




DevOps::connect シーズン 3 「rethink CI/CD」

第5回 CI/CD なのだからちゃんとテストを書いてみよう
～ 分散環境のためのコンシューマ駆動契約を添えて～

自己紹介

名前：野村 侑志  @ugnomura

所属：技術統括本部 /
インダストリソリューション部

略歴：AWSソリューションアーキテクト
← 生保会社のSA・インターフェース開発リード
← 米国で日系製造業の現地販社のインターフェース開発リード
← 製造業で複合機のプログラム開発
← 音楽大学（打楽器）

興味：API, コンテナ, Haskell, 圏論, 競プロ

好きなAWSサービス：Amazon SQS



内容についての注意点

- 本セッションでは 2022 年 4 月 1 日現在のサービス内容および価格についてご説明しています。
- 最新の情報はAWS公式ウェブサイト (<http://aws.amazon.com>) にてご確認ください。
- 資料作成には十分注意しておりますが、資料内の価格とAWS公式ウェブサイト記載の価格に相違があった場合、AWS 公式ウェブサイトの価格を優先とさせていただきます。
- 価格は税抜表記となっております。日本居住者のお客様には別途消費税をご請求させていただきます。
- AWS does not offer binding price quotes. AWS pricing is publicly available and is subject to change in accordance with the AWS Customer Agreement available at <http://aws.amazon.com/agreement/>. Any pricing information included in this document is provided only as an estimate of usage charges for AWS services based on certain information that you have provided. Monthly charges will be based on your actual use of AWS services, and may vary from the estimates provided.

はじめに **DevAx::connect** とは

DevAx::connect

開発者のための開発者による Web セミナーシリーズ



<これまでのシリーズ> <https://bit.ly/3Lht3u3> にて公開中

Season1: イベント駆動

Season2: Challenge to NoSQL

番外編: CDK 実践勉強会

番外編: AWS App Runner 1周年記念

DevAx::connect シーズン 3 「rethink CI/CD」

開発者のための開発者による Web セミナーシリーズ



<開催時期>

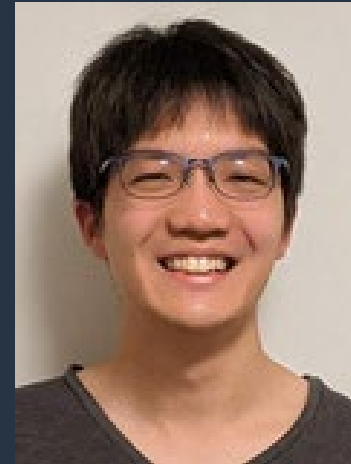
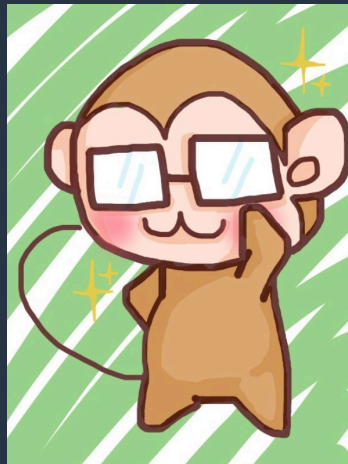
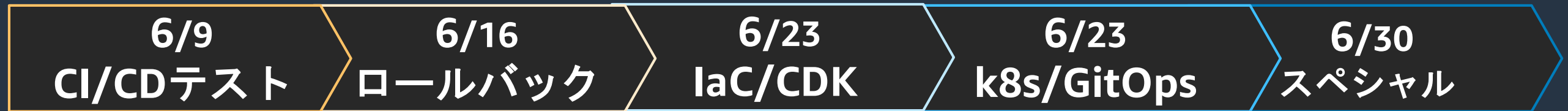
2022/4/7 ~ 4/28、6/9 ~ 6/30 毎週木曜日 16:00 ~ 18:00 に開催予定

<後半セッションタイトル> → 主に「CD (継続的デリバリー/デプロイ)」中心

5. CI/CD なのだからちゃんとテストを書いてみよう ~分散環境のためのコンシューマ駆動契約を添えて~
6. ECS で考える安全なデプロイメント
- 7a. インフラからアプリまで — すべてをコードで一元管理。安全なCI/CDも実現するには
- 7b. Kubernetes で実現する GitOps
8. スペシャルセッション

開発者のための開発者による Web セミナーシリーズ

AWS DevAx:connect 3rd 「rethink CI/CD」 (後編)



毎週木曜 16:00-18:00

令和も早や 4 年。私たちは「CI/CD」をできているのか

DevAx::connect シーズン 3 「rethink CI/CD」

開発者のための開発者による Web セミナーシリーズ



<開催時期>

2022/4/7 ~ 4/28、6/9 ~ 6/30 毎週木曜日 16:00 ~ 18:00 に開催予定

<前半セッションタイトル> → 主に「CI (継続的インテグレーション)」が中心テーマ

1. 雰囲気でもダン開発手法の実践をしている人のための CI/CD 再入門
- 2a. 適材適所で選ぶ CI/CD パイプライン構築サービス&ツール
- 2b. 複雑化する DevOps を可視化しよう ~ CI / CD を整えてからの次の一歩
3. トランクベース開発で信頼できる CI を実践する
4. その \よーいドン! / デプロイが、あなたのサービスの信頼性を損ねている

FAQ

Q. 毎回の参加は必須ですか?

A. 各回は独立して受講しても理解できるように構成しています。
また、毎回初回に前回の振り返りを行います。

Q. 後から動画を見返すことはできますか?

A. 終了後、動画/資料は公開される予定です。

Q. ブログなどで紹介してもよいですか?

A. 是非!! 公開したら教えてください!!

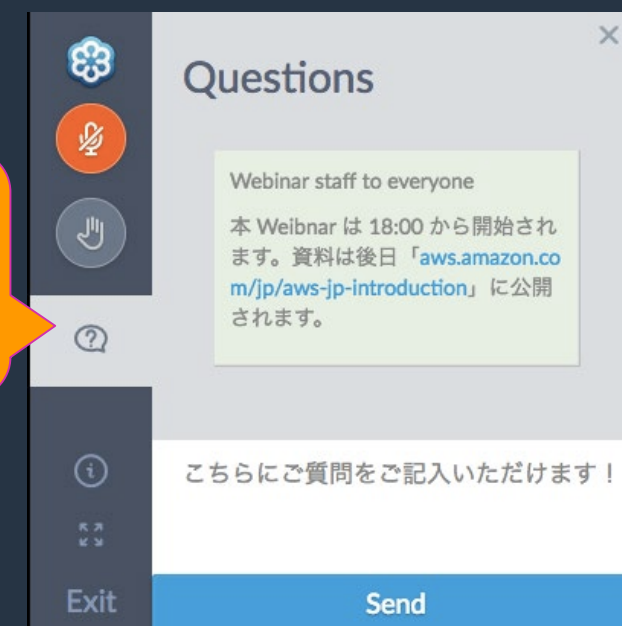
参加者の皆様へ

- ハッシュタグあります!! 

#devaxconnect にて、感想/質問など自由に投稿ください

- 適宜 QA にお答えします
 - Go To Webinar の QA 機能
 - ハッシュタグ  #devaxconnect にて、ご質問を投稿ください
登壇者がピックアップして回答します

- ① 吹き出しをクリック
- ② 質問を入力
- ③ Sendをクリック



- 休憩はありません

アンケートご協力をお願い

セッション終了後のアンケートにご協力ください!!

- 登壇者へのフィードバック
- 次回以降の DevAx::connect について
 - 開催方法や時間帯など
 - 取り上げて欲しいテーマ

DevAx::connect は 皆さんと共に作っていきたいと考えています!!

