

Rubyによるアジヤイルとデブオプスを
実現するチームづくり
—みんなのウェディングの経験から

みんなのウェディング

- ・ 結婚式場紹介口コミサービス
- ・ 月間ユニークブラウザ数 約400万
- ・ 結婚式場選びで52.4%の人が参考に
(ブライダル産業新聞調べ)



株式会社みんなのウェディング

所在地 東京都中央区築地1-13-1

銀座松竹スクエア9階

資本金 1,436百万円 (2015年6月末現在)

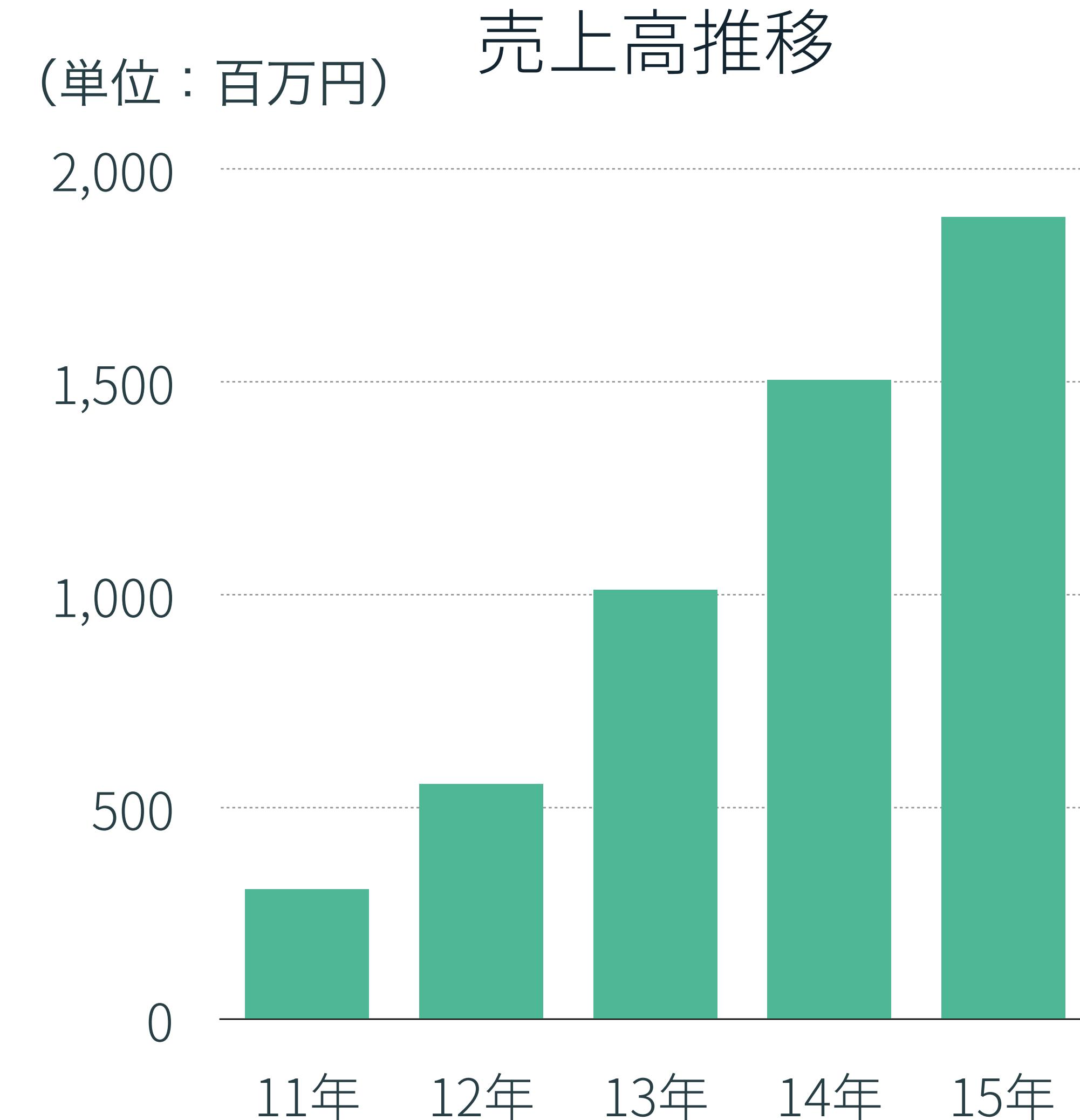
沿革

2008年2月 DeNAの新規事業として開始

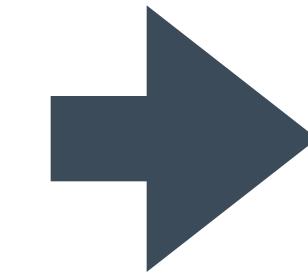
2010年10月 会社設立

2014年3月 東京証券取引所マザーズ上場

2015年4月 クックパッドと資本業務提携



「クックパッド」から「みんなのウェディング」へ



cookpad



みんなの
ウェディング

PerlはRubyのお兄さん(or お姉さん)
——まつもとゆきひろ



MobaSiFとは？

- DeNAによって開発された「ガラケー」向けフレームワーク
- FastCGIで動作し、Apache HTTP Serverを前提としている
- オープンソース版は、2008年5月に最後のリリース

```
sub pageDiaryView {
    my $func = shift;
    my $rhData = Common::cloneHash($_::F, '^[a-z]');
    my $dbh = DA::getHandle($_::DB_DIARY_R);
    my $sth = $dbh->prepare(<<'SQL');
    select
        diary_id, user_id, post_date, subject, content,
        comment_num, access_num, last_user
    from diary_data where diary_id=?SQL
    $sth->execute($rhData->{d});
    if (!$sth->rows) {
        MException::throw({ CHG_FUNC => '.404' });
    }
    Common::mergeHash($rhData, $sth->fetchrow_hashref());
    my $html = HTMLTemplate::insert("diary/dia_view", $rhData);
    Response::output(\$html);
}
```

```
def show
  @diary = Diary.find(prams[:id])
end
```

みんなのウェーディング開発体制（当時）

- エンジニア … 15名
- 定義された開発プロセスやプロジェクト管理手法がない
 - エンジニアはビジネス側からの要求で「アサイン」される
- 自動テストがない
 - 変更によってシステムが壊れても分からない

みんなのウェディング開発環境（当時）

- 共用の開発用サーバーが1台
 - ソースコードディレクトリをsambaで共有
- Subversionでバージョン管理
 - 他の開発者の変更が混ざらないように、声をかけあってファイルをコミット
 - コミットユーザーは全員同じユーザーなので、コミットメッセージに名前を含める

みんなのウェディングインフラ環境（当時）

- ・仮想専用サーバー
 - ・エンジニアの手作業による設定管理
 - ・サーバーの費用が高く、柔軟に調達できない
 - ・ひとつのサーバーに複数の役割が同居
 - ・単一障害点（SPOF）が存在し、バックアップが不明確
- ・監視体制が不十分で障害を検知できない
 - ・クライアントからの連絡でシステム障害が判明

Ruby・Ruby on Railsの歴史

2003年8月	Ruby 1.8 公開
2004年7月	Ruby on Rails 0.5 公開
2005年12月	Ruby on Rails 1.0 公開
2006年6月	日本Rubyカンファレンス2006（基調講演 DHH）
2007年6月	日本Ruby会議2007（基調講演 Dave Thomas）
2007年12月	Ruby on Rails 2.0 公開
	Ruby 1.9 公開
2008年4月	Github サービス開始

- ・ ビジネス成長のための圧力が強く、技術的な投資がなおざりになっていた
- ・ 継続的な改善をしていくための開発体制がなかった

エンジニアリングの責任者として
どのような戦略をとるか？

- ・ コモディティを利用する
- ・ プロセスのアジリティを向上させる

コモディティとは

【英】 commodity

コモディティとは一般化したため差別化が困難
となつた製品やサービスのことをいゝ。

——トライベック・ブランド戦略研究所

コモディティのメリット

- 実戦で検証されている (battle-proven)
- ユーザーコミュニティがありノウハウが流通している
- 豊かなエコシステムが存在する

- ✓ Ruby on Rails
- ✓ Amazon Web Services (AWS)

プロセスのアヅリティ？

This is Agility There is no more

- A. Where do we want to be?
- B. Where are we?
- C. How do we improve our position?



- A. Where do we want to be?
- B. Where are we?
- C. How do we improve our position?



「つくるもの」だけではなく
「つくりかた」もカイゼンする

- ✓ アジャイル開発
- ✓ デブオプス

DevOps is the practice of operations and development engineers participating together in the entire service lifecycle, from design through the development process to production support.

