観測からはじめる
DevOps

8.5 [WED] 14:00-15:00

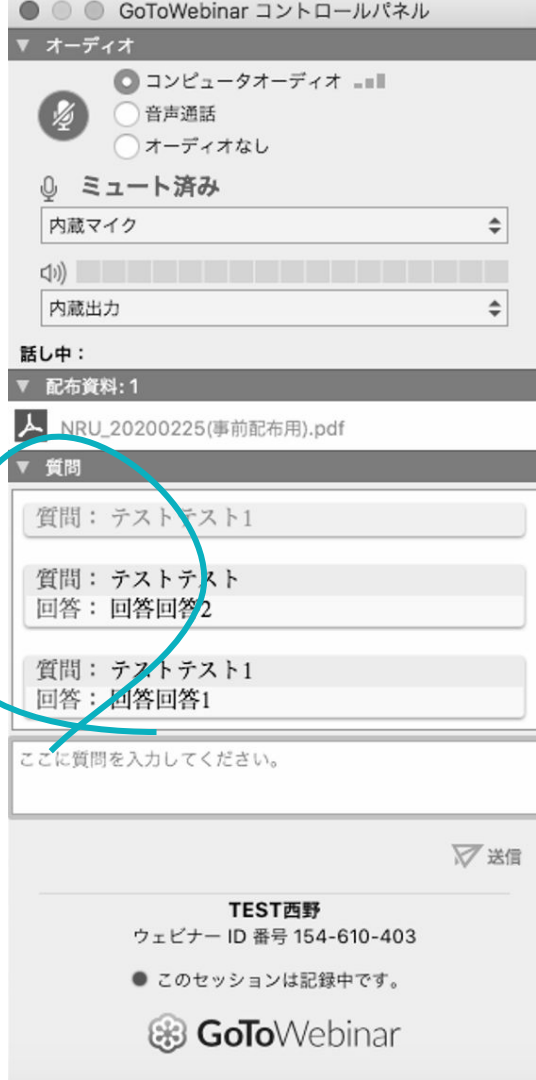
車井 登
Solution Engineer
circleci

東 卓弥
Solutions Consultant
New Relic

ご質問やコメントに関して

質問がある場合はTwitterでハッシュタグ #CircleCIJP #newrelic を付けて投稿をお願いします。

Twitter アカウントをお持ちでない場合は、GoToWebinar の質問ボックスからご質問ください。(質問ボックスからの質問は開催者にのみ表示されます)



自己紹介

名前：車井 登 / Noboru Kurumai

ポジション：Solutions Engineer

経歴：パッケージソフト開発

クラウドサービスエンジニア

New Relic®

New Relic と CI/CD

New Relic からみる DevOps の課題

Takuya Azuma
Solutions Consultant

©2008–20 New Relic, Inc. All rights reserved

東 卓弥

Solutions Consultant

NewRelicソリューションコンサルタント。ERPパッケージベンダーにてSaaS製品を開発。SREも担当し運用自動化に励む。その後総合系コンサルティングファームに転職し、BtoCサービスの構築支援としてモバイルアプリを主としたサービスの開発リーダーを担当。アーキテクチャの設計からCICD、バックエンド・フロントエンド開発という全領域の開発に加え、SREで経験した運用も踏まえたアプリケーションを中心としたパフォーマンス管理・チューニングを得意とする。業務上経験が多いのはJava、Javascript。

JJUG CCC 2016 Javaのパフォーマンスチューニング
AWS Dev Day Tokyo 2017 バッチ処理の高速化に AWS Lambdaを使った事例