参考コンテンツ/イベントのご紹介

サーバーレス活用と効果の実際

日時: 2022年6月17日 10:00 ~

<https://pages.awscloud.com/JAPAN-event-OE-jp-EIB22-WWSO-CMP-Serverless-20220617-reg-event.html>

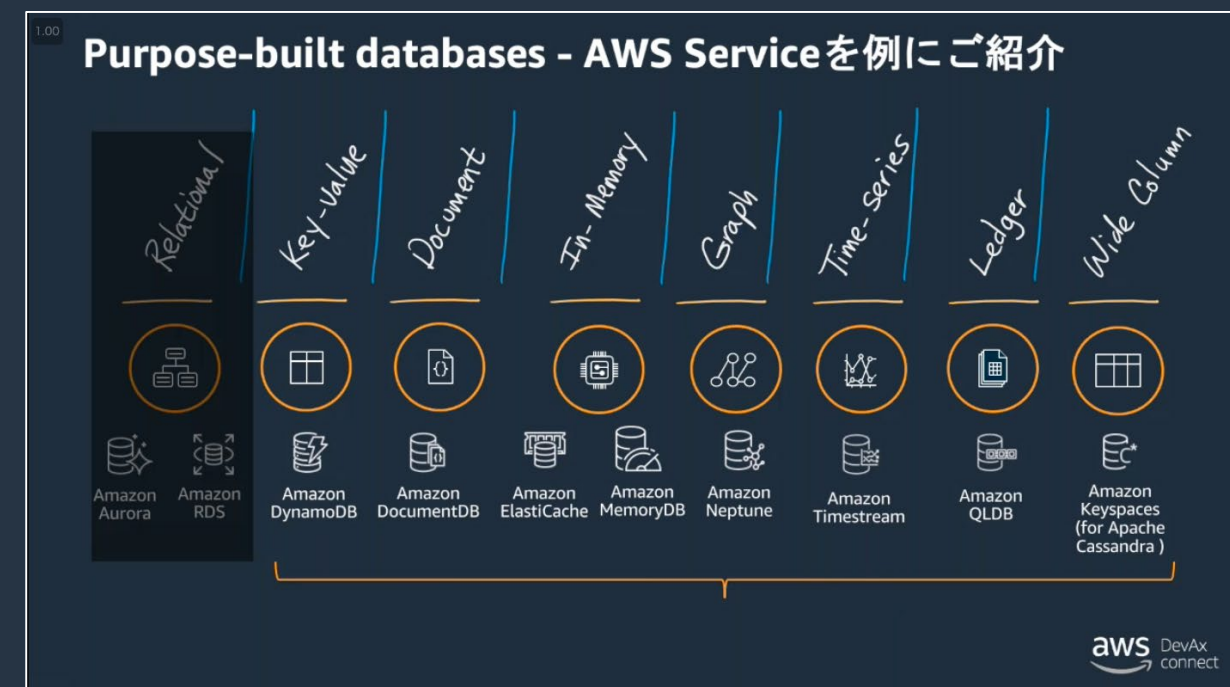
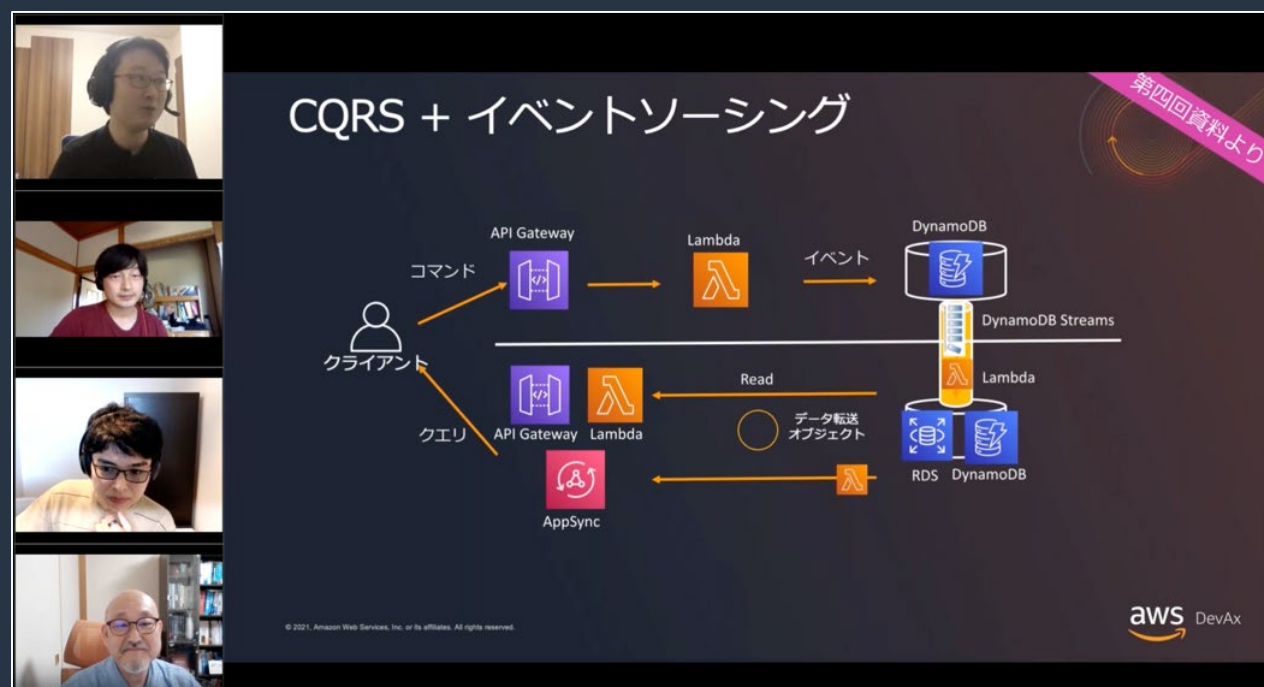
10:00 - 10:10	オープニング	
10:10 - 10:40	Amazon API Gateway 導入による オリックスグループ向けAPI の提供	オリックス様
10:45 - 11:15	DevOps チーム構築のための AWS サーバーレス移行への挑戦	横浜ゴム様
11:20 - 11:50	小さなチームで素早く開発！サーバーレスを 活用した記事データ同期システムの開発	朝日新聞社様

初めてのサーバーレスプロジェクトをどう進めたのか、サーバーレス/マネージドの活用で内製化へ向かうには？ 実際、どれだけ効果やメリットがあったのか、その実情を生の声でご確認ください。

AWS DevAx::connect

開発者のための開発者による Web セミナーシリーズ

シーズン1「イベント駆動」、シーズン2「Challenge to NoSQL」
をTwitchにて配信中!!



<https://www.twitch.tv/devaxconnect/videos>

builders.flash

デベロッパーの皆様へ向けて AWS を通じて様々な体験や学びをご提供するウェブマガジン

様々な AWS サービスの選択肢を発見していただき、クラウドの活用方法の理解を深めていただくことができます。



- サーバーレス、コンテナ、IoT、機械学習をはじめとしたクラウド最新技術動向の解説記事
- 日常に身近なテーマで AWS 活用方法を習得できるハンズオン
- グラフィックレコーディングを使った AWS サービス解説
- 開発における困難、悩みの解決のヒントとなる開発者インタビュー
- 初心者デベロッパー向けお悩み解決記事
- Enterprise Developer 向けソリューション解説記事

無料メールメンバー登録で、メンバー限定特典もご提供

<https://aws.amazon.com/jp/builders-flash/>



AWS Hands-on for Beginners

- 動画にそって実際に **手を動かしながら AWS の使い方を学ぶ** 無償コンテンツ
- **オンデマンド** 形式で公開しており、いつでもどこでもご視聴いただける
- **短い動画 x 5-10本** という形式でスキマ時間でも学ぶことが可能
- 現在、**22** のコンテンツを公開中

「AWS ハンズオン」で検索 or
クラウド活用資料集 > ハンズオン資料 

AWS ハンズオン資料

AWS をステップバイステップでお試しいただくのに役立つ動画および資料を掲載しています。

その他の資料は以下をご覧ください。

[初心者向けの資料](#) >

[サービス別の資料](#) >

[AWS オンラインセミナースケジュール](#) >

[AWS クラウドサービス活用資料集トップ](#) >

AWS 初心者向けハンズオン

AWS 初心者向けに「AWS Hands-on for Beginners」と題し、初めて AWS を利用する方や、初めて対象のサービスに触る方向けに、操作手順の解説動画を見ながら自分のペースで進められるハンズオンをテーマごとにご用意しています。