- ・ バージョン管理
- ・ 課題管理
- ・ 構成管理
- ・ 繼続的インテグレーション
- ・ 監視
- ・ テスト自動化
- ・ デプロイ

Lowering risk of change
through tools and culture

— John Allspaw

culture

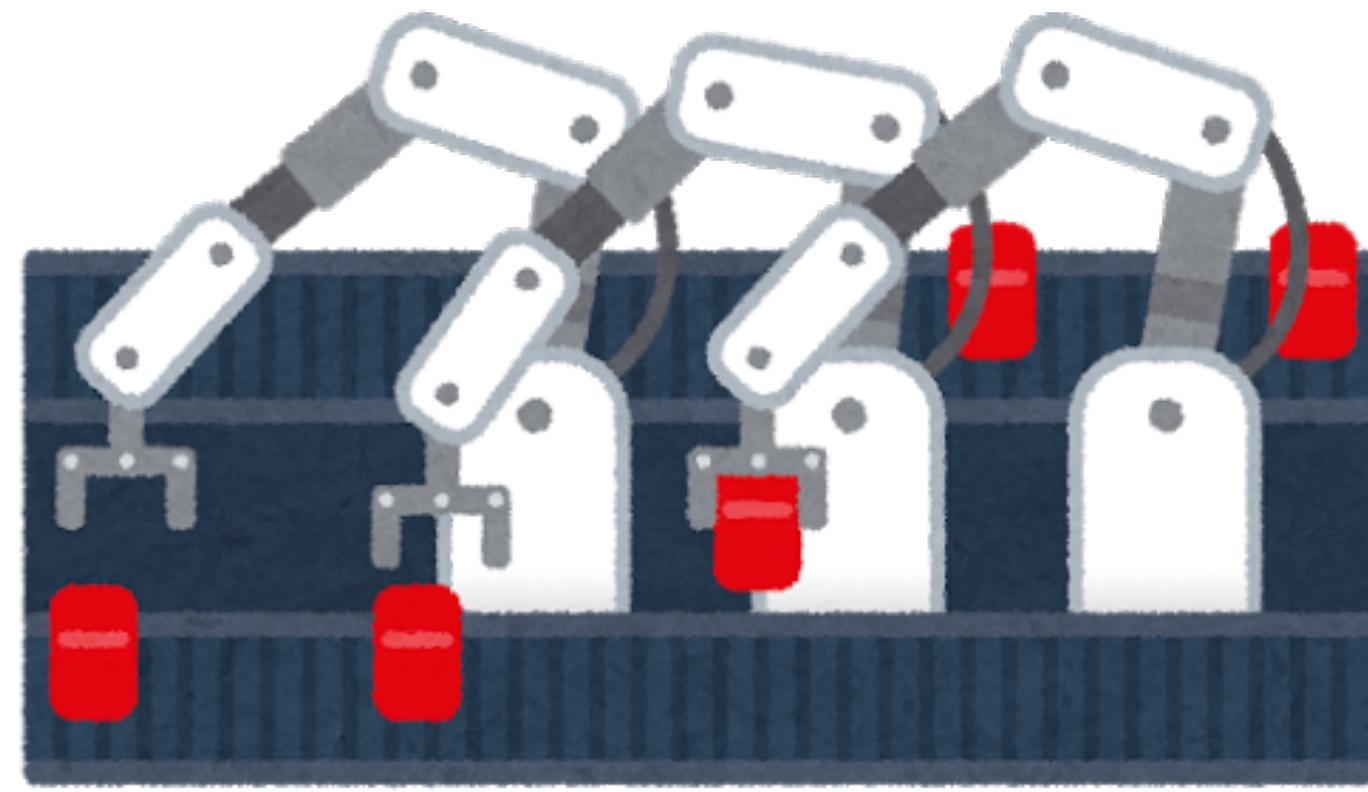
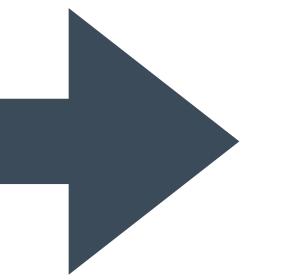
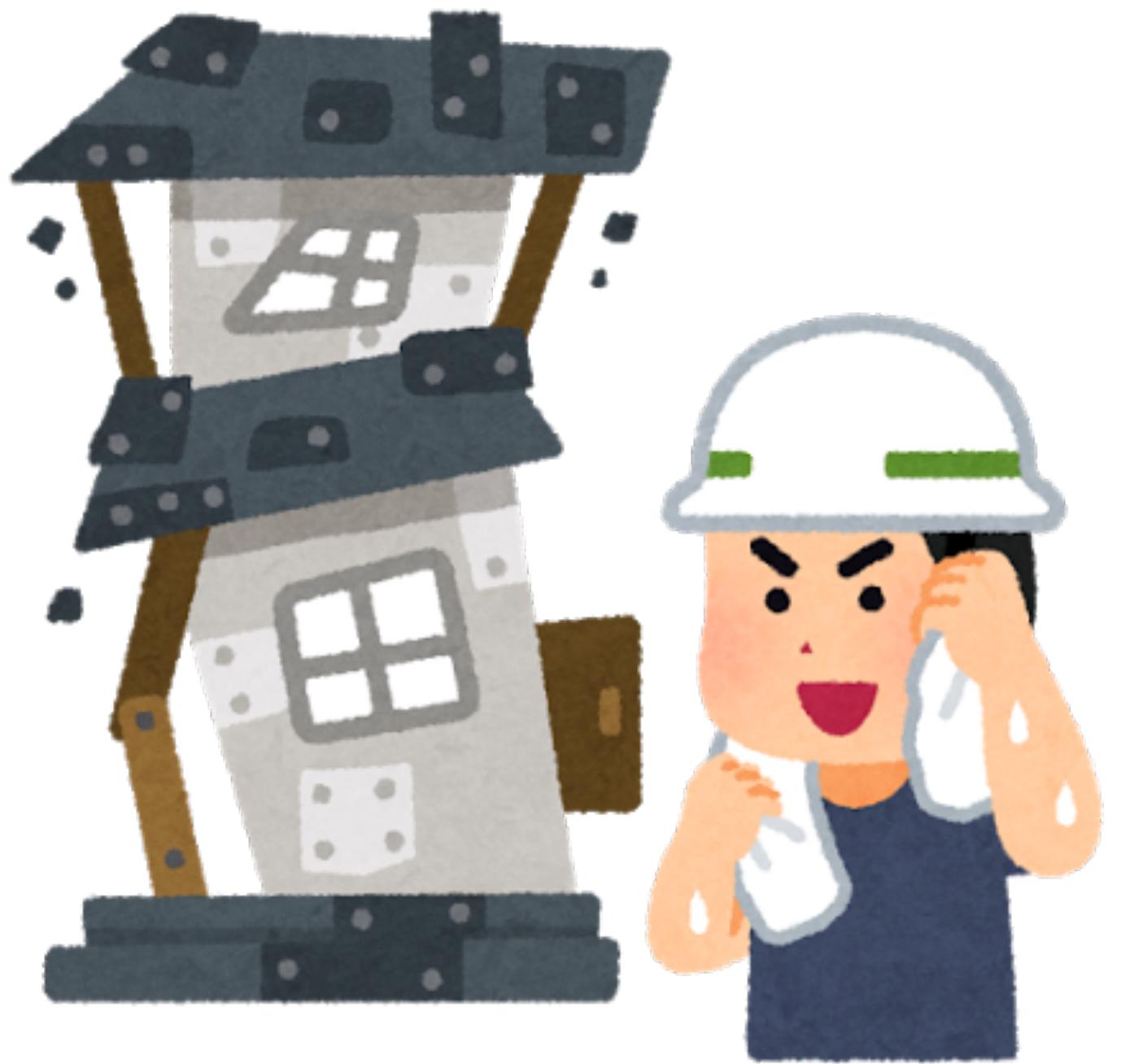


イデオロギーは、物質的なイデオロギー装置のなかに存在し、この装置は物質的な儀式によって調整される物質的な諸実践を命令し、これらの諸実践は自己の信仰に従って全く意識的に行動する主体の物質的な諸行為のなかに存在する

——ルイ・アルチュセール

エンジニアリングの責任者として
どのような戦略をとるか？

- AWS とデブオプス
- Ruby on Railsとアジャイル開発



今までのインフラ

これからのインフラ



今までのインフラ



これからのインフラ

お客様事例

ユースケース

移動

Amazon.com ギフトカードを取得する
AWS ユーザビリティスタディに参加する »

新たに AWS アカウントを作成いた
だくと、1年間の無料利用枠をご利
用いただけます。

[まずは無料で始める »](#)

AWS 導入事例：株式会社みんなのウェディング



～ AWS クラウドを活用したアプリケーション構築・運用経験が、
当社エンジニアの今後のキャリア形成にもプラスに～

バックグラウンド

株式会社みんなのウェディングは、「みんなの『大切な日』をふやす」を経営理念として掲げ、花嫁・花婿の視点に立った口コミ情報等結婚関連情報提供事業を行っています。“ユーザー第一主義”的な視点を徹底し、実際に結婚式を挙げた方の本当の声（口コミ）や実際の費用明細を掲載することで、ユーザーにとって結婚式の“本当”を伝えて納得いく結婚式に出会えることをサービスの一番の目的にしています。サービスは 2008 年より開始し、現在では月間 300 万を超えるUU数を獲得しています。

ビジネス面での課題

当社のようなインターネットサービス企業にとって、ユーザーの利便性を徹底的に追求し、ユーザーから支持されるサービスづくりをすることが非常に重要となっています。そのためには、アプリケーション開発の俊敏性を高め、素早く改善サイクルを回していく体制づくりをしなければならないという課題がありました。

サービスの根幹にあるインフラストラクチャも同様です。その時のニーズにあわせて、柔軟にサーバー資源を確保したり構成することが容易でなければなりません。にもかかわらず、AWS 導入以前に利用していた環境では料金体系や納期の関係で簡単にサーバーの追加や廃止ができなかったため、サービス改善のボトルネックになりかねないという状況でした。



アマゾン ウェブ サービスが選ばれた理由

柔軟なサーバー資源の確保、構築環境を実現するために、いくつかのクラウドサービスを検討していった結果、AWS が最も最適である

たくと、1年間の無料利用枠をご利
用いただけます。

Rubyによるインフラ管理

- ・ サーバー構成管理ツール (itamae) による設定管理
- ・ サービス構成管理ツールによるAWS設定管理
(bukelatta／eipmap／kelbim／piculet／roadworker)
- ・ データベーススキーマ管理 (ridgepole)

+ GitHubでのバージョン管理とコードレビュー

```
execute "hostname #{node[:ec2][:tags][:Name]}" do
  not_if "[ $(hostname) = \"#{node[:ec2][:tags][:Name]}\" ]"
  notifies :restart, 'service[rsyslog]'
end

file '/etc/sysconfig/network' do
  action :edit

  block do |content|
    unless content[node[:ec2][:tags][:Name]]
      content.sub!(/^HOSTNAME=.*$/,,
                  "HOSTNAME=#{node[:ec2][:tags][:Name]}")
    end
  end
end
end
```

```
$ bundle exec rake itamae:apply HOSTS=tools-001
INFO: Apply /Users/alice/infra/itamae/roles/base/default.rb,/User
itamae ssh -h 10.1.2.197 -j /var/folders/l7/16_nsg7s35b6f4_brh_pv
infra/itamae/functions.rb /Users/alice/infra/itamae/roles/base/de
infra/itamae/post-processing.rb
INFO : Starting Itamae...
INFO : Loading node data from /var/folders/l7/16_nsg7s35b6f4_brh_
/Users/alice/.rbenv/versions/2.3.1/lib/ruby/gems/2.3.0/gems/net-s
Object#timeout is deprecated, use Timeout.timeout instead.
/Users/alice/.rbenv/versions/2.3.1/lib/ruby/gems/2.3.0/gems/net-s
Object#timeout is deprecated, use Timeout.timeout instead.
INFO : Recipe: /Users/alice/infra/itamae/functions.rb
INFO : Recipe: /Users/alice/infra/itamae/roles/base/default.rb
INFO : Recipe: /Users/alice/infra/itamae/roles/base/disable-ip
INFO : Recipe: /Users/alice/infra/itamae/roles/base/nofile.rb
INFO : Recipe: /Users/alice/infra/itamae/roles/base/yum.rb
INFO : Recipe: /Users/alice/infra/itamae/roles/base/http_prox
```

あざ
つす 1



Amazon Linux AMI Security Bulletins BOT 09:18

[ALAS-2016-759 \(critical\): java-1.8.0-openjdk](#)

CVE-2016-5542, CVE-2016-5554, CVE-2016-5597, CVE-2016-5573, CVE-201

[ALAS-2016-760 \(important\): python-twisted-web](#)