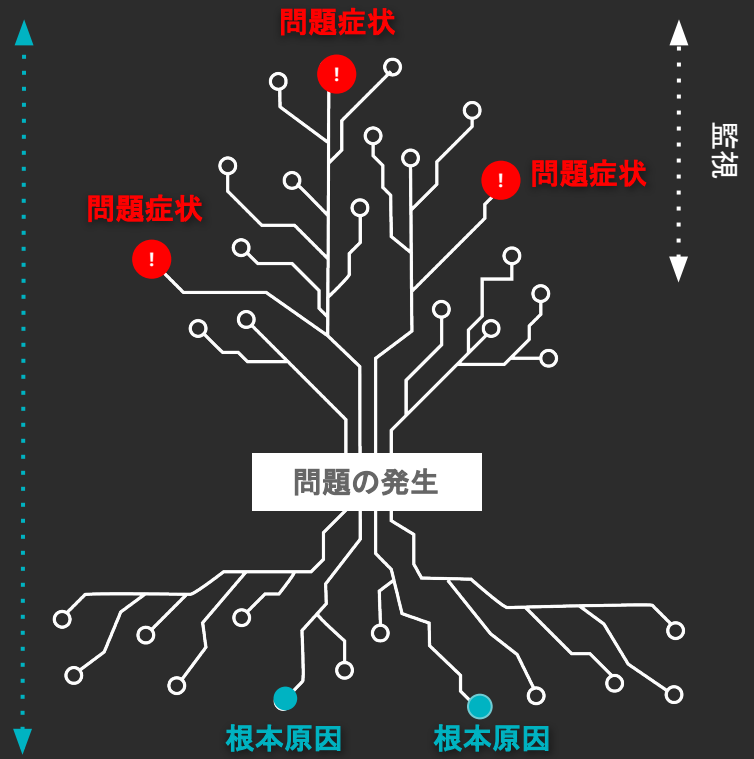
キーワード:オブザーバビリティ

©2008–20 New Relic, Inc. All rights reserved

監視からオブザーバビリティの時代へ

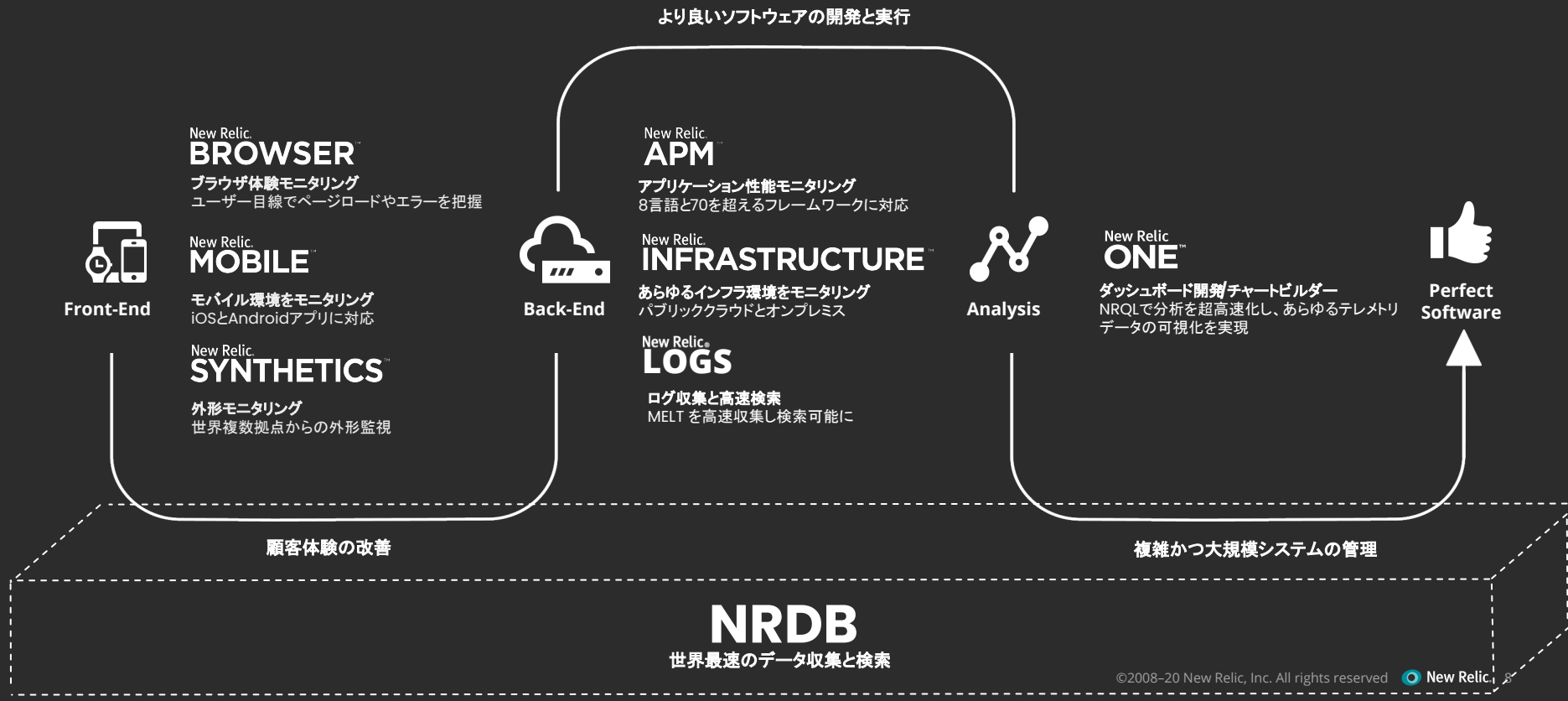
オブザーバビリティ
Observability

とは、デジタルサービスを支えるシステムのメトリクス・イベント・ログ・トレースのデータをリアルタイムに取得し続け、常に状態把握と改善できる状態にあることを指す。



©2008–20 New Relic, Inc. All rights reserved

New Relic Observability Platform



©2008–20 New Relic, Inc. All rights reserved

©2008–20 New Relic, Inc. All rights reserved New Relic.