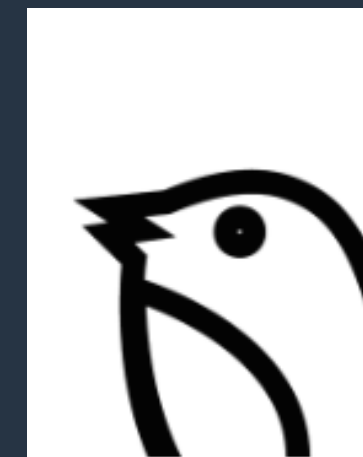
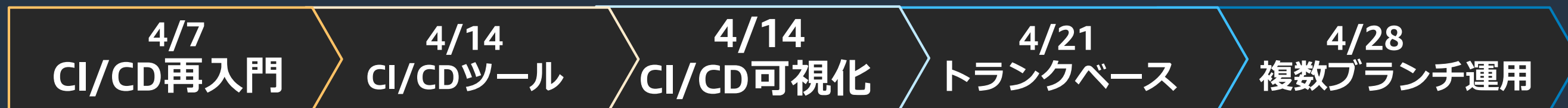
<p>ハンズオンははじめの一步: AWS アカウントの作り方 & IAM 基本のキ ></p>  <p>AWS アカウントの作成と IAM に関する説明/ハンズオンを行います。IAM ユーザーや IAM ポリシーなどの基本がわかり設定できるようになります。</p>	<p>Security #1 アカウント作成後すぐやるセキュリティ対策 ></p>  <p>AWS アカウント取得後まず最初にやっておきたい最低限の機能の有効化方法と、それらの確認方法について学べます。</p>	<p>Network編#1 AWS上にセキュアなプライベートネットワーク空間を作成する ></p>  <p>Amazon VPC の基本および VPC 内の通信と VPC から VPC 外に接続する方法について学べます。</p>
<p>Network編#2 Amazon VPC間およびAmazon VPCとオンプレミスのプライベートネットワーク接続 ></p> 	<p>Network編#3 クライアントVPNをつかって、リモート接続環境を構築しよう ></p>  <p>ハンズオンのアーキテクチャ</p>	<p>スケーラブルウェブサイト構築編 ></p>  <p>AWS Hands-on for Beginners スケーラブルウェブサイト構築編</p>

前半の振り返り

DevAx::connect シーズン 3 「rethink CI/CD」

開発者のための開発者による Web セミナーシリーズ

 #devaxconnect



令和も早や 4 年。私たちは「CI/CD」をできているのか

第1回 雰囲気でもダン開発手法の実践をしている人のためのCI/CD再入門

野村 侑志

アマゾンウェブサービスジャパン合同会社
ソリューションアーキテクト



CI/CD は組織で使っているけれど、なぜ CI/CD を使わないといけないかや具体的にどんないいことがあるのかが曖昧になっていませんか？本セッションでは、歴史的な背景やどのような意図で CI/CD を行うべきなのかという根本的な部分を見直すことによって、自分の開発環境の全体像を理解したり見直したりするきっかけにさせていただくことを目的にしています。

CI/CDとは

継続的インテグレーション
Continuous Integration (CI)

+

継続的デリバリー/デプロイ
Continuous Delivery/Deployment (CD)

リリースプロセスのステージ



CI : 継続的インテグレーション

CD : 継続的デリバリー

別のCD : 継続的デプロイ

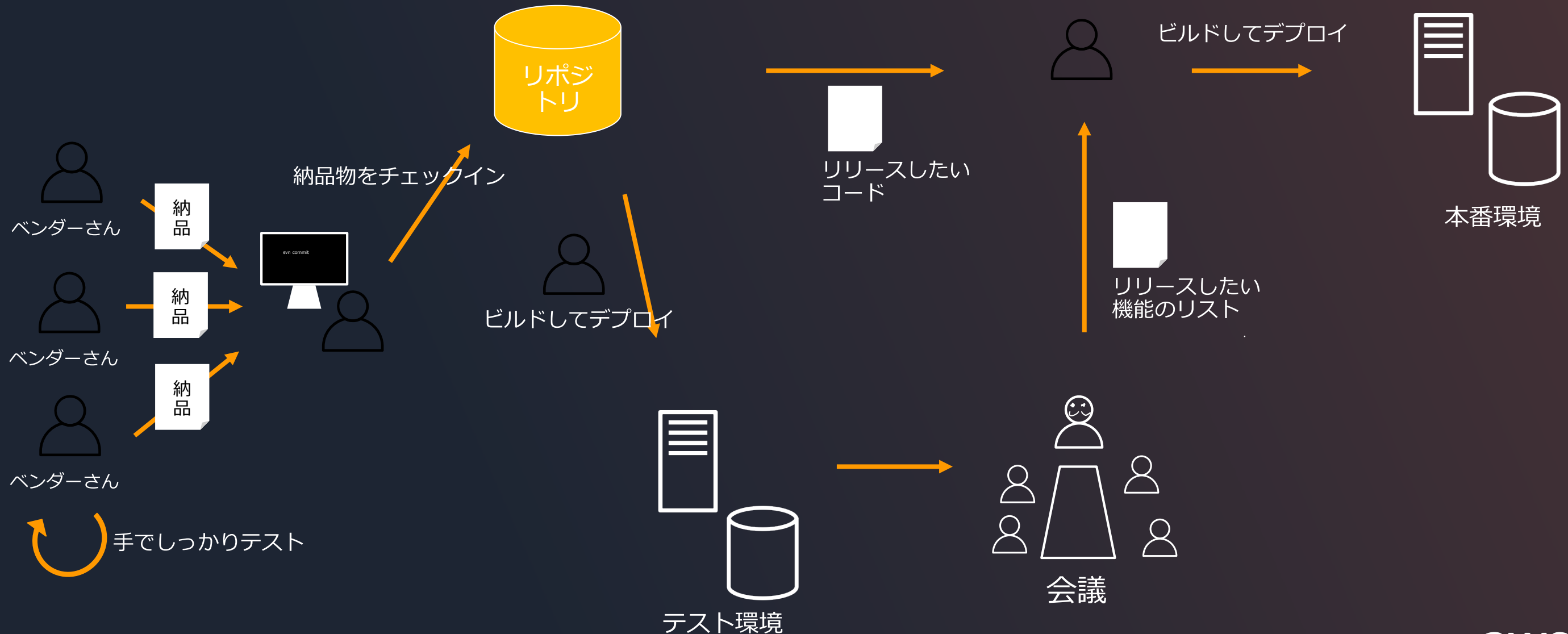


CI/CDの目的

誰かが良いアイデアを思いついたとき、できるだけ早くユーザーに届けるためのソフトウェア開発手法

- ソフトウェアのビルド・デプロイ・テスト・リリースという**プロセスのあらゆる部分が関係者全員から見える**ようにし、共同作業をやりやすくすること
- **フィードバックを改善**し、プロセスにおいてできる限り早い時期に問題が特定されて解決されるようにすること
- ソフトウェアの任意のバージョンを任意の環境に対して、**完全に自動化されたプロセスを通じて好きなようにデプロイ**できるようにすること