CVE-2016-1000111



Itamae BOT 09:48

Itamae: naoto-takai is going to apply to tools-001



github BOT 11:25

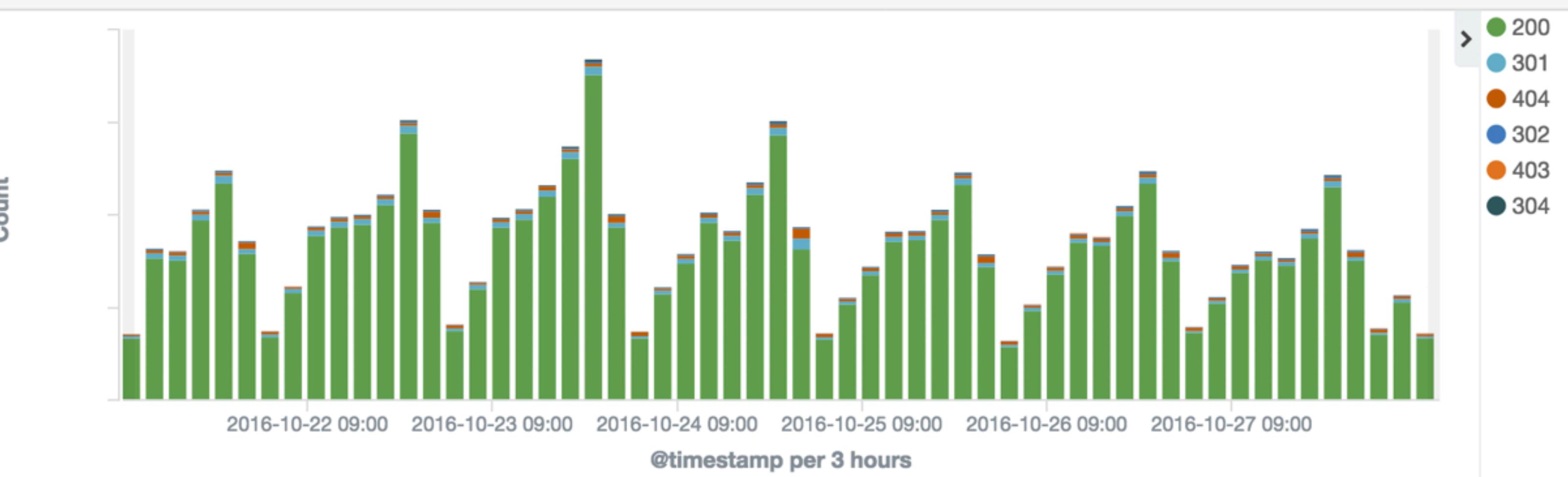
[mwed/infra] Pull request submitted by [REDACTED]