成長企業とその他の企業の差

208
TIMES MORE
frequent code deployments

デプロイ頻度

106
TIMES FASTER
lead time from
commit to deploy

コミットからデプロイまでの時間

2,604
TIMES FASTER
time to recover from incidents

障害からの復旧時間

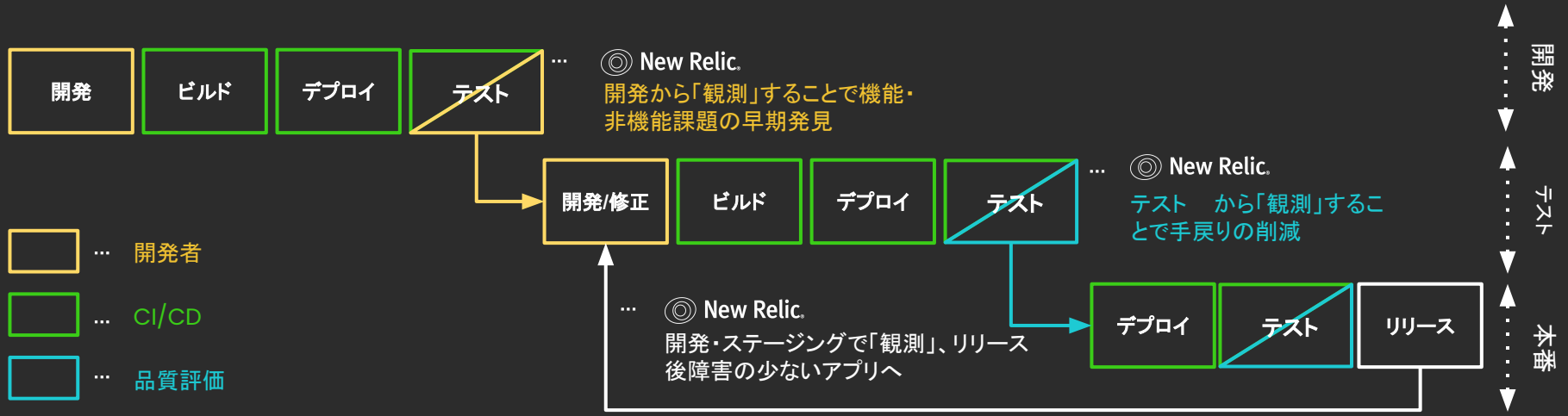
7
TIMES LOWER
change failure rate
(changes are 1/7 as likely to fail)

変更時のエラー発生率

Throughput
Stability

© 2008-20 New Relic, Inc. All rights reserved
New Relic.