例：とある企業の開発アプローチ



CI/CDの導入

- **CI準備編**

1. コードのバージョン管理できてますか？
2. ユニットテスト書いてますか？
3. ビルドは自動化出来ていますか？

- **CI接続編**

4. ビルドとテストを自動化する環境はありますか？
5. Webhookは使っていますか？
6. チェックインのルールを決めていますか？

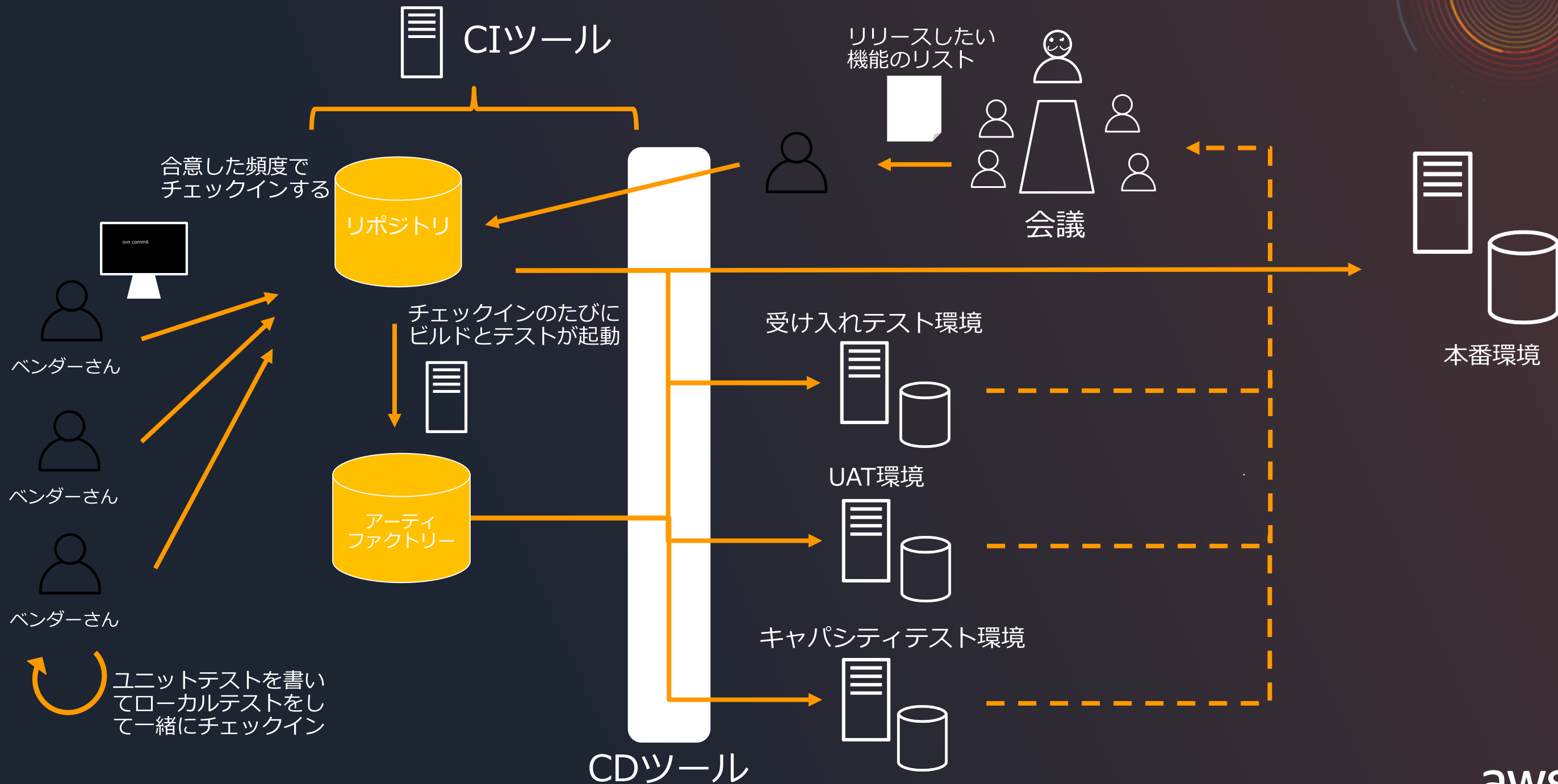
- **CDステージ構築編**

7. CIできてますか？
8. テストを人に頼っていませんか？
9. デプロイは自動化してますか？

- **CDパイプライン構築編**

10. 本番環境のデプロイは安全に素早く出来ていますか？
11. CDツールは使っていますか？

CD : パイプライン構築編





#devaxconnect

第2回 適材適所で選ぶ CI/CD パイプライン 構築サービス&ツール

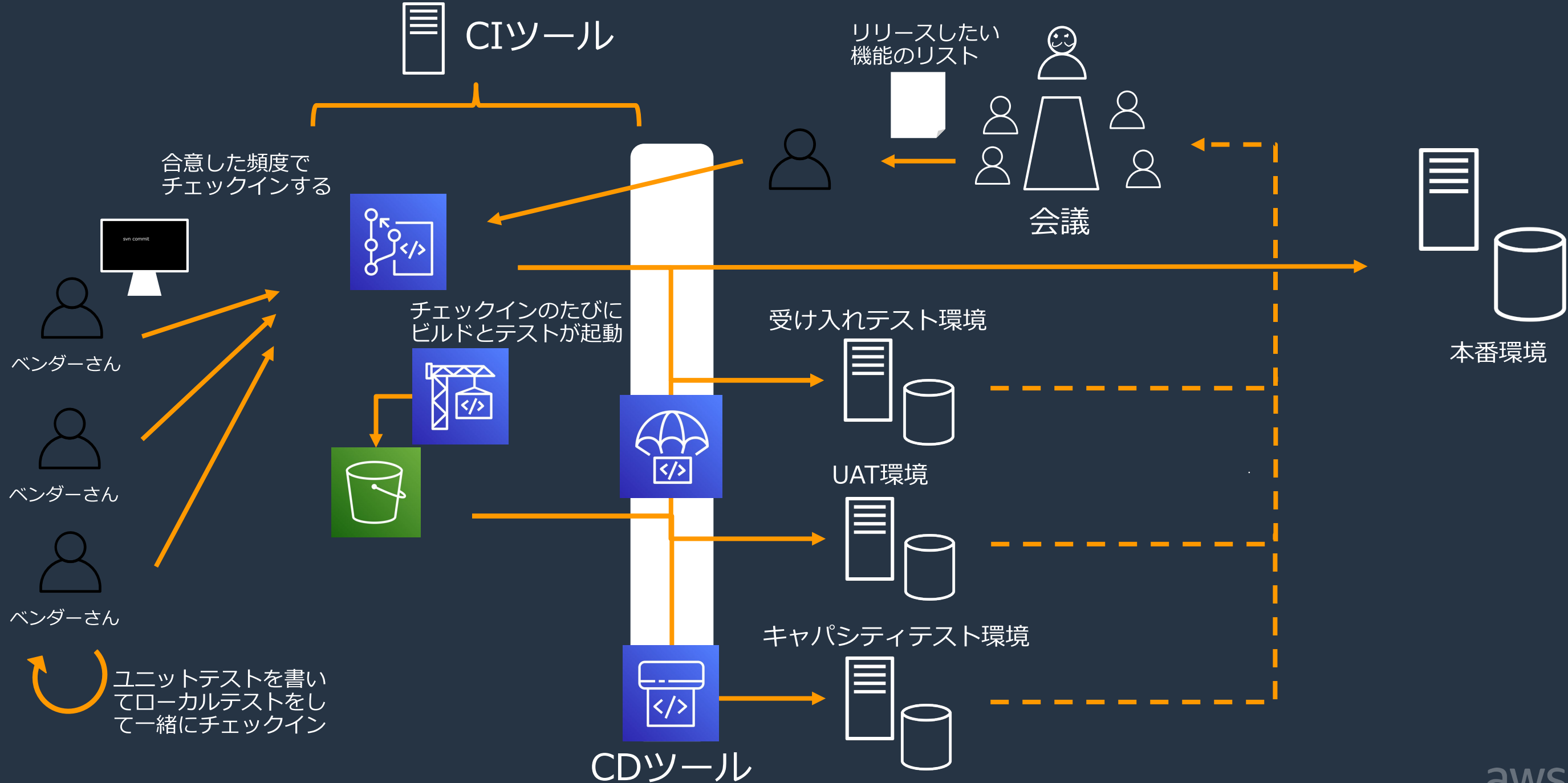
松本 雅博

アマゾンウェブサービスジャパン合同会社
シニアソリューションアーキテクト



AWS の CI/CD パイプラインサービスというと、まずは CodePipeline を思い浮かべられる方が多いのではないのでしょうか？ 現在、AWS には CDK Pipeline, AWS Copilot, AWS SAM Pipeline などのパイプライン構築に活用できるツールや、AWS Amplify Console, AWS Proton など CI/CD パイプラインが組み込まれたサービスがあります。また、3rd Party のサービス、ツールをご利用いただくこともできます。このセッションでは CI/CD に関する多彩なサービス、ツールをどのような場面で活用するのかについてお話しします。