インフラ振り返りダッシュボード

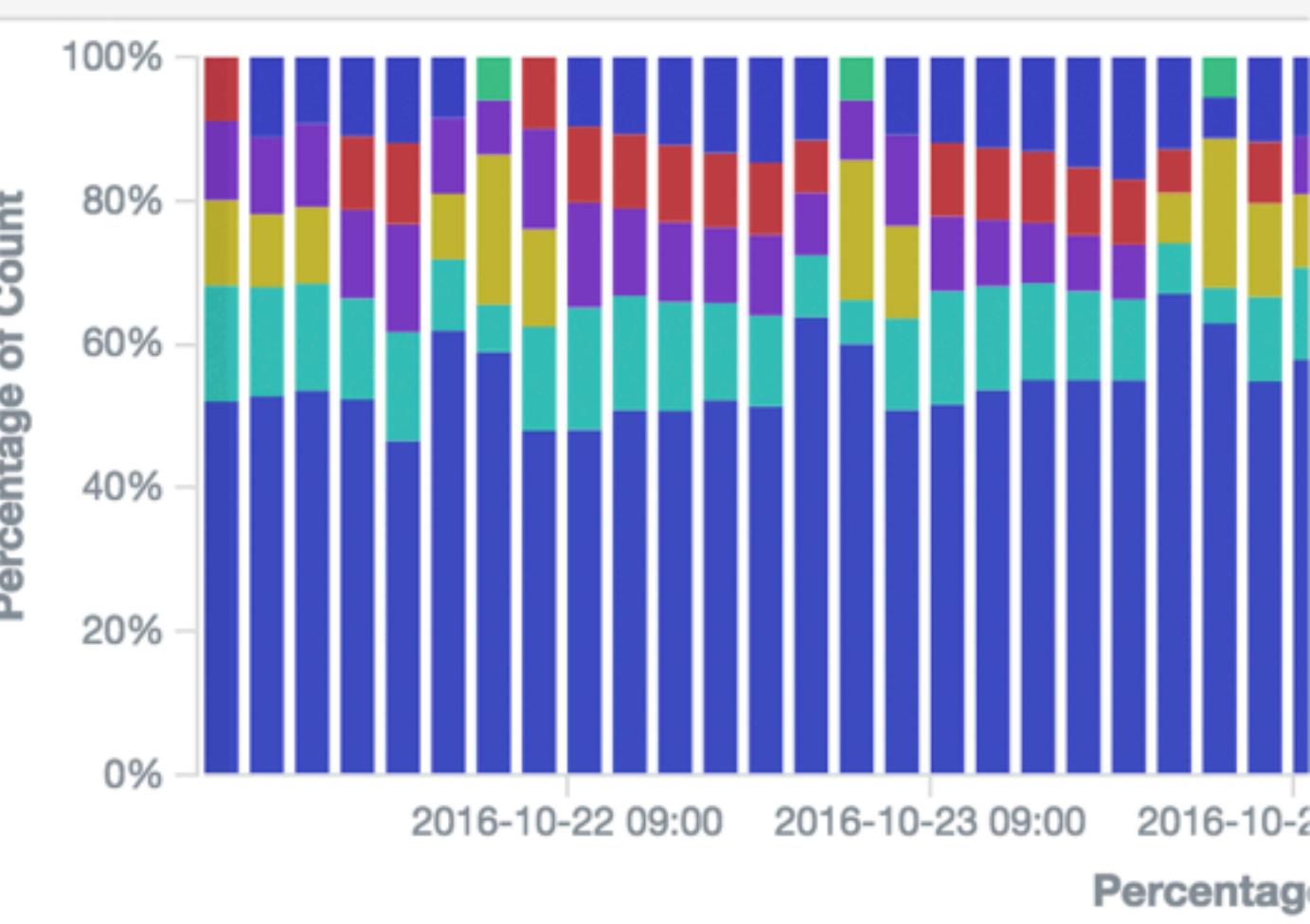
*

request count by status



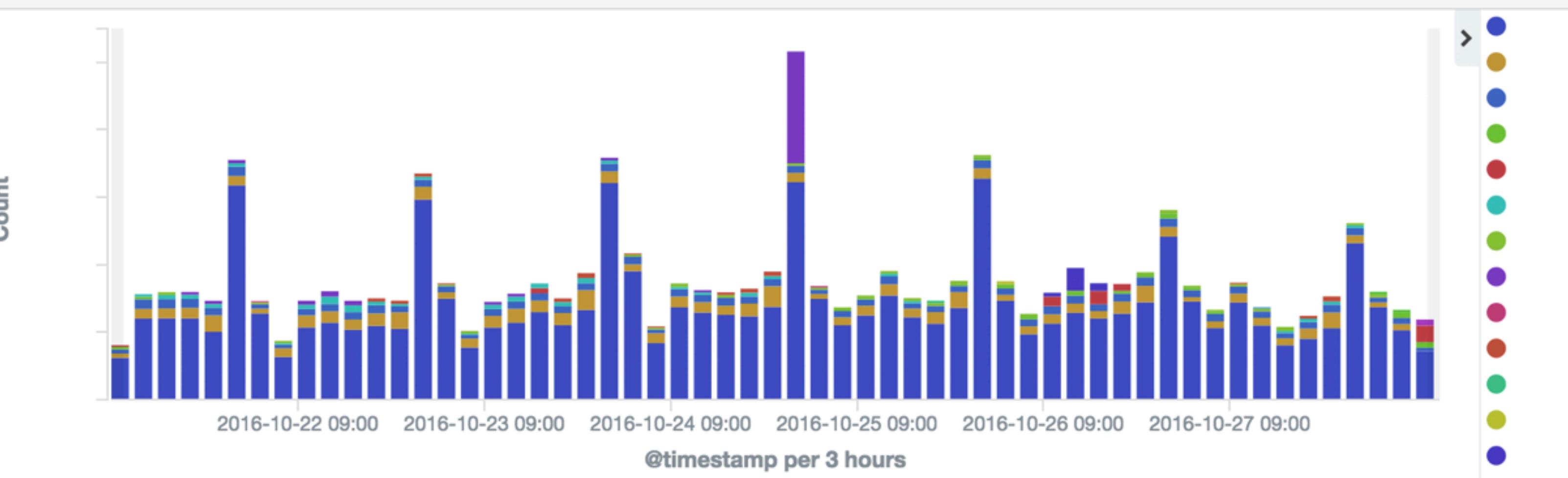
edit ×

request rate by top_level_path



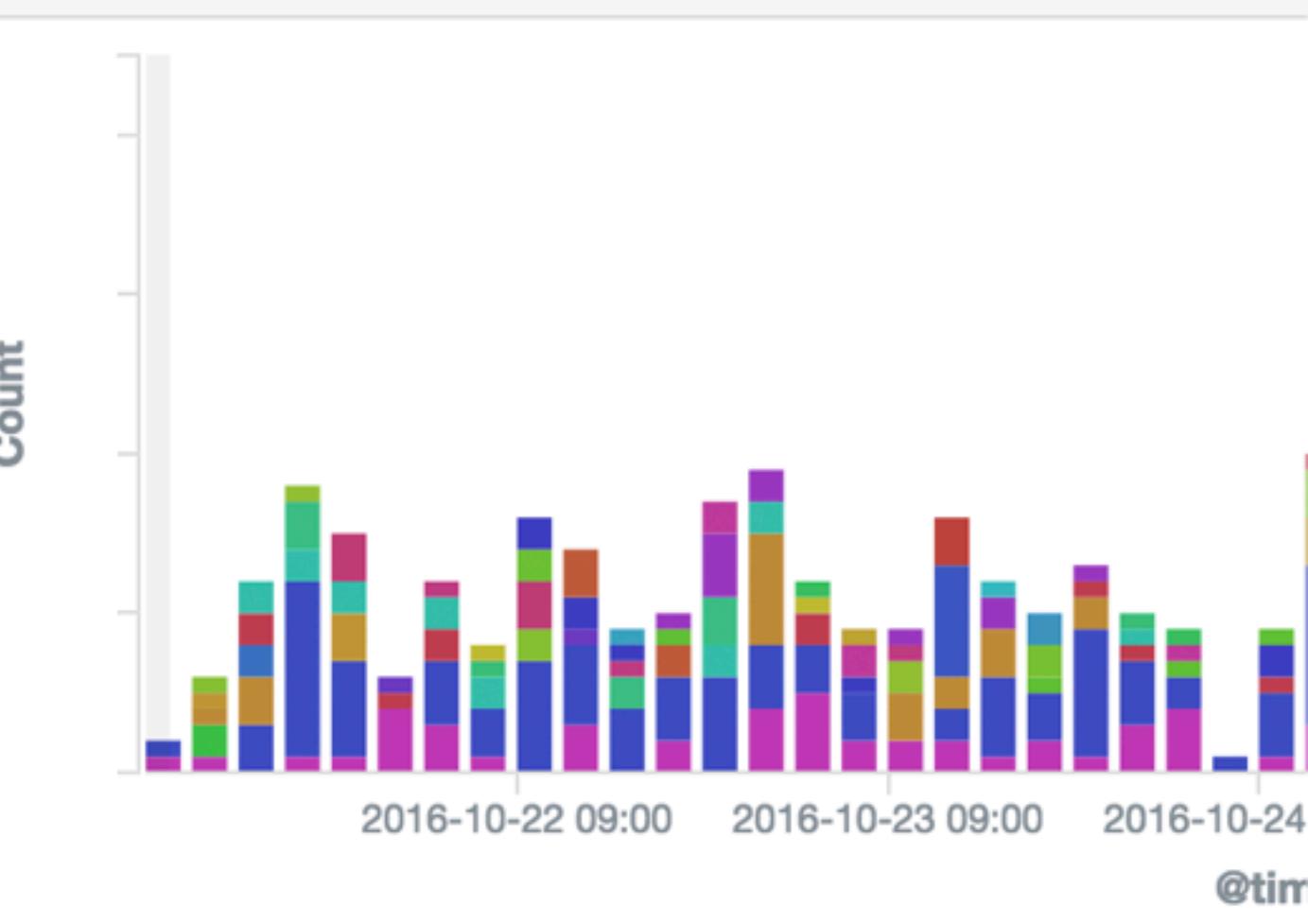
Percentag

4xx count by status



edit ×