デプロイ頻度向上のためのライフサイクル



circleci

パイプラインを自動化し、正確にミスなく流せるようにすることでデプロイ作業全体を短縮化

＋

New Relic

テストフェーズやリリース後の問題早期発見で手戻り少なく、デプロイごとのアプリ性能を観測し信頼性向上。

©2008–20 New Relic, Inc. All rights reserved

CircleCIはアイデアを製品化するための プラットフォーム

10 年前

ソフトウェアプロジェクトは、
ほとんどが手作りのカスタムコードでした

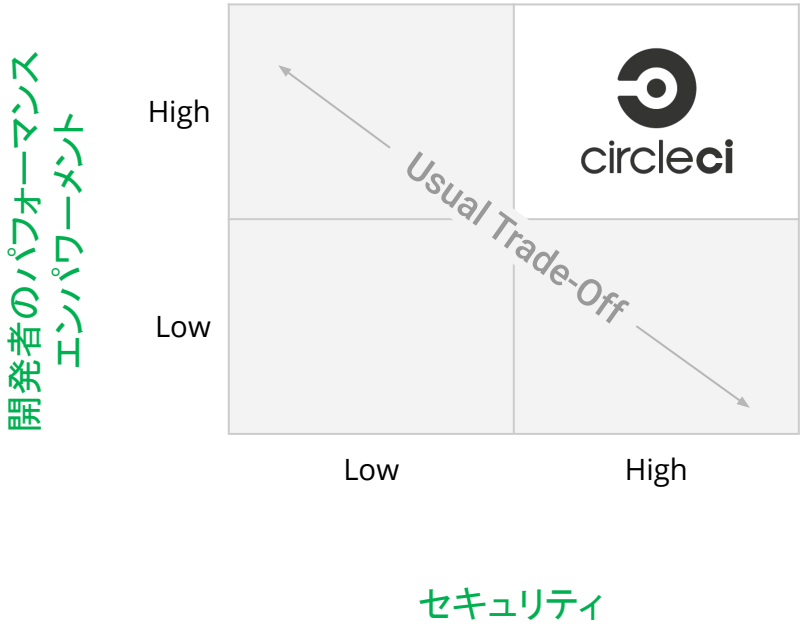
今日

マイクロサービス、Docker、および共有オープンソースライブラリは、途方もない規模を生み出しますが、制御しきれない複雑さも生み出します。

circleci

Move fast without breaking things

最高のチームはスピードとコントロールのどちらを選ぶのではなく、両方を手に入れます。

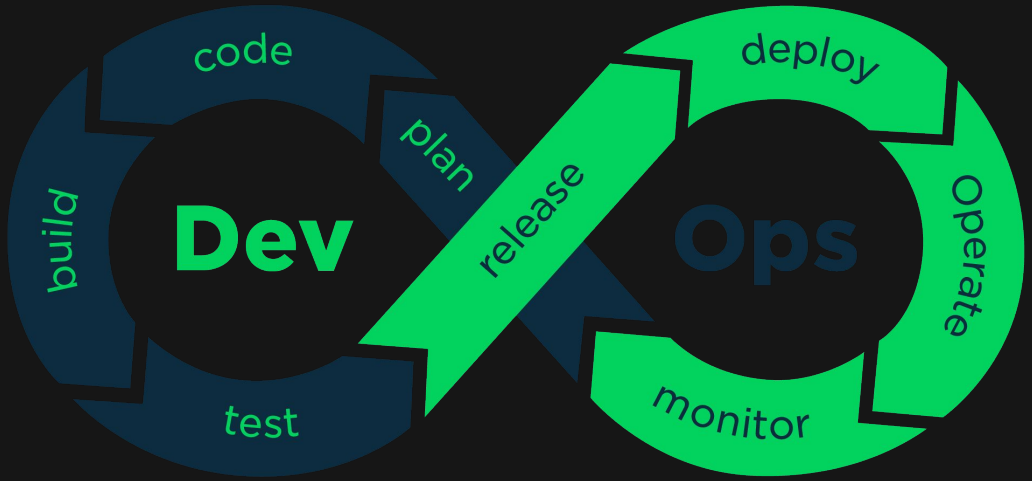


“

Poll 1: お使いのソースコード管理ツールはなんでしょうか？

”

What is CI


code
plan
build
Dev
test
release
deploy
Operate
Ops
monitor

How to CI

- 常に測定できること
- 必要なビルドプロセスとテストを設定すること
- 可能な限り自動化すること
- 素早く頻繁にmasterにマージすること
- 日々のコミット、日々の結合

CIのメリット

- より高品質で安定した製品がリリースできる
- チームの生産性と効率が向上する
- 開発者が幸せになる
- 平均復旧時間の短縮
- より多くの頻度でデプロイ（リリース）できる

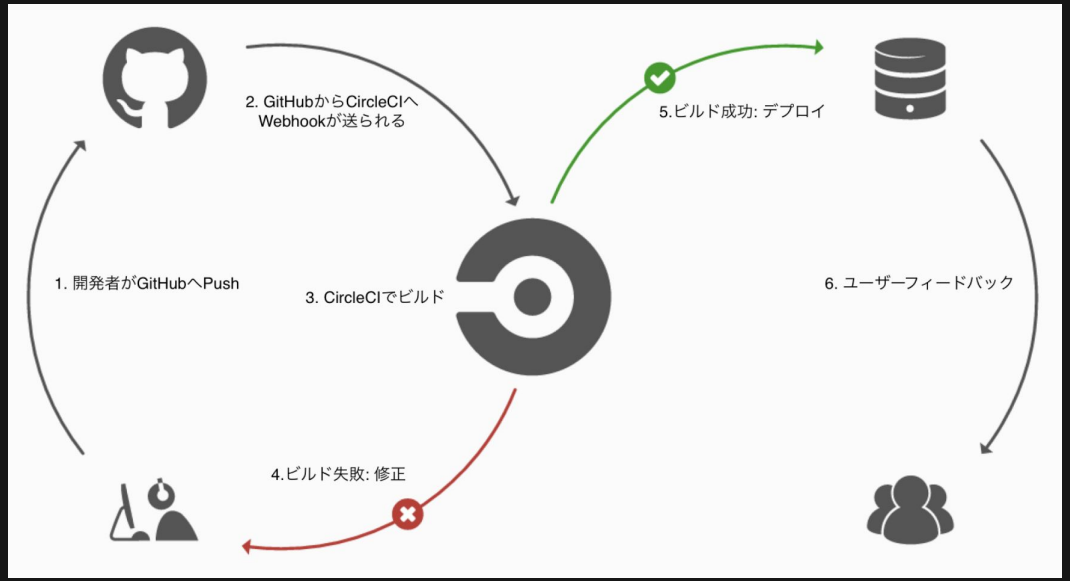
“

Poll 2: お使いのCI/CD製品はなんですか？

”

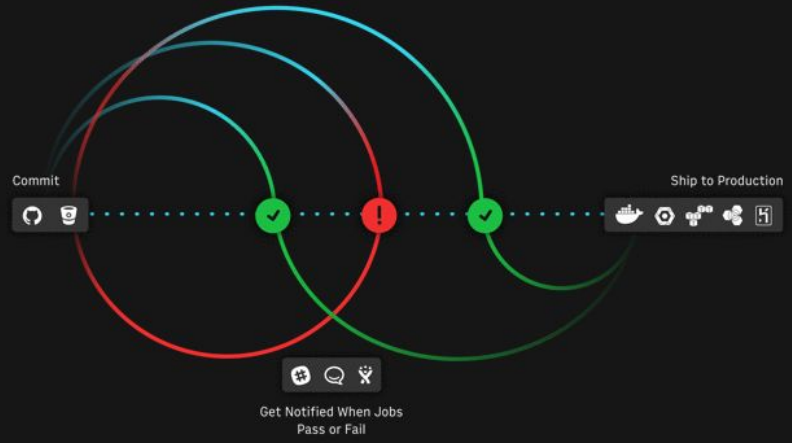
CircleCI の基本

- すべてのコミットをトリガーに、
  自動テスト、自動ビルドが実行されます。
  - スケーラブル
  - 決定論的(Deterministic)
  - 柔軟性