CD : パイプライン構築編



デプロイの自動化

- 素早く、安全にデプロイできる
- ロールバック機能



AWS CodeDeploy



AWS CDK



AWS CloudFormation



AWS SAM



Terraform

パイプラインの構築

- ソースコードのチェックインをトリガーにした一連の処理を実行
- パイプラインの可視化



AWS CodePipeline



Jenkins



Bitbucket



CircleCI



GitHub



GitLab

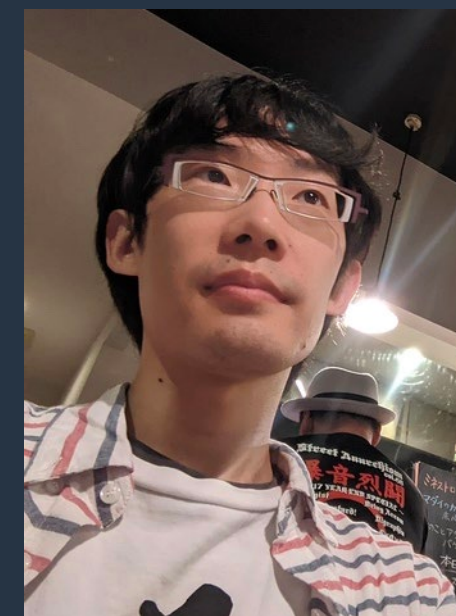


#devaxconnect

第2回 複雑化する DevOps を可視化しよう ～ CI/CD を整えてからの次の一歩～

光吉 隆雄

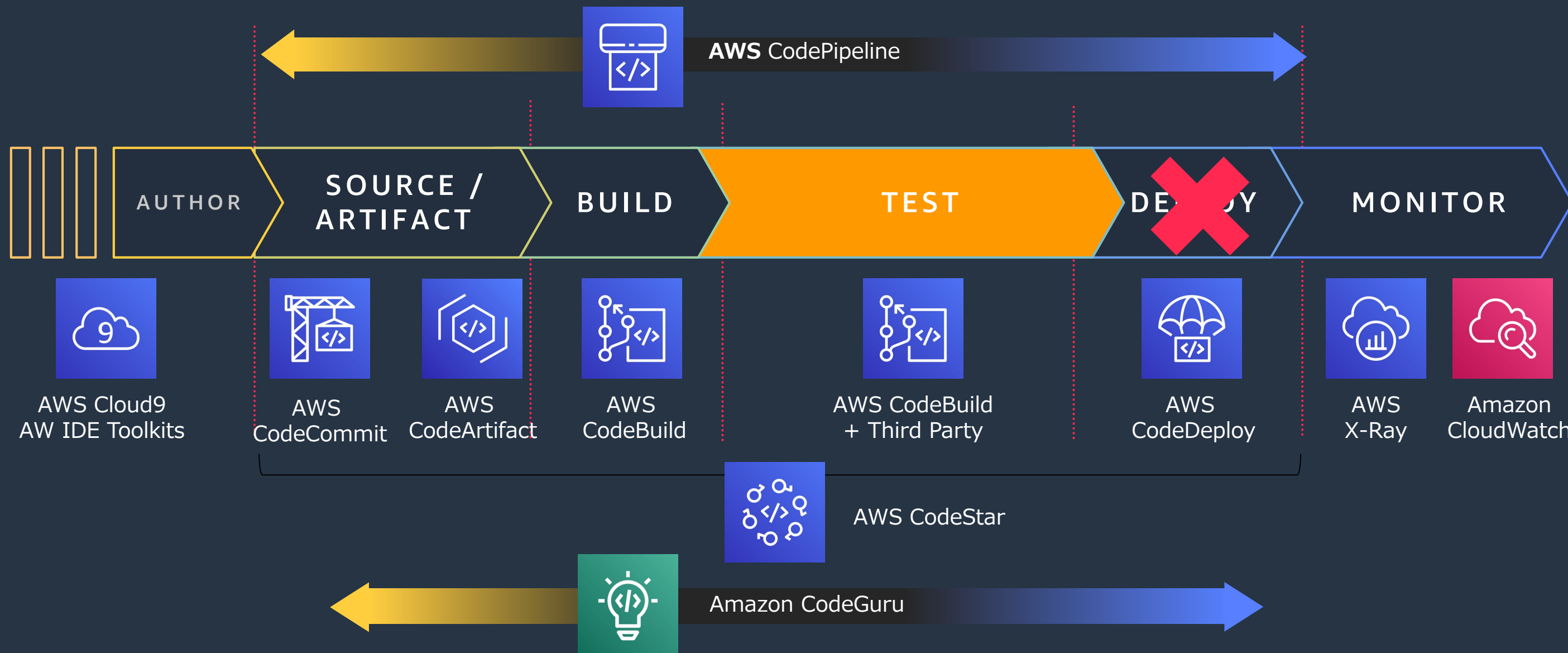
アマゾンウェブサービスジャパン合同会社
ソリューションアーキテクト



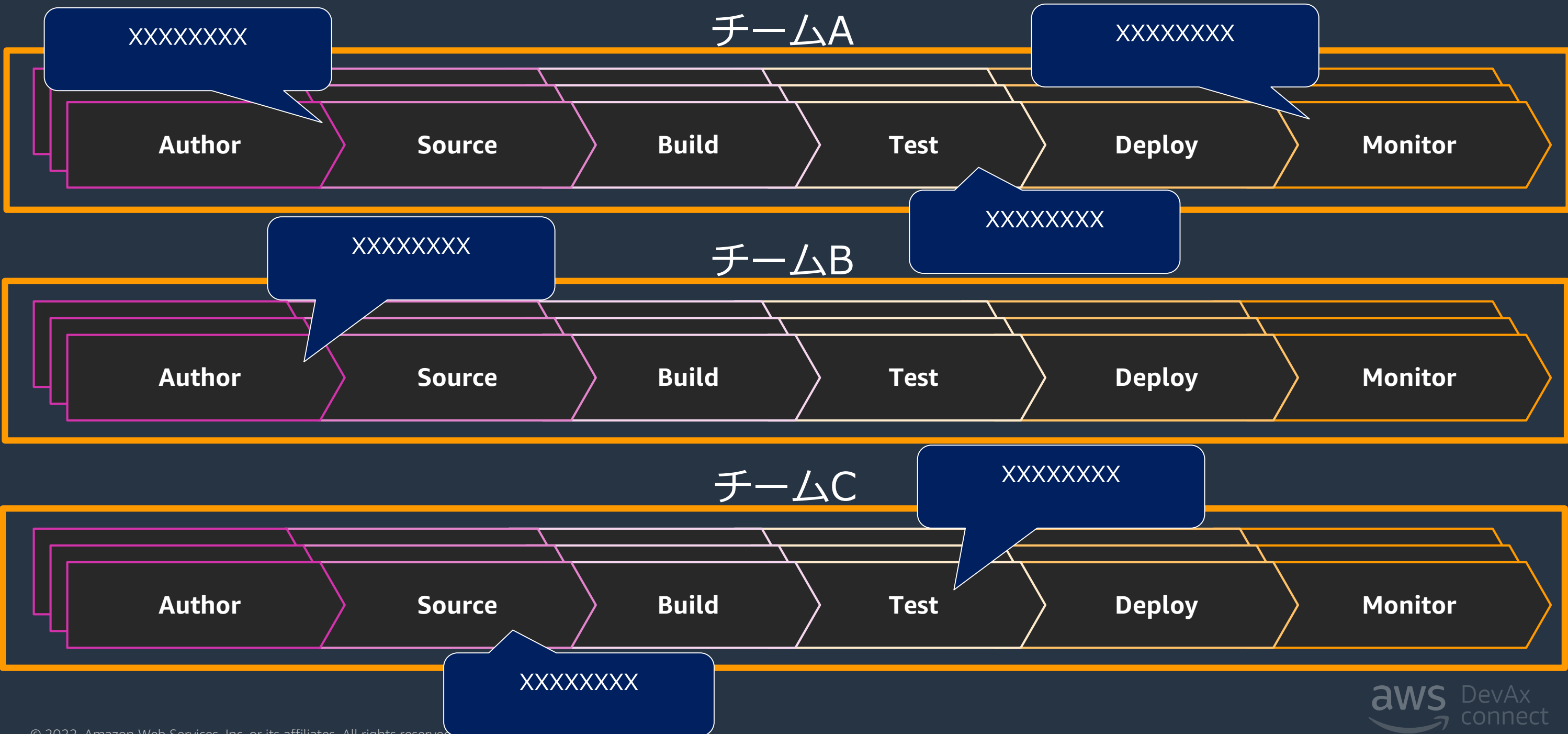
コード管理、ビルド、テスト、デプロイそしてそれらを繋ぐパイプライン。現代の開発スタイルはこれらの CI / CD ツール群によってシームレスに連携されています。これらのツール群は数も多く、AWSが提供しているものもあれば様々なサードパーティー製ツールやSaaSなどが存在しますが、そうになると各ツールのモニタリングはそれぞれのツールで完結してしまいます。しかしチームやプロジェクトを推進する人々の観点だと、それぞれのツールに留まったメトリクスのみを見たいわけではなく、開発・運用自体を可視化して把握したいのではないのでしょうか。本セッションはこういった課題を炙り出し、そして解決するための手法を考察します。そして解決手法のひとつである AWS ソリューションの AWS DevOps Monitoring Dashboard を紹介し実際に動かすまでを実践します。