5xx count by status



@tim

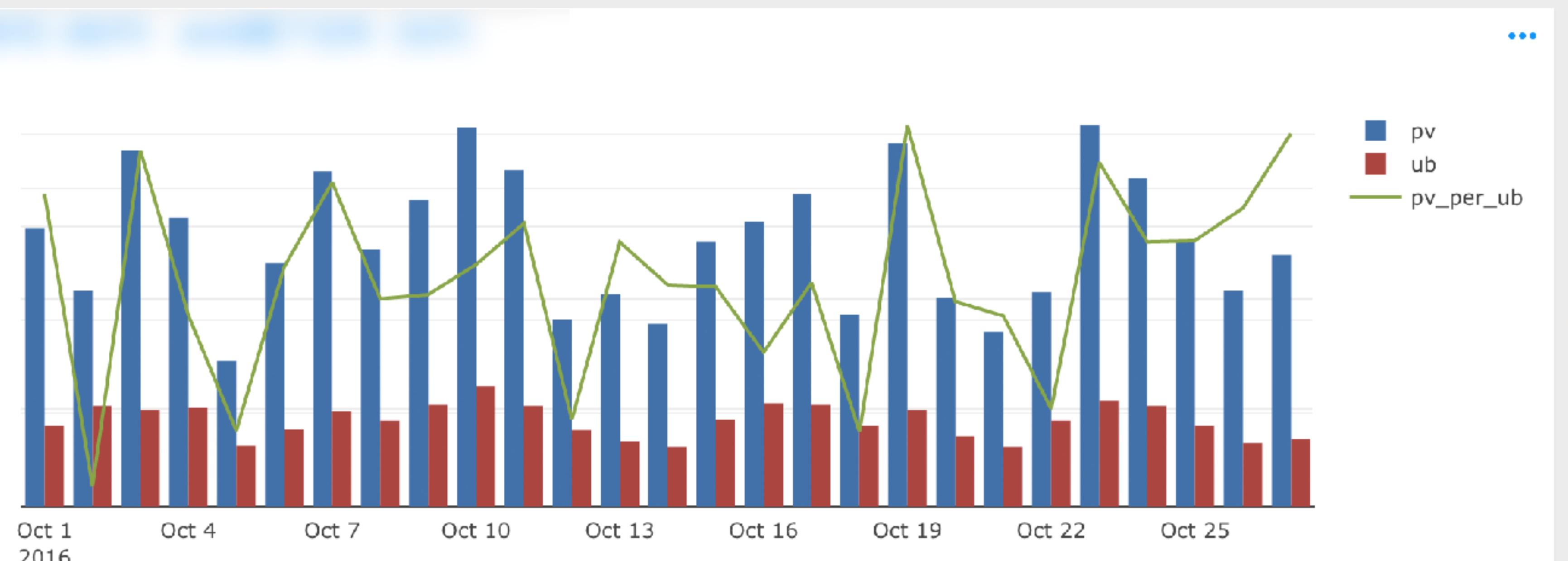


Dashboards ▾

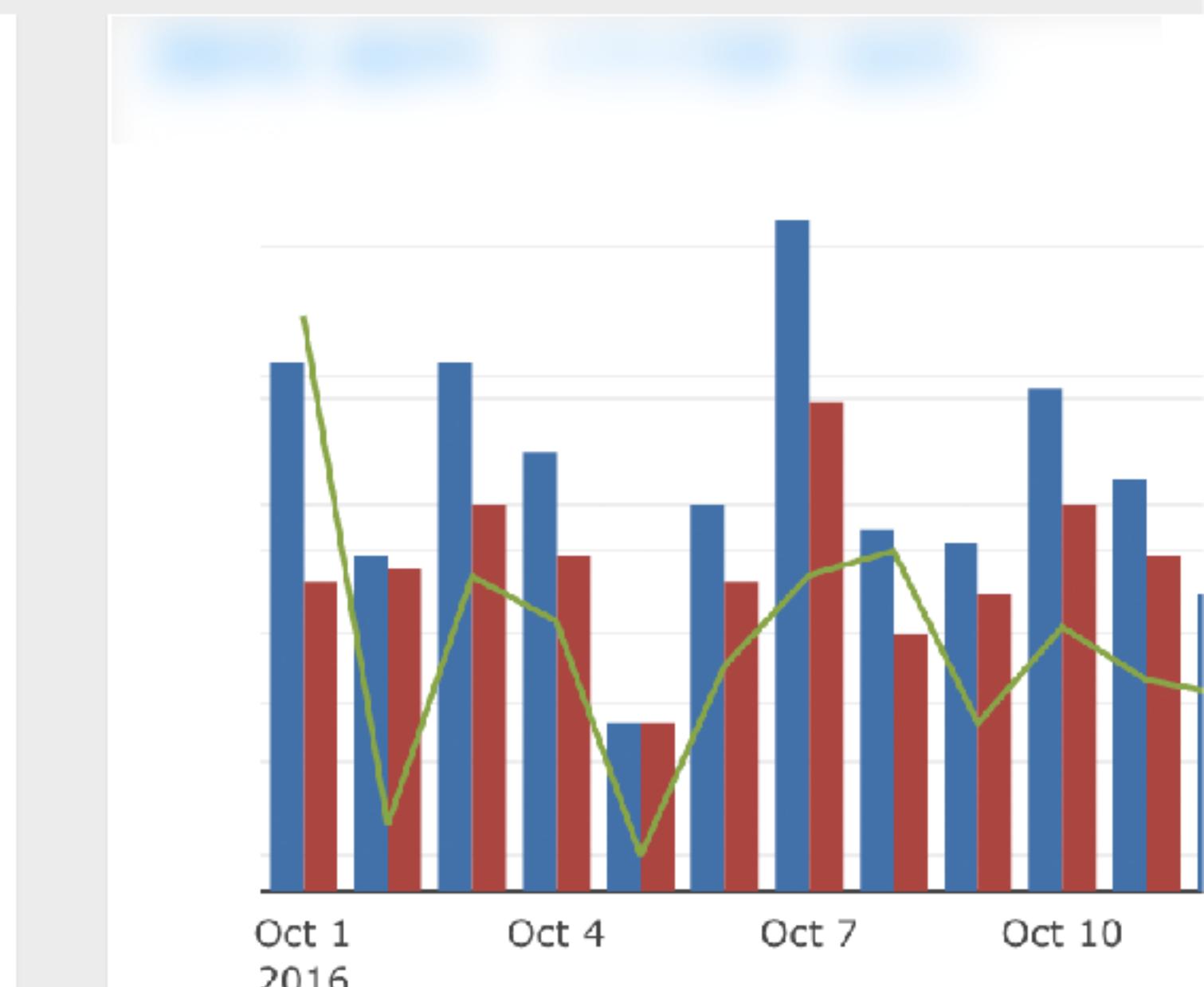
Queries ▾

Alerts

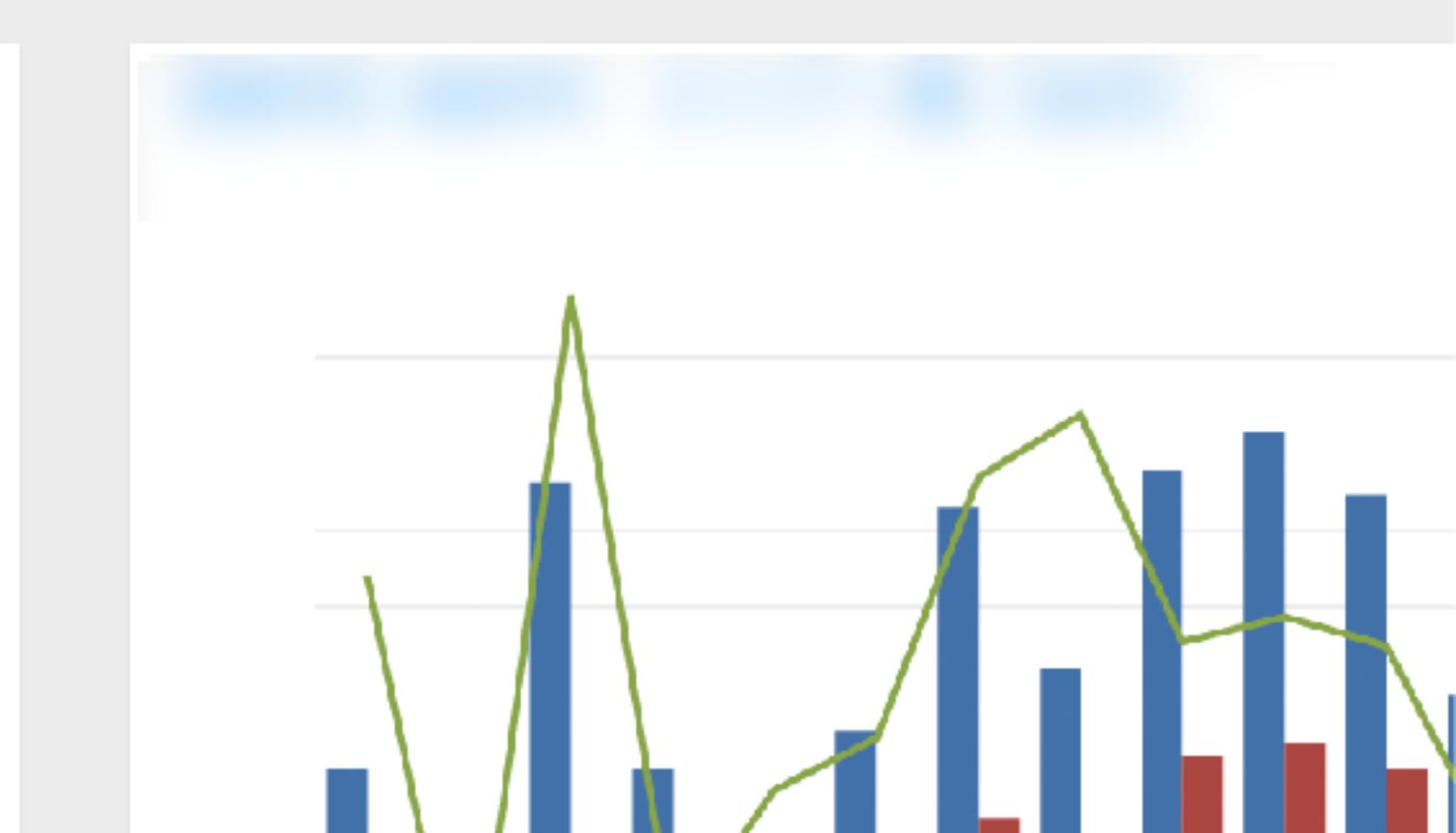
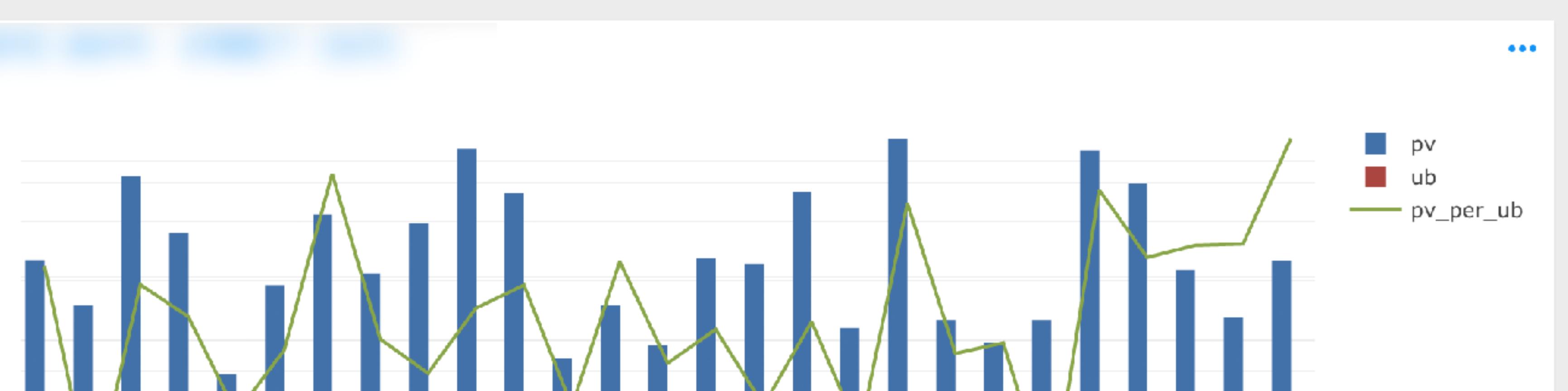
Search queries...