CircleCIの基本

- もしビルドやテストが失敗した場合は、すばやくリカバリすることができます
  - VCSとのインテグレーション
  - 失敗した箇所のログ
  - 結果の通知
  - SSHデバッグ



設定

Continuous Integration with CircleCI

- 設定ファイルはソースコードと同様に管理
- config.ymlに実行したいステップを記述

```yaml
version: 2.1
orbs:
  node: circleci/node@1.1.6
jobs:
  build-and-test:
    executor:
      name: node/default
    steps:
      - checkout
      - node/with-cache:
          steps:
            - run: npm install
            - run: npm test
workflows:
    build-and-test:
      jobs:
        - build-and-test
```

Orbs

Orb Quick Start Guide

1. Use CircleCI version 2.1 at the top of your .circleci/config.yml file.

```
version: 2.1
```

Copy This Code

If you do not already have Pipelines enabled, you'll need to go to Project Settings -> Advanced Settings and turn it on.

2. Add the `orbs` stanza below your version, invoking the orb:

```
orbs:
  artifactory: circleci/artifactory@1.0.0
```

Copy This Code

3. Use `artifactory` elements in your existing workflows and jobs.

Usage Examples

build-integration-command

Install and configure the JFrog CLI, then upload build information (includes git and environment info by default) to JFrog Artifactory

```
jobs:
  build:
    docker:
      - image: 'circleci/node:10'
    steps:
      - checkout
      - artifactory/install
      - artifactory/configure
      - artifactory/build-integration
orbs:
  artifactory: circleci/artifactory@1.0.0
version: 2.1
```

CircleCIの特徴

- Build Intelligence
  - さまざまなサイズ（スペック）のコンテナ/VMによってビルド可能
  - Orbsを使ってベストプラクティスをすばやく取り込める
    - 各種クラウドに応じたデプロイ用Orbsを取り揃えている
- Test Intelligence
  - 過去のテスト結果を活用したテスト並列処理
  - CircleCI公式Dockerイメージを使って言語に応じたビルド環境をすばやく起動
- Change Validation
  - ビルド失敗時にSSHビルドによるすばやい原因調査が可能
  - **Insights Endpointによるビルド状況の可視化**