CI/CD を導入と劣化



マイクロサービスによるパイプラインの増大



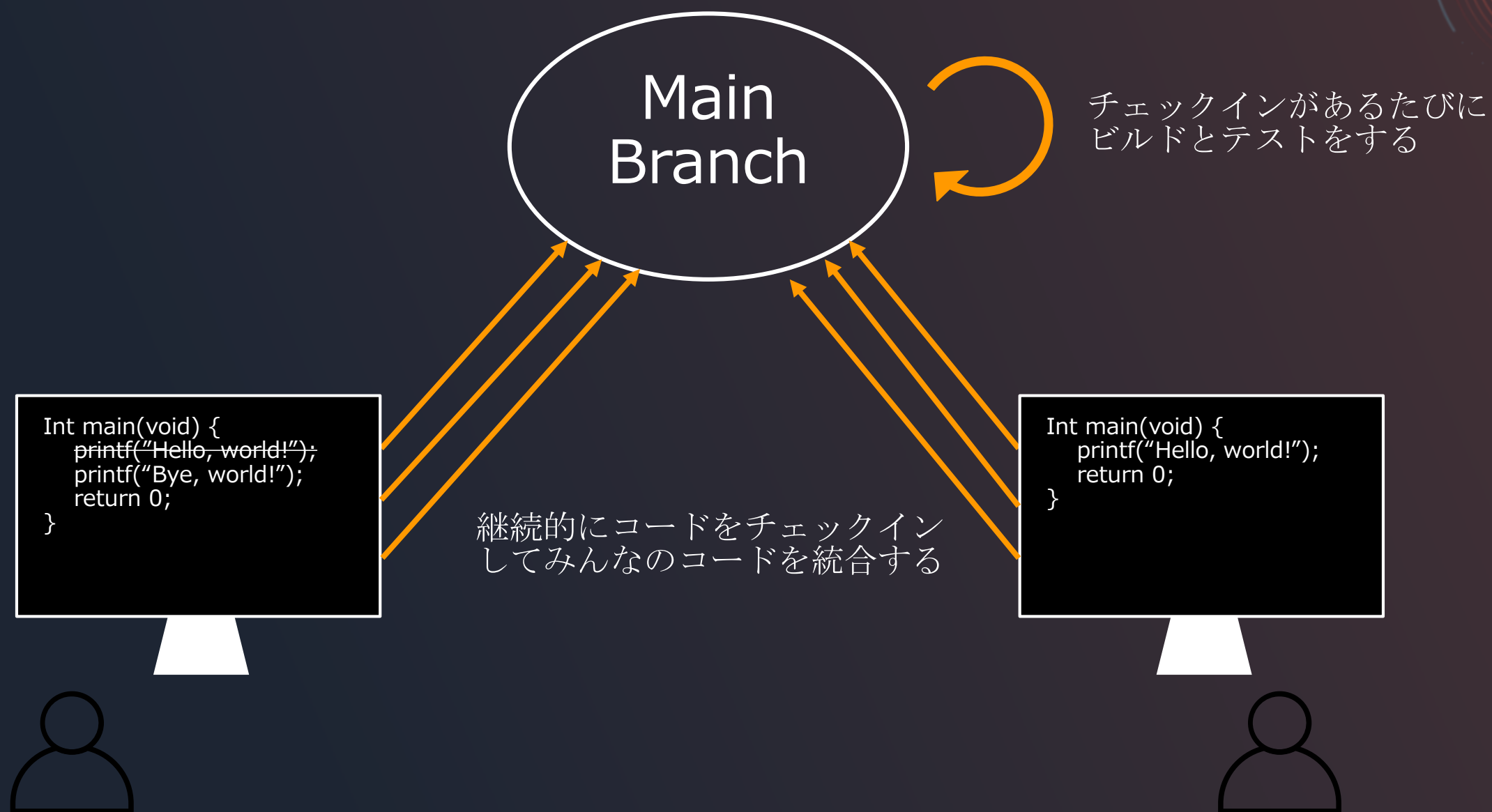
指標と関係する測定対象のまとめ

	機能の提供時間	デプロイ回数	エラー発生率	平均修復時間
その他	Issueステータス	Issueステータス		Issueステータス
Source	誰がPushしたか リポジトリ活性度	ホットフィックス	ホットフィックス	誰がPushしたか ホットフィックス
Build	成功率/失敗率 ビルドサイズ	実施頻度		成功率/失敗率 ビルドサイズ
Test	成功率/失敗率 カバレッジ	実施頻度		成功率/失敗率 カバレッジ
Deploy	成功率/失敗率 所要時間	実施頻度	成功率/失敗率	成功率/失敗率
Monitor	正常稼働状態か		正常稼働状態か	正常稼働状態か