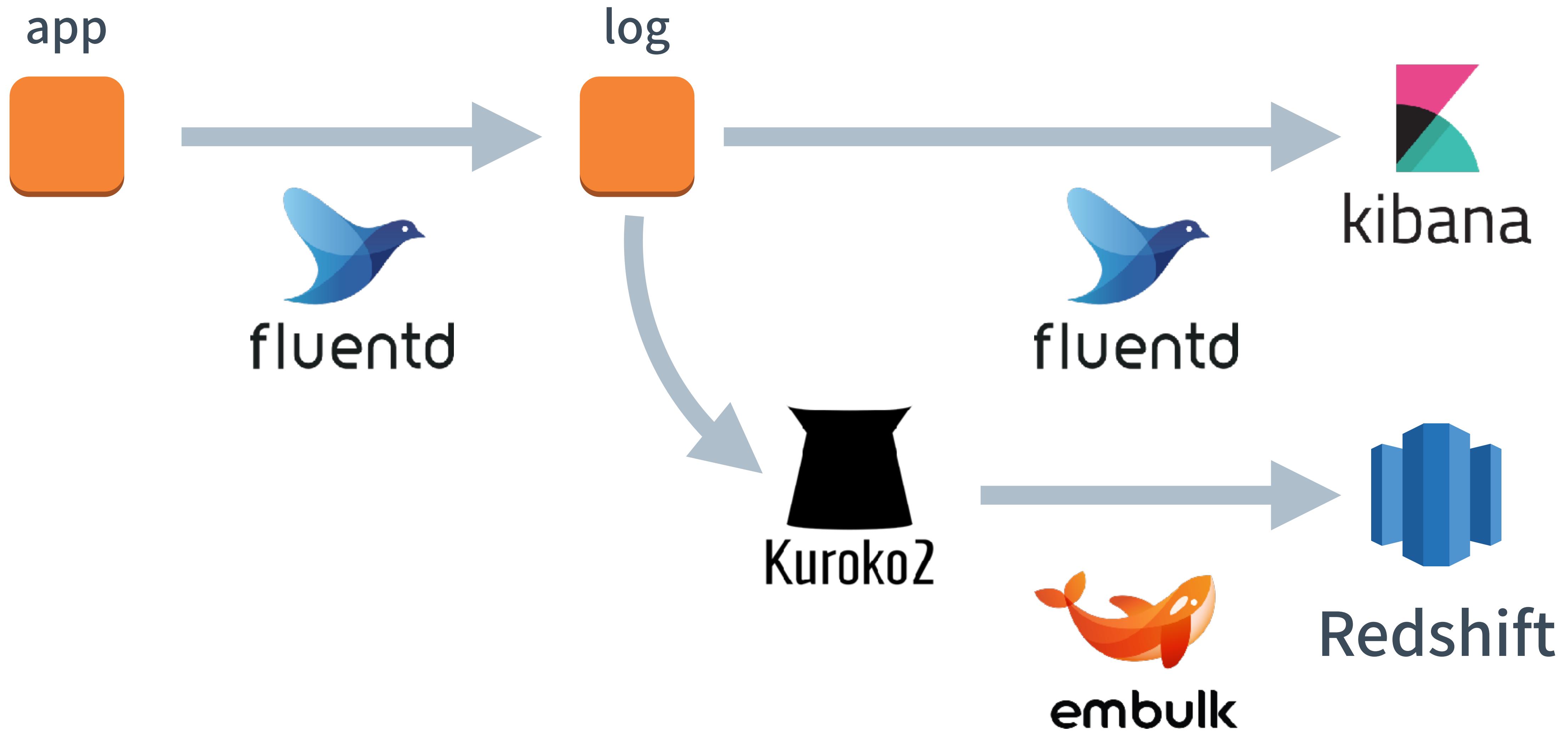
Updated: 4 hours ago



Updated: 4 hours ago



Rubyによるログの可視化





Zabbix BOT 10:13

PROBLEM: kuroko2-staging-001: kuroko2-schduler is down

Date / Time

2016.10.27 / 10:13:07

Status

PROBLEM

Host

kuroko2-staging-001

Trigger

kuroko2-schduler is down: 0

Failure Comments

<http://zabbix-001.mwed.work/zabbix/acknow.php?eventid=466597>

10:13 ★

PROBLEM: kuroko2-staging-001: kuroko2-processor is down

Date / Time

2016.10.27 / 10:13:16

Status

PROBLEM

Host

kuroko2-staging-001

Trigger

kuroko2-processor is down: 0

Failure Comments

<http://zabbix-001.mwed.work/zabbix/acknow.php?eventid=466598>

OK: kuroko2-staging-001: kuroko2-processor is down

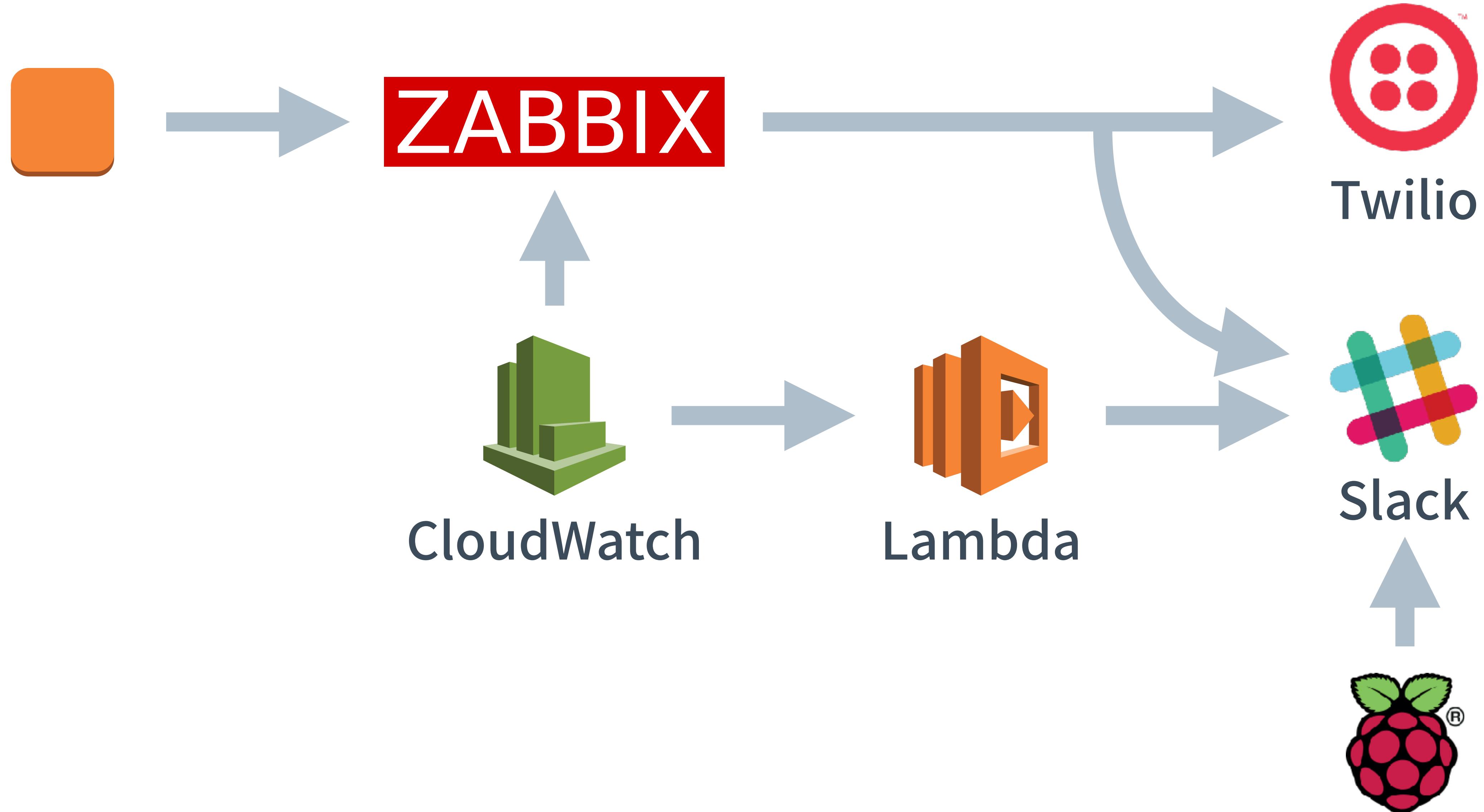
Date / Time

2016.10.27 / 10:14:46

Status

OK

監視の自動化





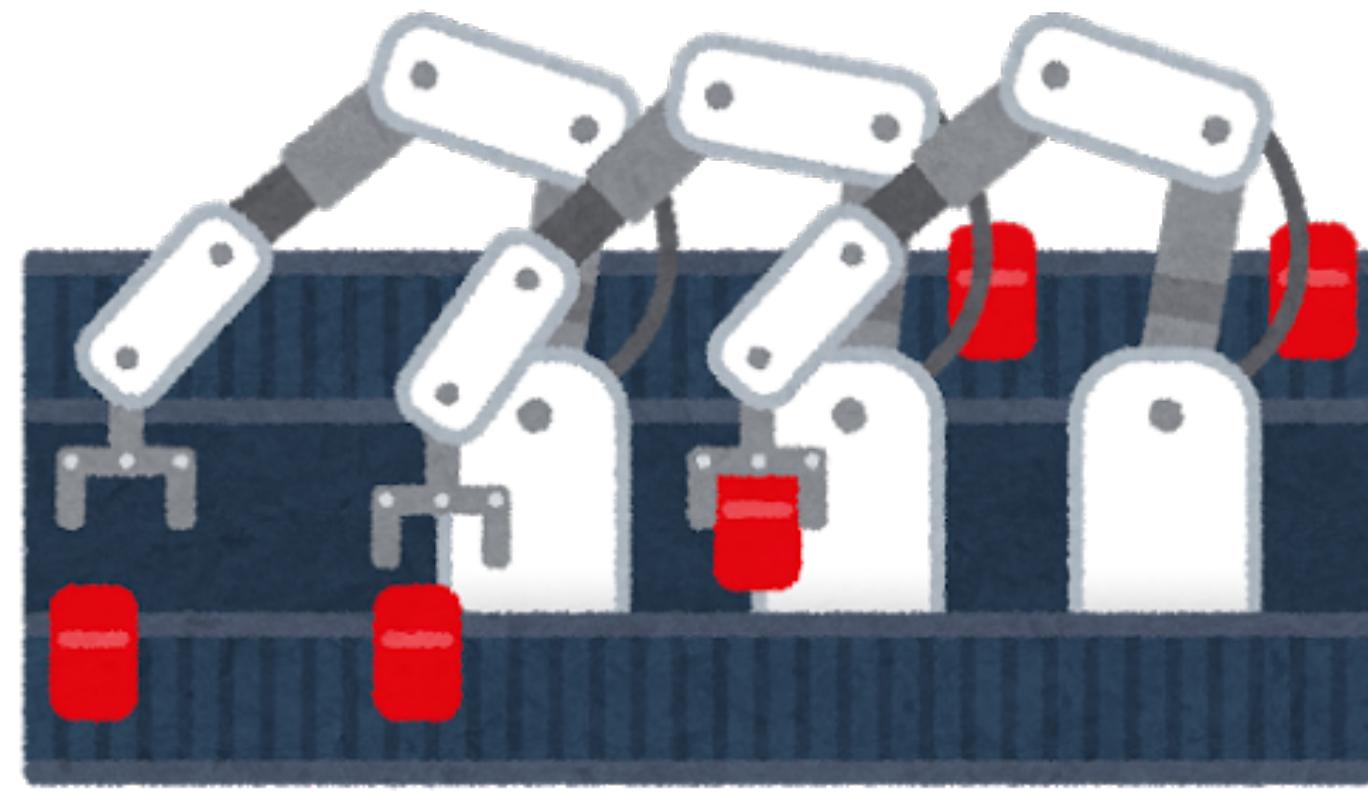
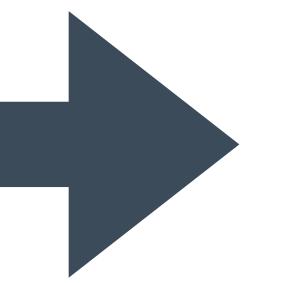
- Raspberry Pi 3
- USBスピーカー
(MM-SPU8BK)

音声でインフライベントを
通知する仕組み
(CV: 姫野つばさ)

```
module Mina
  module Handlers
    class Announce < Ruboty::Handlers::Base
      BASE_DIR = File.join(__dir__, '../..../audio')

      on(/\A\[(High|Disaster)\]/i, name: 'alert', all: true)

      def alert(message)
        Thread.start { invoke_command("mpg123 -q #{BASE_DIR}/#{file_name}")}
      end
    end
  end
end
```

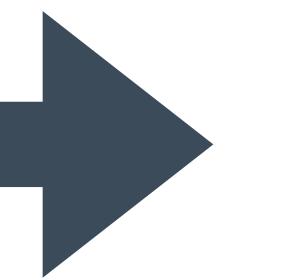


今までの開発

これからの開発



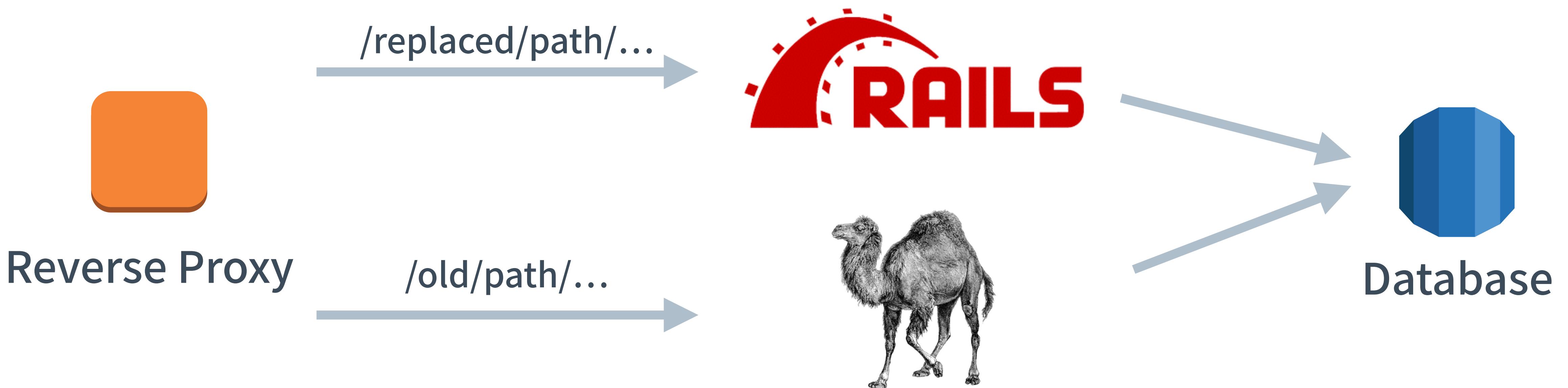
今までの開発



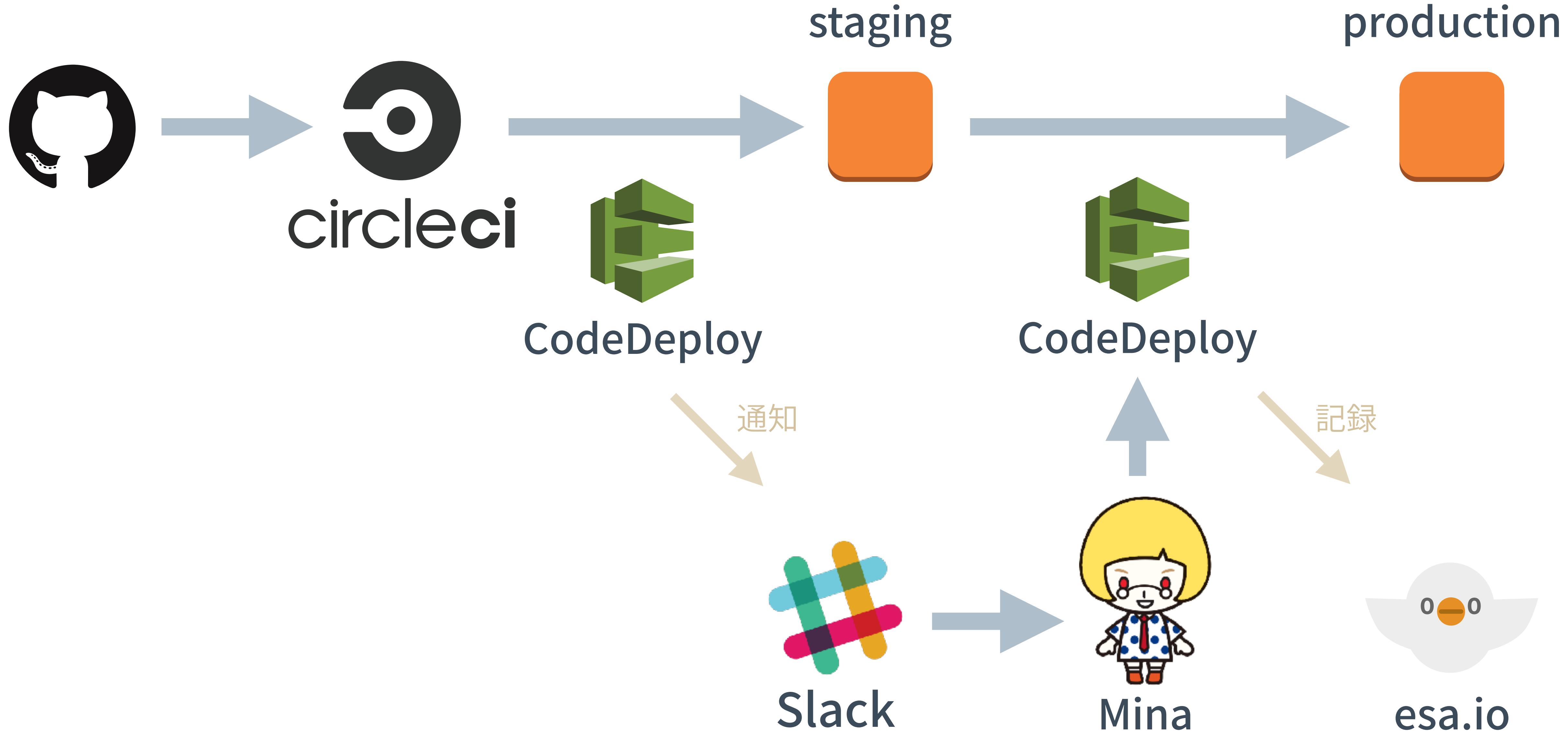
これからの開発

Ruby on Railsへの移行戦略

- MobaSiFとRuby on Railsでデータベースを共有する
- リバースプロキシを設置し、URL単位で切り替えていく



ビルドプロセスの自動化





Speaker Deck

Search...

Browse

Upload



Naoto Takai

Speaker Deck

Published on Nov 21, 2013

Continuous Delivery in COOKPAD

RubyWorld Conference 2013



◀ ▶ ⏪ ⏩ share

Continuous Delivery in COOKPAD by Naoto Takai

Published November 21, 2013 in Programming

at RubyWorld Conference 2013, on Nov. 21st, 2013.