CircleCIの3000 万件のワークフローから得られたDevOpsに関する知見

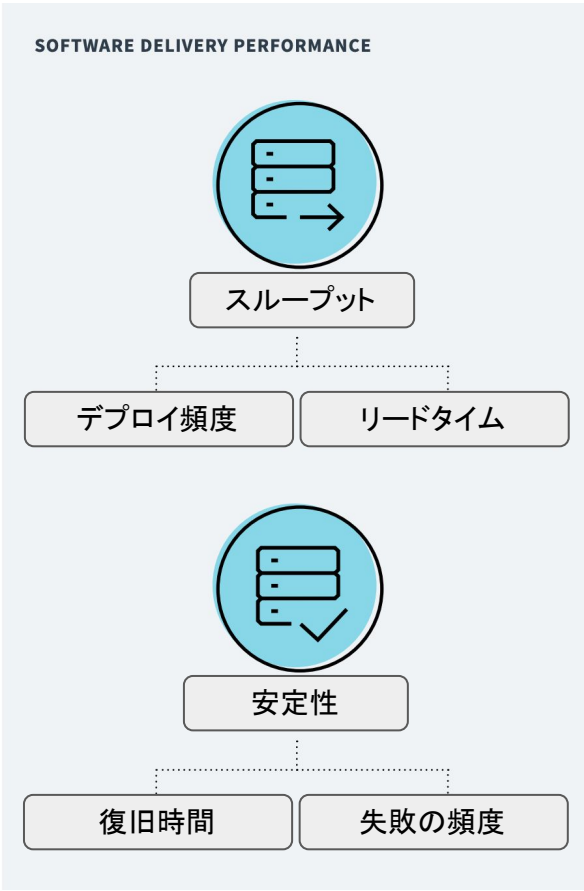
2020.03.18

Solutions Engineer @ CircleCI

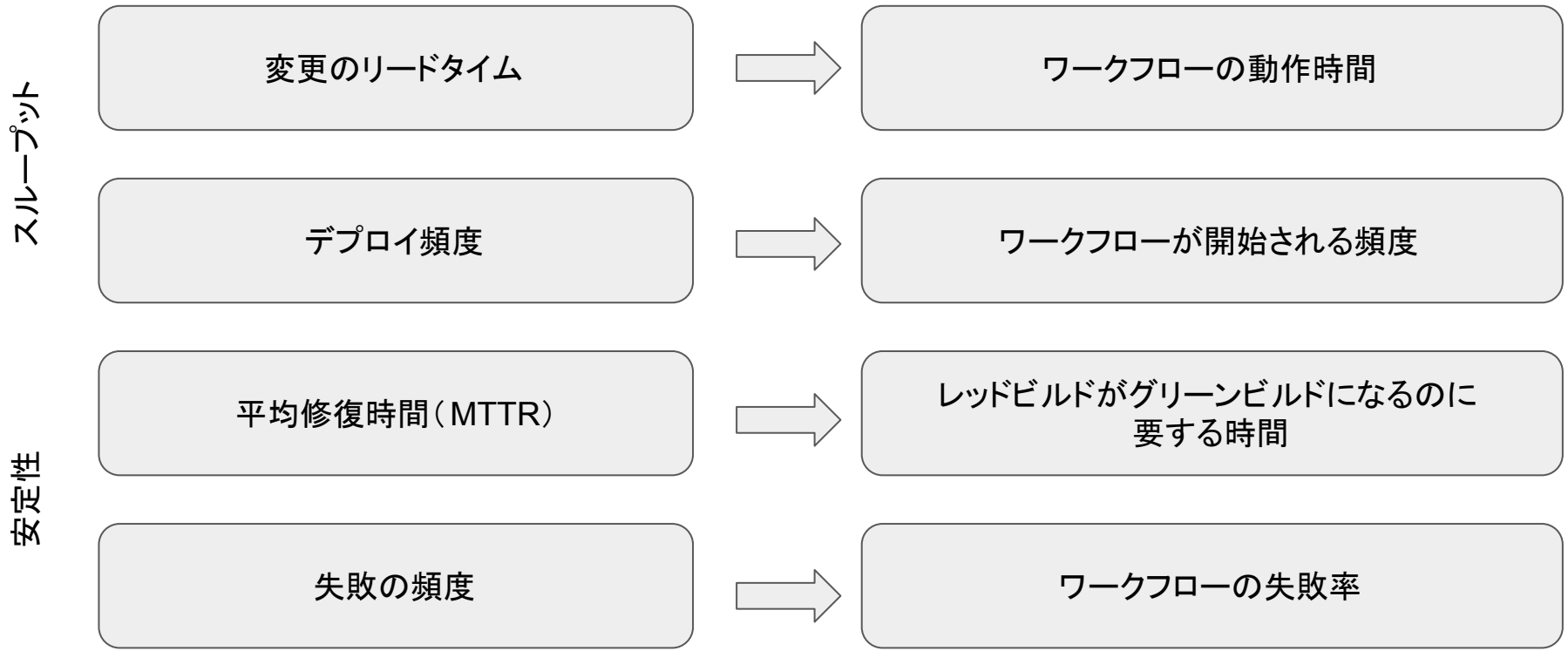
Noboru Kurumai

4つのメトリクス

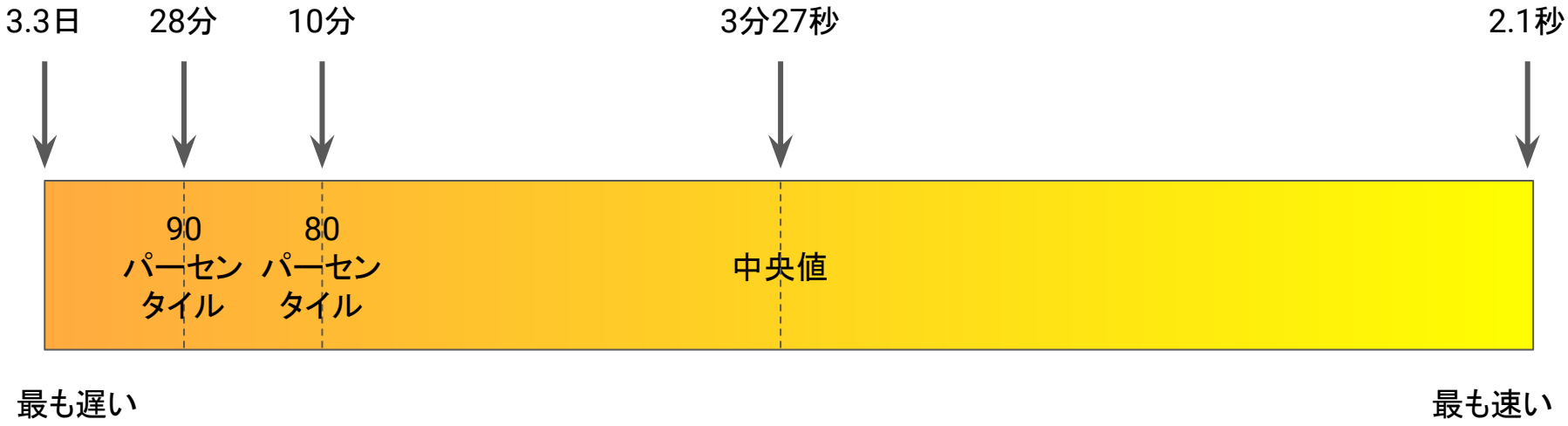

2018
Accelerate:
State of DevOps
Strategies for a New Economy
Presented by
DORA
DEVOPS RESEARCH & ASSESSMENT
Diamond Sponsor
Google Cloud
Gold Sponsors
Deloitte.
XebiaLabs
pagerduty
redgate
cloudbees
Datical
sumologic
TRICENTIS
GitLab
Electric Cloud
Microsoft Azure
Pivotal
aws
ca technologies