継続的インテグレーション



#devaxconnect



コード管理戦略の種類



#devaxconnect

- トランクベース
- フィーチャーブランチ
- Git Flow
- GitHub Flow
- Pull Request





第3回トランクベース開発で信頼できるCIを実践する

林 政利

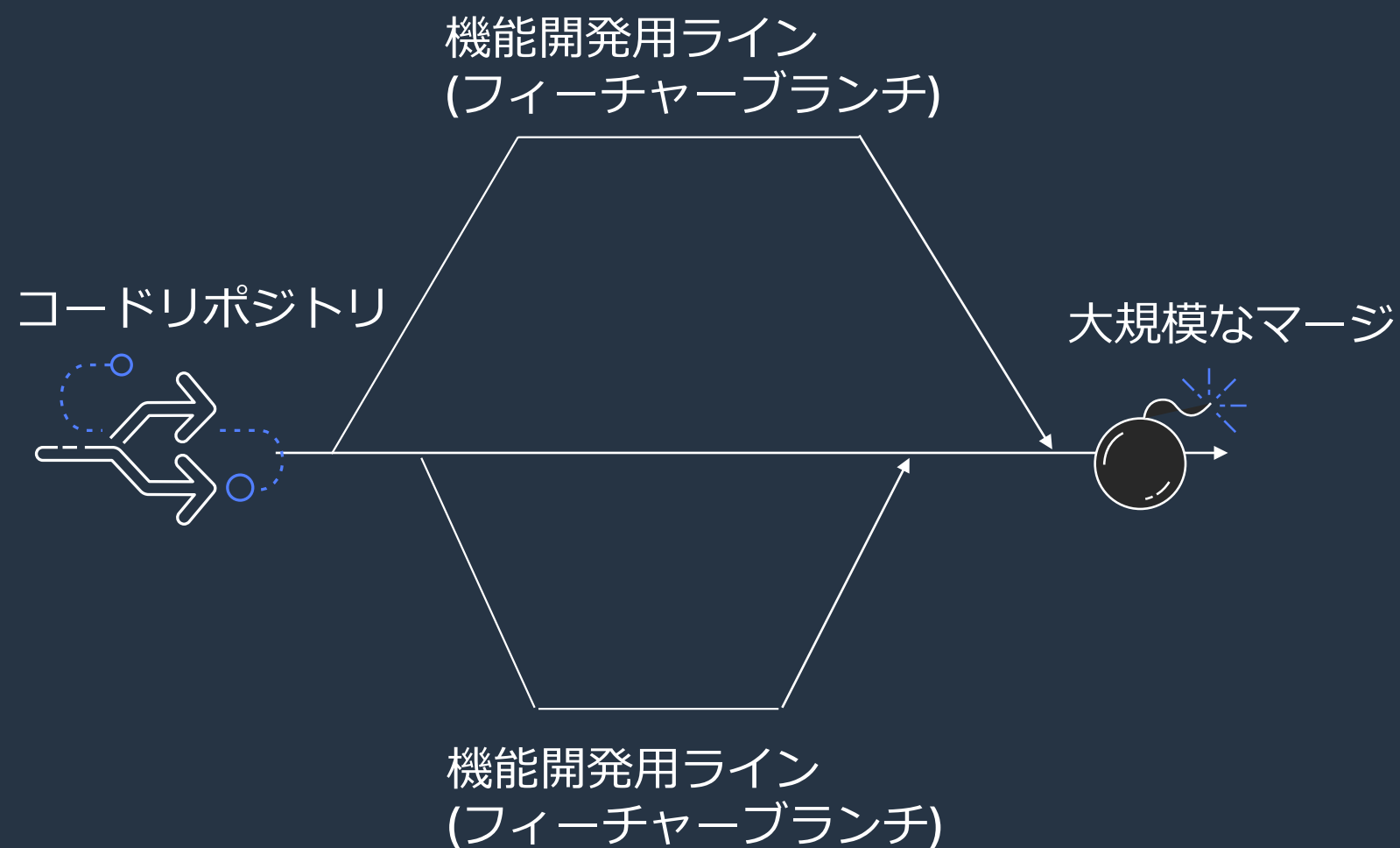
アマゾン ウェブ サービス ジャパン合同会社
ソリューション アーキテクト



CI、継続的インテグレーションのベースとなるトランクベース開発、実践できているでしょうか？ メインブランチに継続的にコードをインテグレーションするというこの技術、信頼できるCIのためには必要不可欠ですが、実践しようとするのとこの足を踏む、との声も多く聞かれます。このセッションでは、コンテナ技術とCI/CDを題材に、トランクベース開発を阻むブロッカーとその解消手段について掘り下げてみます。

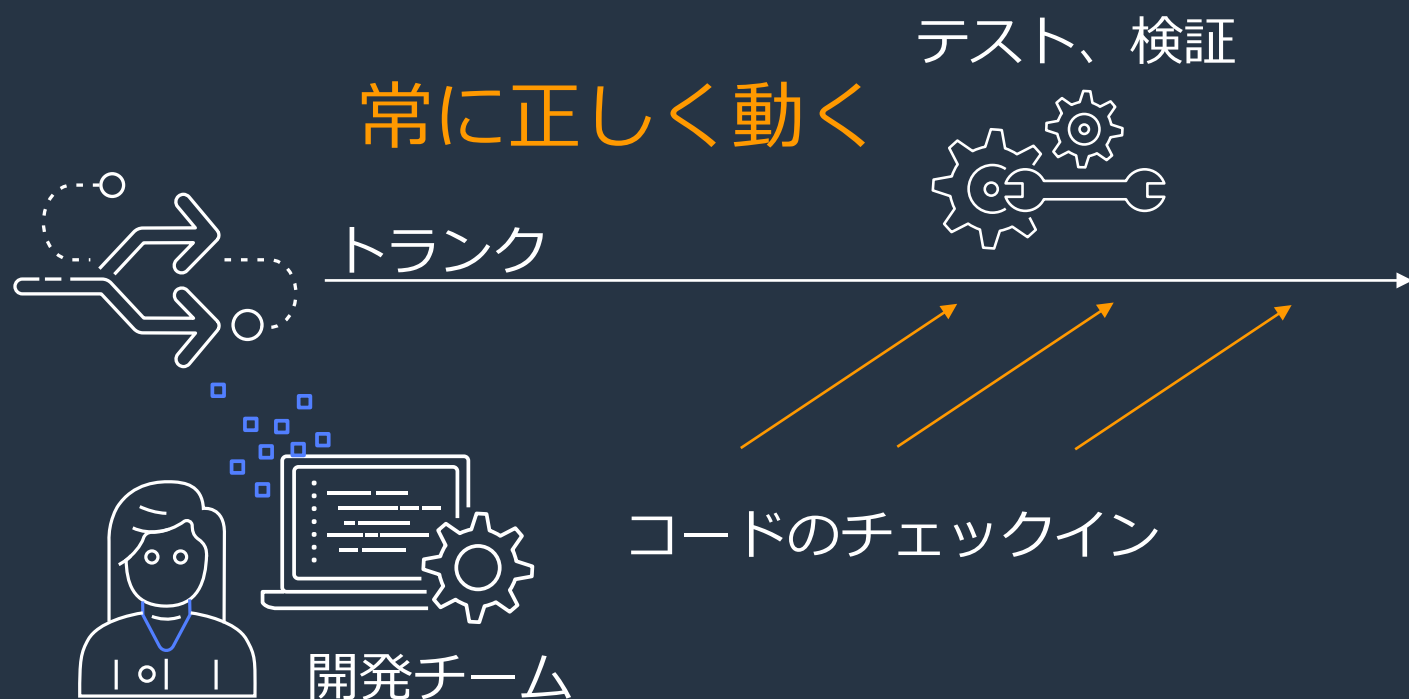
なぜ、トランクベース開発が必要なのか

ソースコードの大規模なマージを避け、小さなマージに分割する



- リリースの安定性
- 継続的な改善 (リファクタリング)

トランクベース開発に必要なプラクティス



自動テスト

正しく動くが分からないと頻繁にトランクにチェックインし続けることはできない

コードレビュー

重厚長大なマージプロセスでは頻繁にトランクにチェックインし続けることはできない

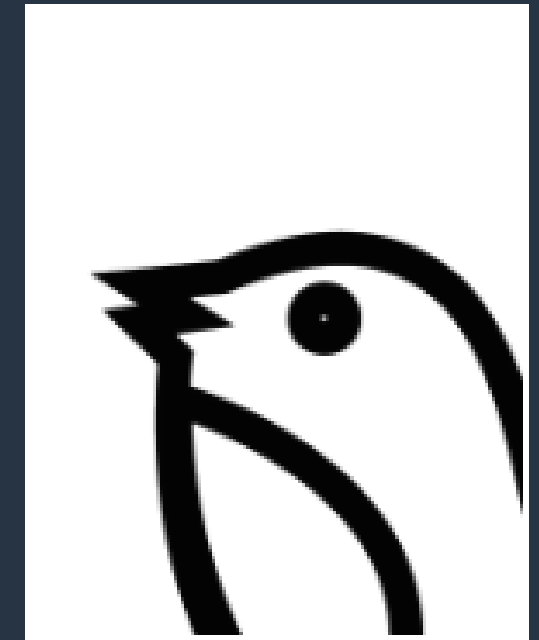
小さいチャンクでの機能開発

小さい変更でなければ、頻繁にトランクにチェックインし続けることはできない

まとめ

- 適切な CI/CD の導入を目指す
 - トランクベース開発が CI/CD のキモ、前提になる
- トランクベース開発を導入するためのアクションを検討する
 - リリースブランチ、デプロイとリリースの分離などでブロッカーに対処
- ブランチ戦略を決定する
 - 長期ブランチを切る前に、本当に必要なのかよく検討する
 - なるべく長期ブランチを減らす、トランクベース開発を目指す

第4回 スペシャルセッション 「その \よーいドン！ / デプロイが、 あなたのサービスの信頼性を損ねている」



Tori Hara (@toricls)
Software Engineer, Kaminashi

本セッションは、昨年(たぶん)大絶賛を受けた CI/CD Conference 2021 でのセッション「永続複数ブランチ運用は『単一のコードベース』と言えるのか」をベースに、カミナシ社のリアルワークロードの例も踏まえつつ 2022 年のビッグアップデート版としてお届けする予定です。



