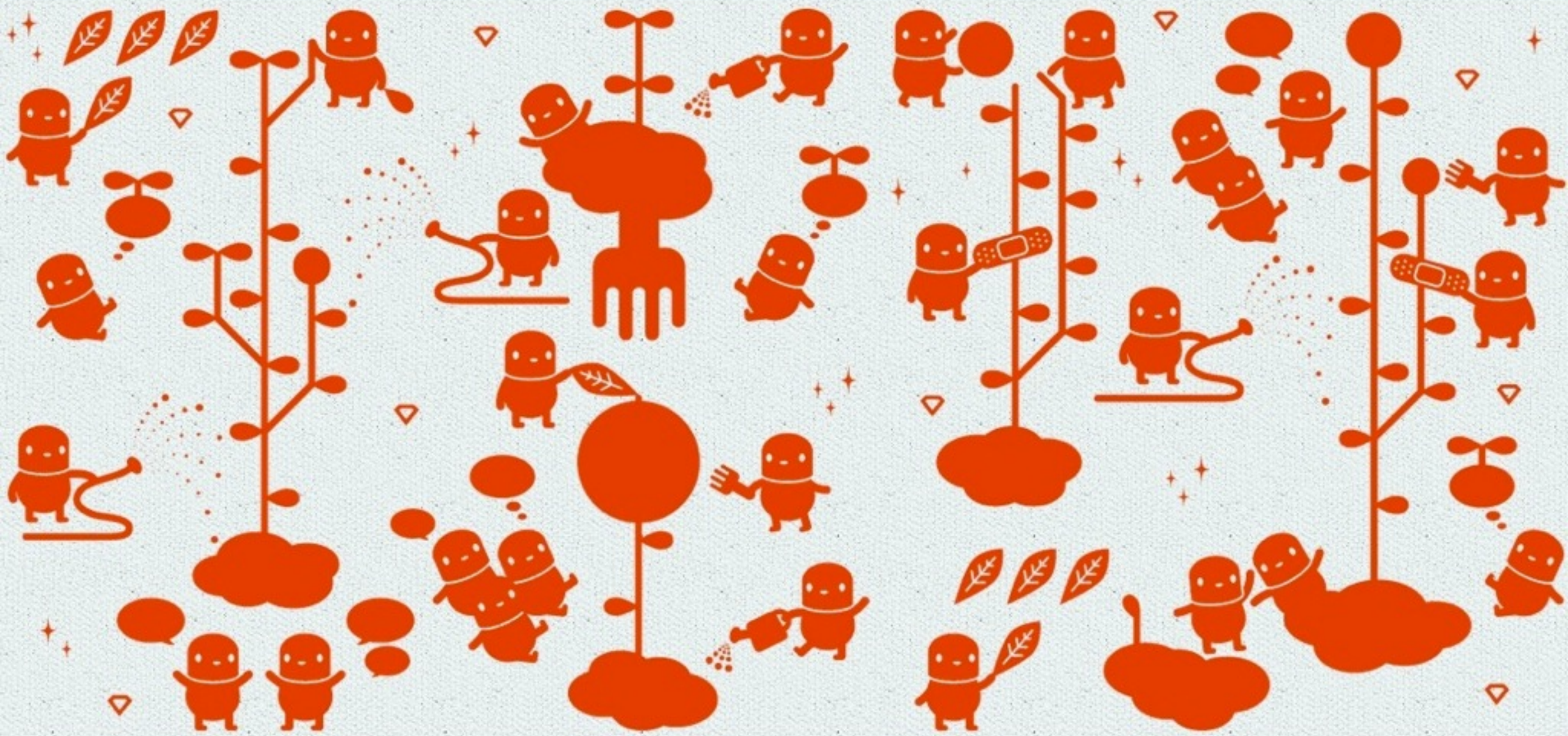
確実なフィードバックと改善を行える仕組み

サイクルタイムの短縮と継続した価値の提供

Sapporo RubyKaigi 2012

We Code.

September 14-16th, 2012
at Sapporo Business Innovation Center



Keynote:

- Yukihiro Matsumoto
- Aaron Patterson

Show Skitch

日本語 | English

rence Registration

*3日間通し券

REGISTER