SOFTWARE DELIVERY PERFORMANCE
スループット
デプロイ頻度
リードタイム
安定性
復旧時間
失敗の頻度

4つのメトリクス


スループット
安定性
変更のリードタイム
ワークフローの動作時間
デプロイ頻度
ワークフローが開始される頻度
平均修復時間（MTTR）
レッドビルドがグリーンビルドになるのに
要する時間
失敗の頻度
ワークフローの失敗率

ワークフローの動作時間（分析結果）



デプロイの頻度（分析結果）

50パーセンタイル	6回 Organizationレベルで 1日に実行される数	3回 プロジェクトレベルで 1日に実行される数	1回 プロジェクトレベルで デフォルトブランチの ワークフローが実行される数
95パーセンタイル	250回 Organizationレベルで 1日に実行される数	74回 プロジェクトレベルで 1日に実行される数	39回 プロジェクトレベルで デフォルトブランチの ワークフローが実行される数

ここまでのまとめ

ソフトウェアデリバリーパフォーマンスの要素	エリート *	平均的な CircleCI ユーザー
デプロイ頻度 —> ワークフローの開始頻度	オンデマンド：1 日に複数回デプロイ	**1 日に 6 件のワークフロー**
変更のリードタイム —> ワークフローの期間	1 日未満	**3.5 分**
平均復旧時間	1 時間未満	**1 時間未満**
変更による失敗率 —> ワークフローの失敗率	0 ～ 15%	**デフォルトブランチで 18%**