必要以上に難しくしない

バズワードに踊らされない
未来は不定
デプロイとロールアウトは違う

常に整理整頓

思考停止に陥らない



#devaxconnect

本日のテーマ

開発者のための開発者による Web セミナーシリーズ

AWS DevAx:connect 3rd 「rethink CI/CD」 (後編)



毎週木曜 16:00-18:00

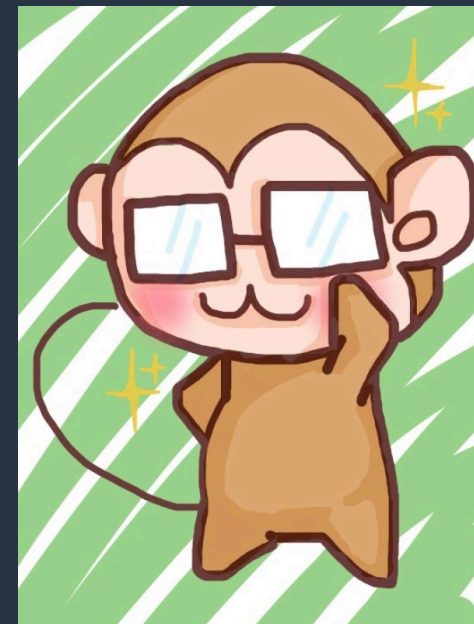
令和も早や 4 年。私たちは「CI/CD」をできているのか



第5回 CI/CD なのだからちゃんとテストを書いてみよう ～ 分散環境のためのコンシューマ駆動契約を添えて～

金森 政雄

アマゾンウェブサービスジャパン合同会社
ソリューションアーキテクト



個々のサービスが独立してデプロイ/リリースを行い、アジリティの高い開発を実現できることがマイクロサービスアーキテクチャが採用される理由の1つです。一方で、多数のサービスが協調して動作するマイクロサービスでは、各サービスを互換性を持って変更/運用していくことが難しくなります。コンシューマ駆動契約は、サービスの利用者(コンシューマ)と提供者の間で仕様を"契約"として定義し、契約に基づいてテストすることで、サービス間連携の整合性を担保するプラクティスです。このセッションでは、コンシューマ駆動契約の基本的な考え方と、それをCI/CDに組み込んだ際の具体的な動きをデモも交えて解説します。

本日 QA にお答えする SA のご紹介 #1

 #devaxconnect

Name

Kensuke Shimokawa

Company

Amazon Web Services Japan G.K.

Role

Serverless Specialist Solutions Architect



@_kensh

本日 QA にお答えする SA のご紹介 #2

 #devaxconnect

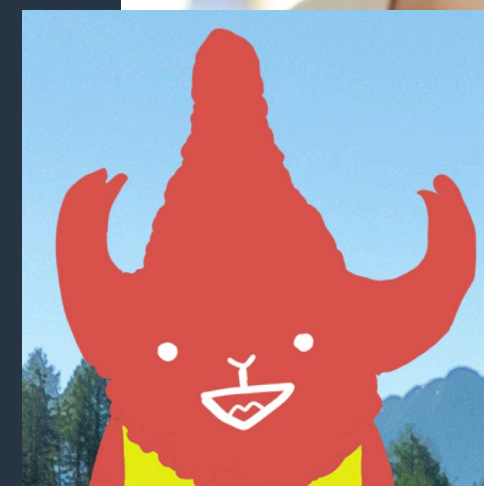
宇賀神 みずき

しょぼちむ @syobochim



主にパートナー様の育成・技術支援を担当

元 Sler のアプリケーション開発エンジニア
著書「いちばんやさしいGit&GitHubの教本」



本日 QA にお答えする SA のご紹介 #3

 #devaxconnect

杉本 晋吾 (すぎもと しんご)

アマゾンウェブサービスジャパン合同会社
技術統括本部 インダストリーソリューション部
ソリューションアーキテクト



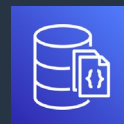
AWS をこれから使い始める方、今後もっと AWS を活用したい、
というお客様向けに、お客様個別の IT 課題解決や技術テーマを中心に、
お客様と共に様々な課題に取り組む「伴走者」として活動中です。

AWS SA ← (IT コンサル CTO ← ソフトハウス CEO ← アパレル会社 CIO ← EC ベンダー CTO) 兼プログラマー

好きなAWSサービス



AWS App Runner, AWS Fargate



Amazon DocumentDB



本日 QA にお答えする SA のご紹介 #4

福井 厚 (ふくい あつし) Twitter: afukui@

アマゾンウェブサービスジャパン合同会社

シニアソリューションアーキテクト

サーバーレス スペシャリスト



❖ 関心領域

❖ ソフトウェア アーキテクチャ、オブジェクト指向設計、アジャイル開発

❖ 好きなAWSサービス

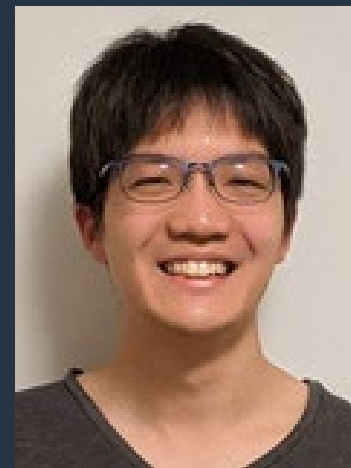
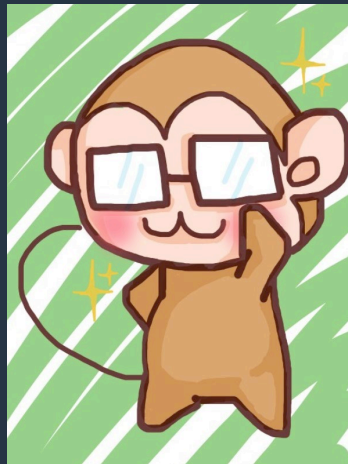
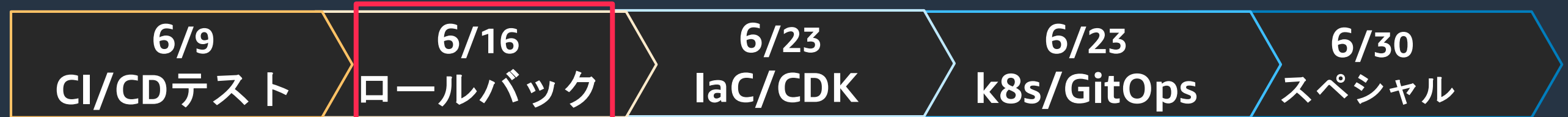
❖ サーバーレステクノロジー全般、AWS Code シリーズ、AWS Amplify

Q&A

お疲れ様でした!! DevAx::connect season 3 次回予告

開発者のための開発者による Web セミナーシリーズ

AWS DevAx:connect 3rd 「rethink CI/CD」 (後編)



毎週木曜 16:00-18:00

令和も早や 4 年。私たちは「CI/CD」をできているのか

第5回 ECS で考える安全なデプロイメント

落水 恭介

アマゾンウェブサービスジャパン合同会社
ソリューションアーキテクト



CI/CD パイプラインの設計において、ロールバックプロセスの検討は最も重要な観点の1つです。アラートをトリガーに運用担当者へメールを送る、いくつかのメトリクスをターゲットにロールバックを判断するアクションをパイプラインに組み込むなど、その方法はさまざまです。本セッションでは、ECS へのデプロイを例に、ロールバックプロセスをどのように設計するのか、CI/CD パイプラインへのロールバックプロセスの導入についてお話します。

アンケートご協力をお願い

セッション終了後のアンケートにご協力ください!!

- 登壇者へのフィードバック
- 次回以降の DevAx::connect について
 - 開催方法や時間帯など
 - 取り上げて欲しいテーマ

DevAx::connect は 皆さんと共に作っていきたいと考えています!!



#devaxconnect

Thank you!