Naoto Takai

11 Presentations

★ Star this Talk

37 Stars

Published in

Programming

Stats

26,618 Views

Edit this presentation

Share

Twitter, Facebook

Embed

Direct Link

Download PDF

[Applications](#) [Service maps](#) [Key transactions](#) [Alerts](#) APPS
weddingTIME PICKER
Last 30 minutes ending now ▾

MONITORING

Overview

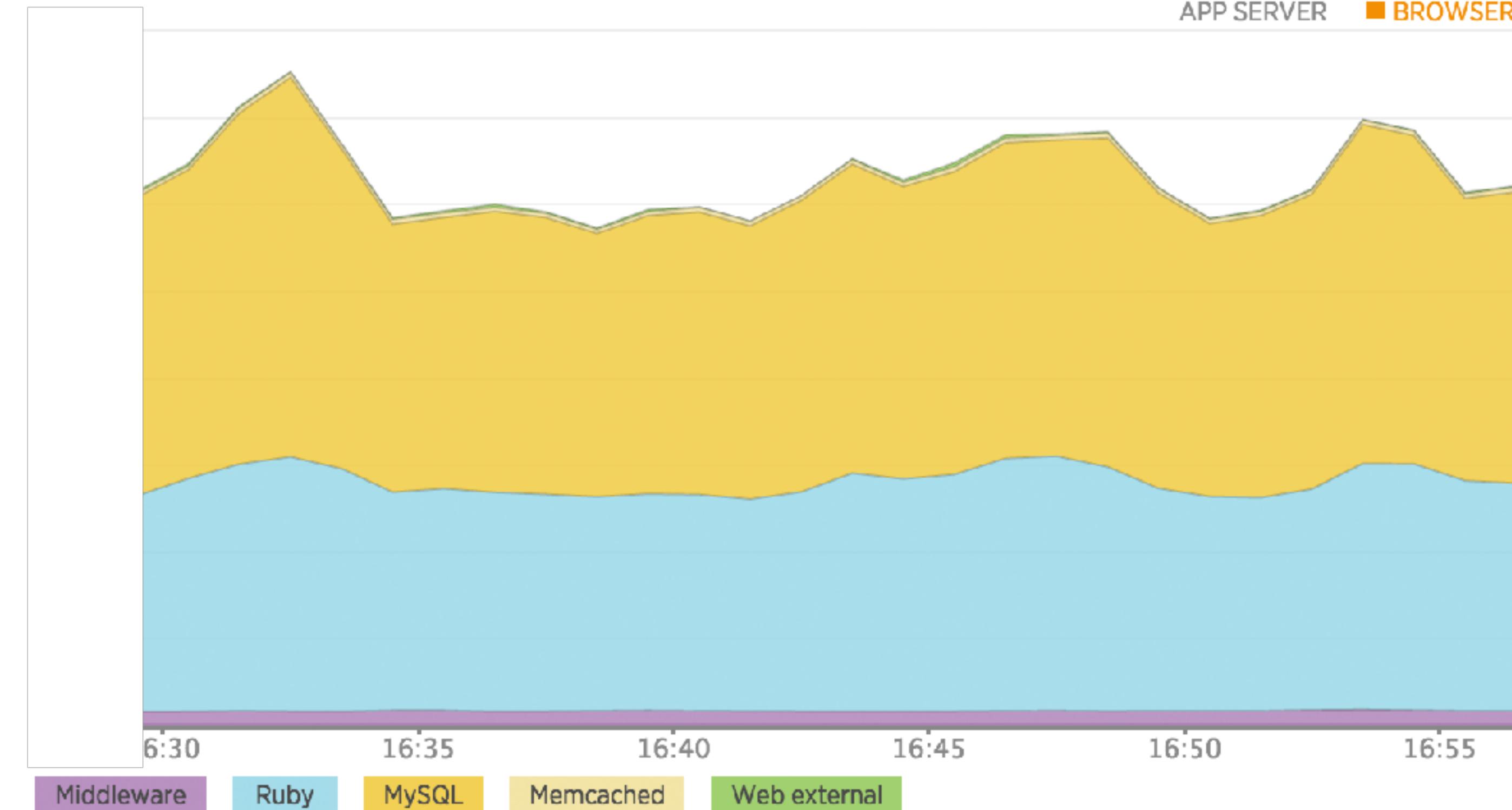
[Service maps](#)[Transactions](#)[Databases](#)[External services](#)[Ruby VMs](#)

EVENTS

[Error analytics](#)[Errors](#)[Alerts](#)[Deployments](#)[Thread profiler](#)

REPORTS

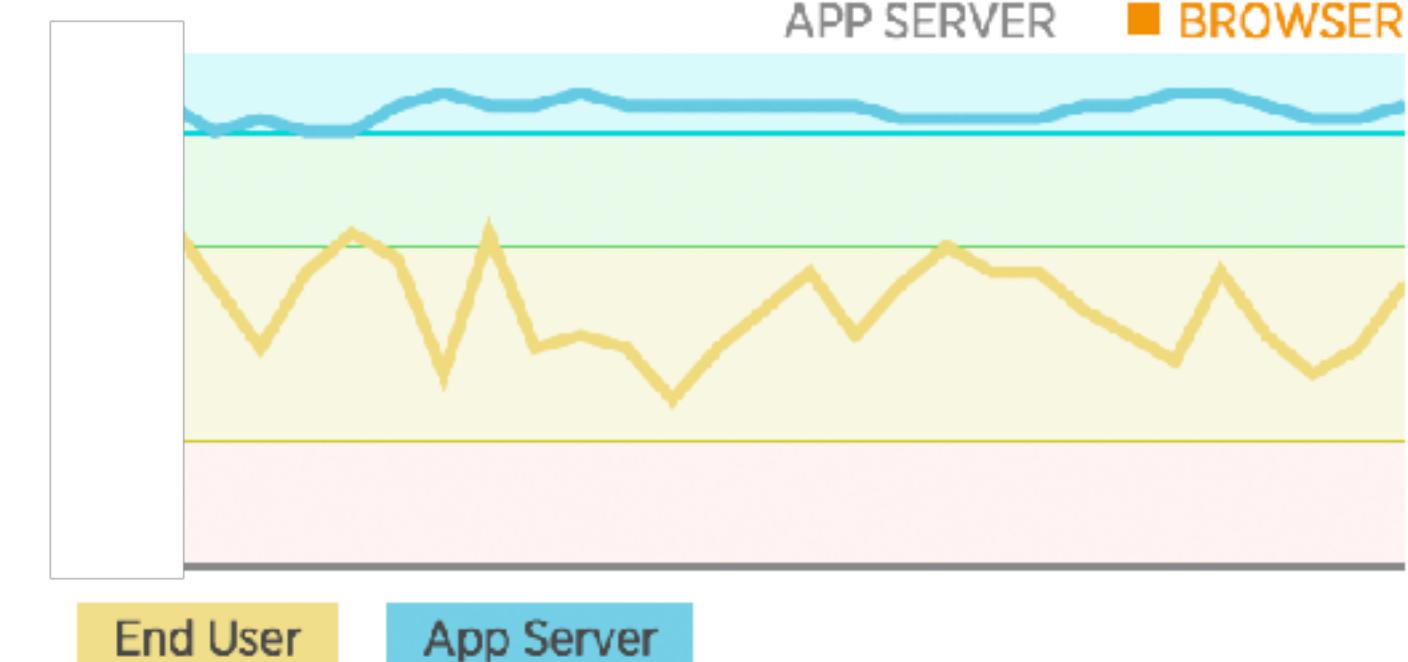
Web transactions time ▾

ms
APP SERVER
s
BROWSER Compare with yesterday and last week[Transactions](#)[App server time](#)[Error rate](#)

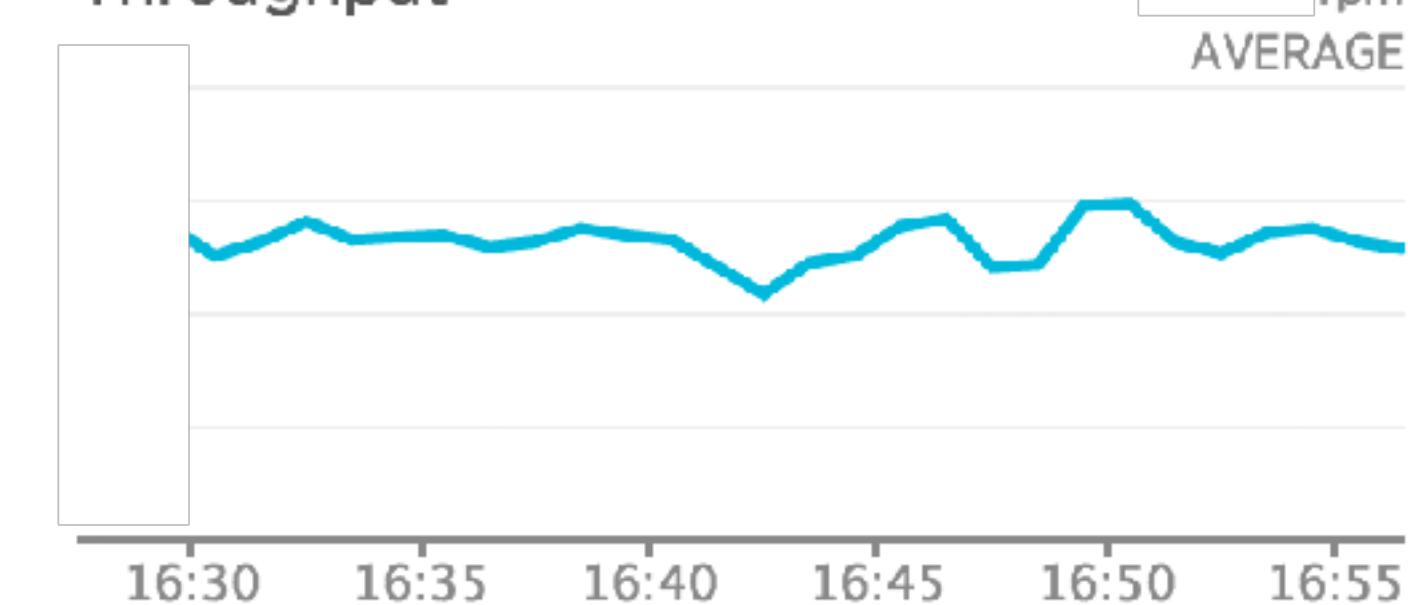
0.00 %

Member of policy Default application alert policy

Apdex score ?

APP SERVER
BROWSER

Throughput

rpm
AVERAGE



ORGANIZATION



Dashboard



Projects & Teams



Stats



ISSUES

Assigned to Me

Bookmarks

History

MANAGE

Members

Auth

API Keys

Audit Log

Rate Limits



Assigned to me

[View more](#)

No issues have been assigned to you.

New this week



ActionView::Template::Error app/models/...

wedding undefined method `+' for nil:NilClass

ActionView::Template::Error app/models/...

wedding undefined method `+' for nil:NilClass

ActionView::Template::Error app/models/...

wedding undefined method `+' for nil:NilClass

NoMethodError app/models/place.rb in rev...

wedding undefined method `+' for nil:NilClass

ActionView::Template::Error lib/wedding/...

wedding undefined method `pref_name' for nil:Nil...