回数は良いチームの指標にはならない
いつでもデプロイできる状態を維持すること

最適なワークフローは数分から数十分
開発者のフィードバックのスピードを意識

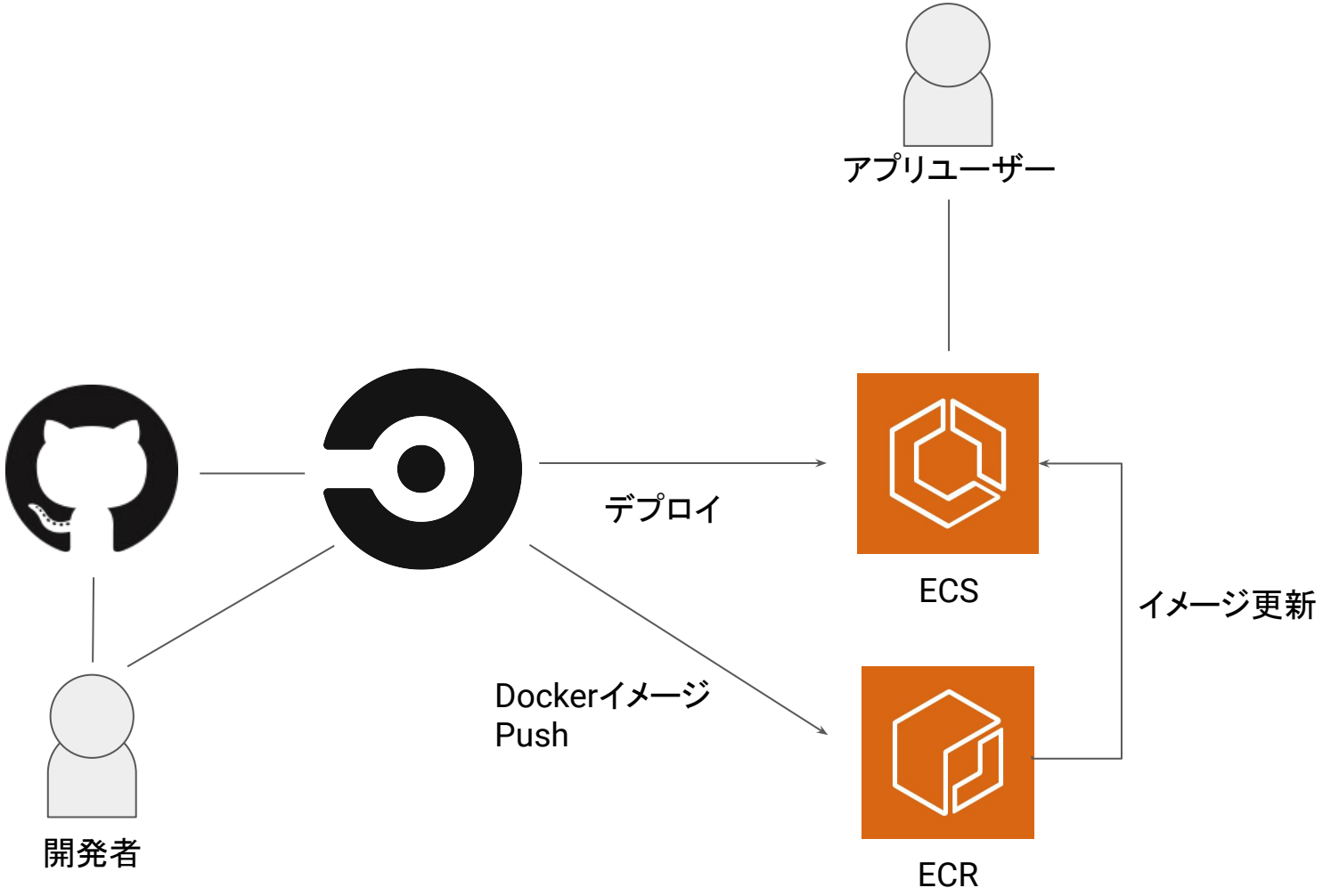
修復時間を短くするにはプロアクティブな対応と高度な自動化が必須

トピックブランチでの失敗は恐れない。むしろ積極的に結合する。
ベストプラクティスを取り入れる

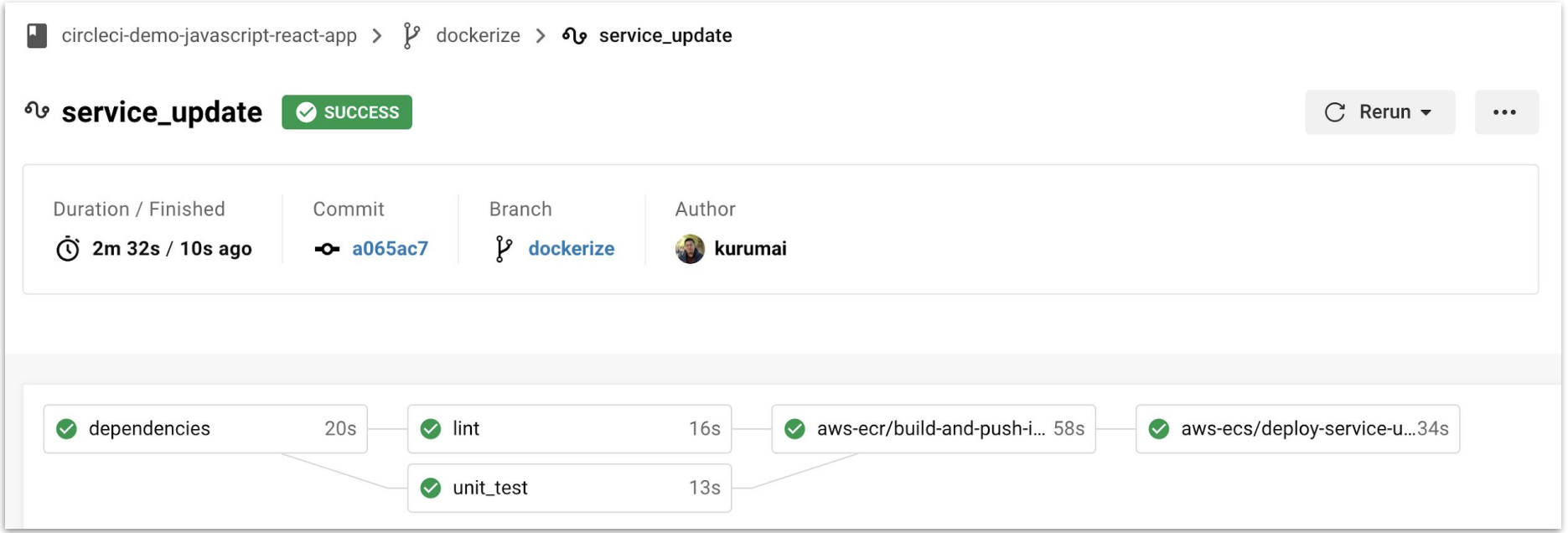
*『2019 Accelerate State of DevOps Report』を出典とするエリートメトリクス。

https://www2.circleci.com/Content-2020-05-14-JPTestinProduction_DeNALandingPage.html

デモ


アプリユーザー
デプロイ
ECS
イメージ更新
Dockerイメージ
Push
開発者
ECR

デモ


circleci-demo-javascript-react-app
dockerize
service_update
service_update
SUCCESS
Rerun
Duration / Finished
2m 32s / 10s ago
Commit
a065ac7
Branch
dockerize
Author
kurumai
dependencies 20s
lint 16s
aws-ecr/build-and-push-i... 58s
aws-ecs/deploy-service-u... 34s
unit_test 13s

Q&A

アンケートのお願い

ウェビナーからの退出やアプリケーションの終了でアンケート画面が現れます。アンケートにぜひご協力ください。

アプリケーションのシャットダウンでアンケートが表示されなかった場合、一時間後にアンケートメールが届きます。どちらかにてご回答をいただけますと幸いです。

Thank you.

無料プランで今日から試してみよう！

CircleCI

NewRelic