EVENTS PER HOUR

[View Stats](#)

PROJECTS

[View All](#)

Bookmark your most used projects to have them appear here.

wedding

Wedding Developers



Recent activity



+

Success: [REDACTED] build (#15696; push in mwed/wedding (pull/4894))

-



github BOT 16:31

[mwed/wedding] Pull request closed:



Sentry BOT 16:32

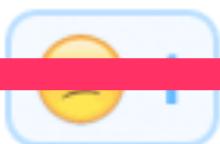
NameError: uninitialized constant HellooooBatch

Culprit

bin/rails in require at line 4

Project

Wedding Developers wedding



github BOT 16:37

[mwed/wedding] New comment by [REDACTED] on pull request [REDACTED]

修正しました！

[mwed/wedding] New comment by [REDACTED] on pull request [REDACTED]

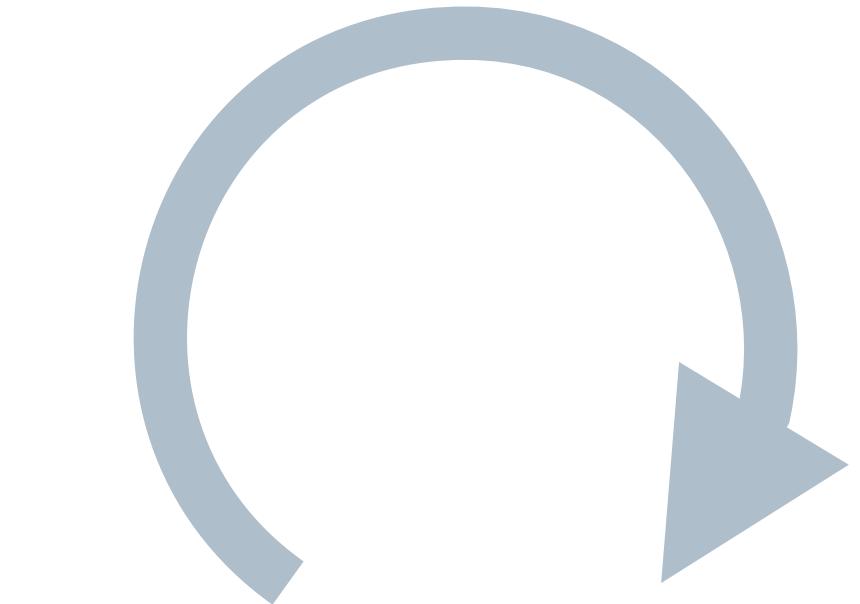


アジャイル開発のお約束

- 声だしていこう
- 約束をまもろう



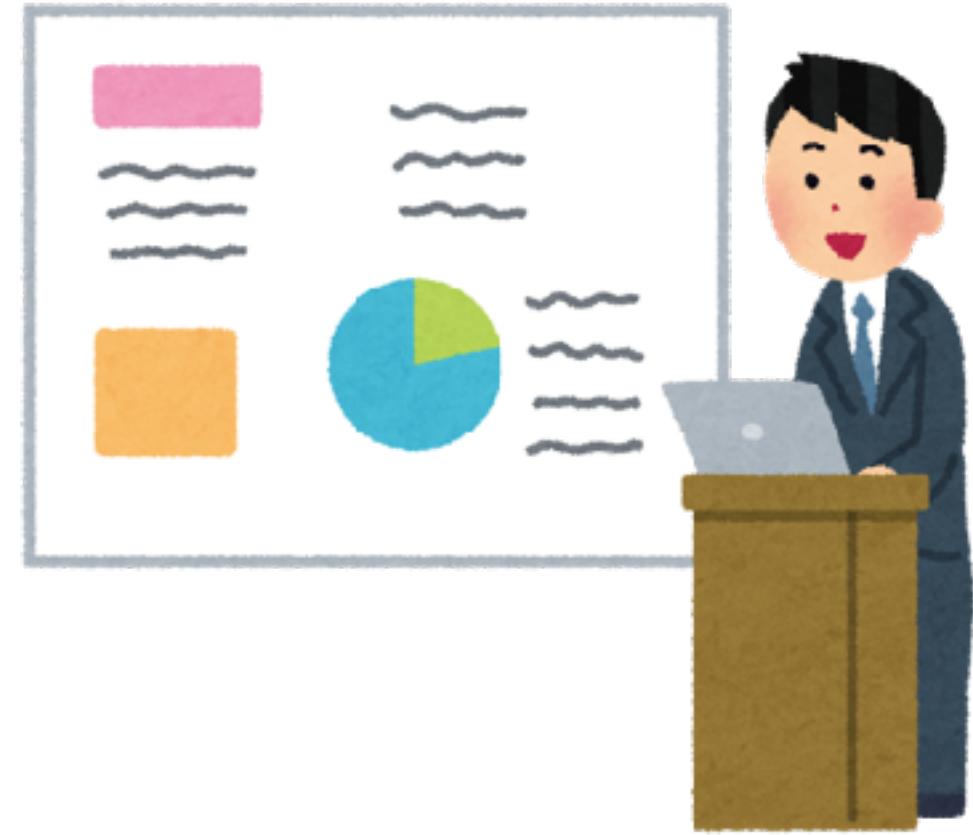
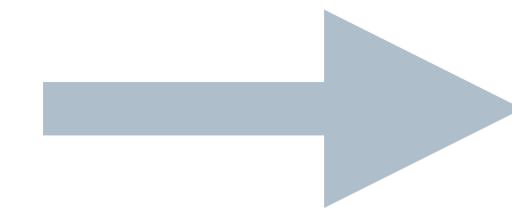
(月) 計画



(毎日) 朝会

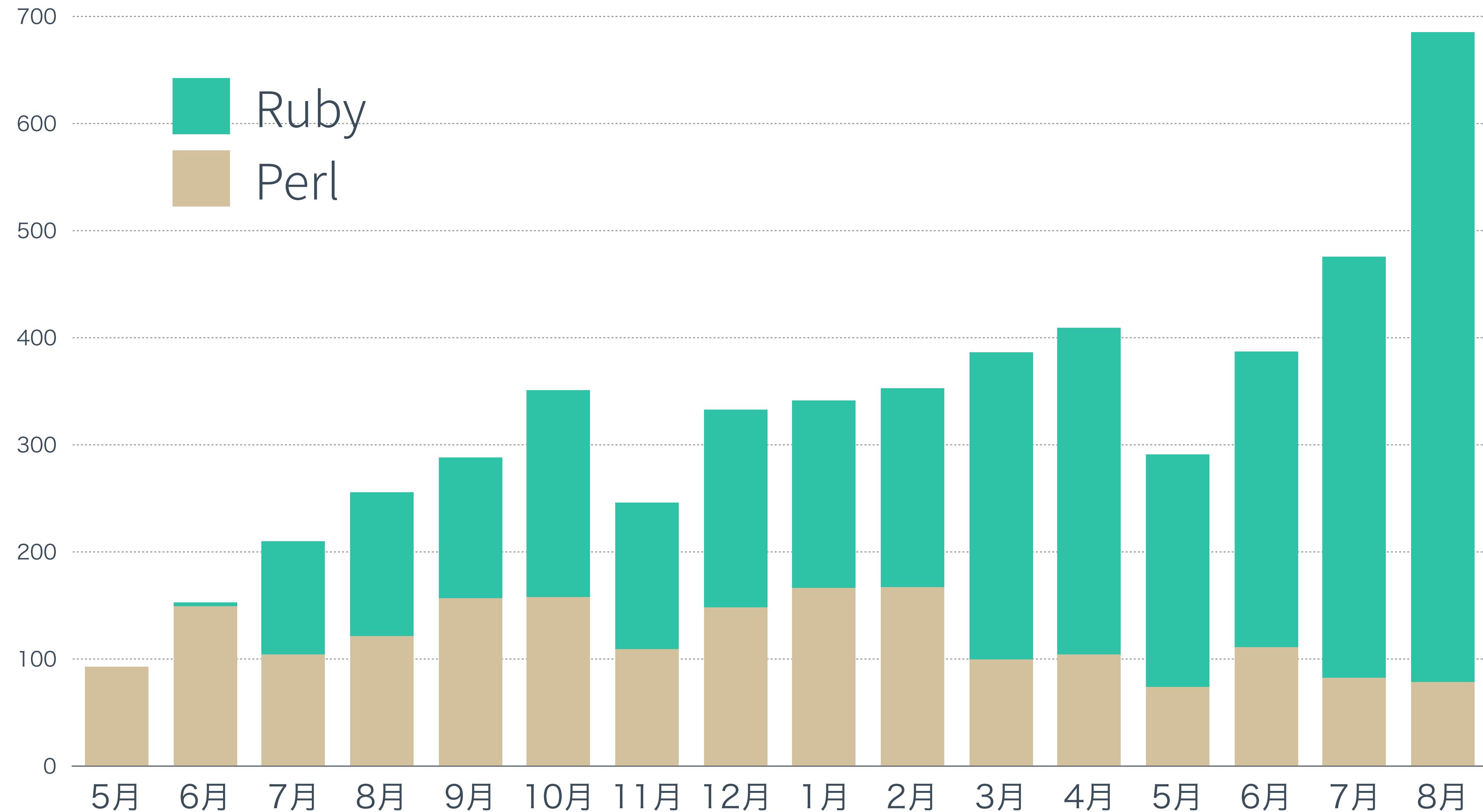


(木) ふりかえり



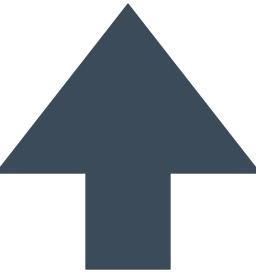
(金) スプリントレビュー

GitHub プルリクエスト数推移

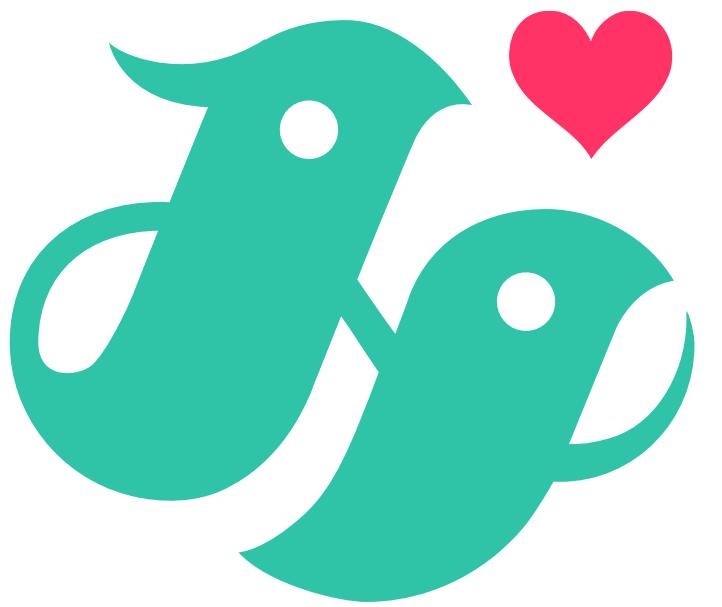


エンジニアリングの責任者として
どのような戦略をとるか？

- AWS とデブオプス
- Ruby on Railsとアジャイル開発



Ruby + チーム



みんなの
ウェディング