



AWS コンテナサービス入門

Yasuhiro Tori Hara
Specialist Solutions Architect, Containers

Amazon Web Services Japan

Jul. 24, 2019

Yasuhiro Tori Hara

Specialist Solutions Architect, Containers
Amazon Web Services Japan



ERP パッケージベンダー R&D チーム SDE

➡ □ UI 自動テスト SaaS

➡ クラウド利用の SI + MSP にて、コンテナやサーバーレスによる設計・開発・運用
Web 技術利用のゲームやビジネスアプリケーション、ML/DL 環境構築運用など

➡ 現職

《 好きな AWS のサービス 》 AWS Fargate, AWS Lambda

本セッションは…

想定聴講者

- docker run はしたことがある
- コンテナの特性はなんとなく知っている
- コンテナワークロード構築を検討中
- AWS が提供するコンテナサービスそれぞれの特徴、どういう時に使うものかを知りたい

ゴール

- 各コンテナサービスが解決しようとしている課題を理解する
- 自身のアプリケーションをコンテナ化して AWS 上で動かす際に、どのコンテナサービスが必要か判断できるようになる

アジェンダ

コンテナとは

コンテナオーケストレーション

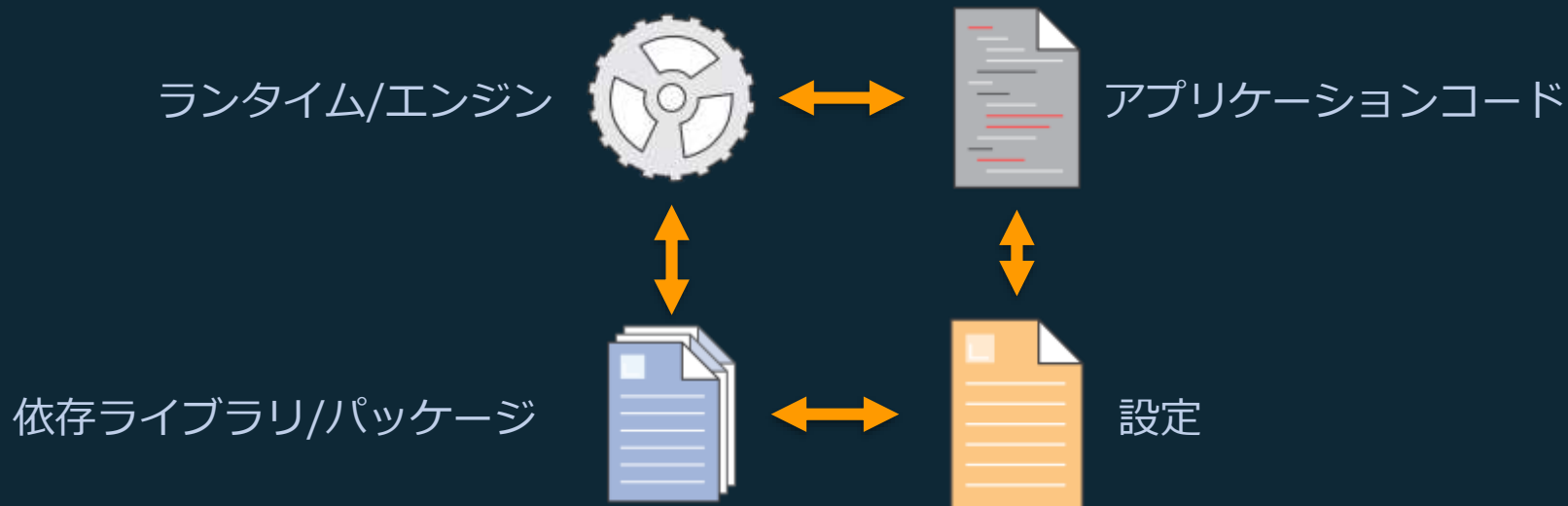
イメージレジストリ、コンテナ実行環境

その他のコンテナ関連サービス

現実世界のコンテナワークロード

コンテナとは

アプリケーションを構成するコンポーネント



ローカルでは動いたけど、本番で動かない？



ローカルラップトップ

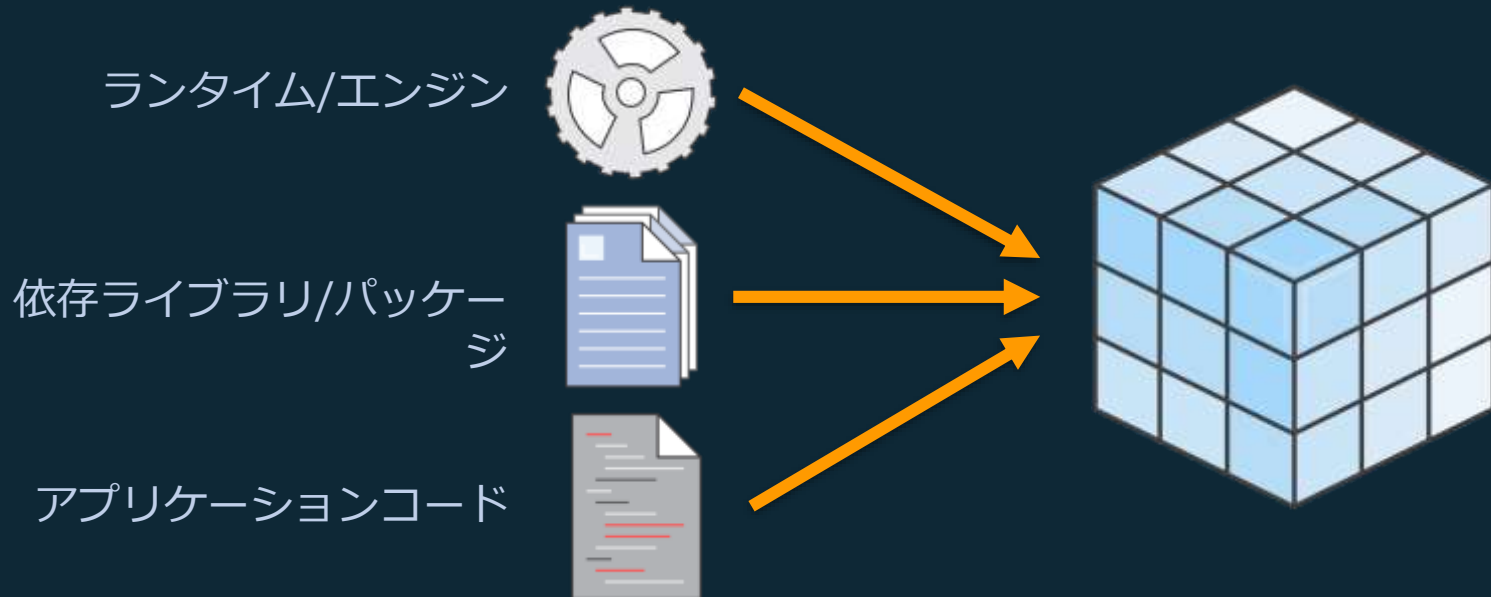


ステージング / QA



本番

「コンテナ」という解決策



Docker

Docker 社が開発

Apache 2.0 ライセンス

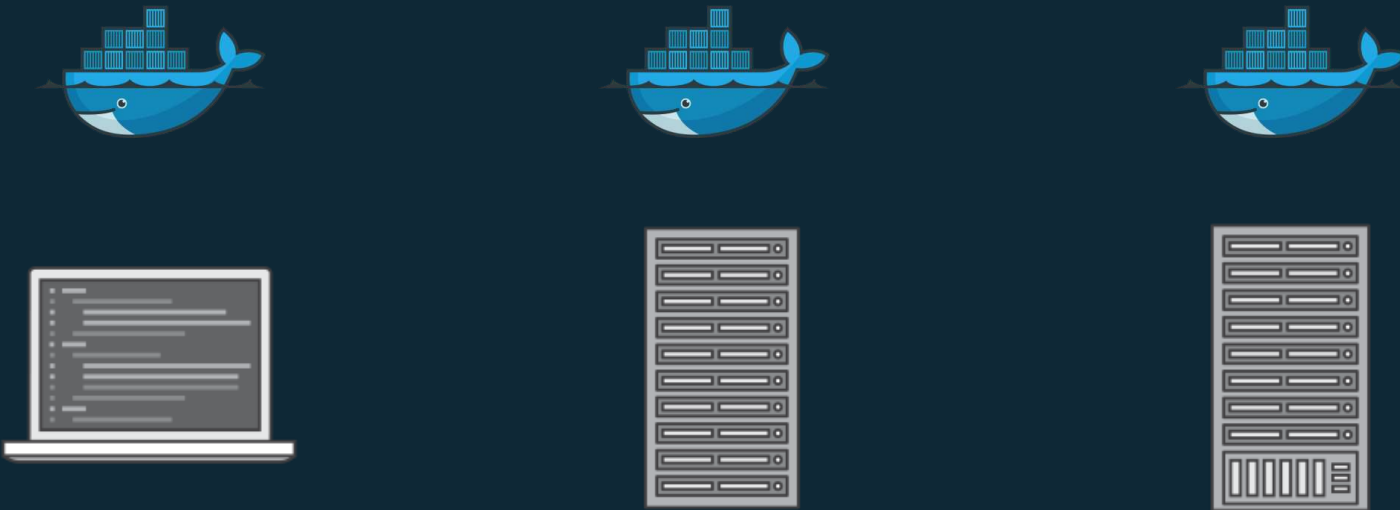
2013年3月 OSS リリース

常駐型コンテナ実行エンジン

コンテナのライフサイクル管理・デプロイツール



全ての環境で同じコンテナを動かす

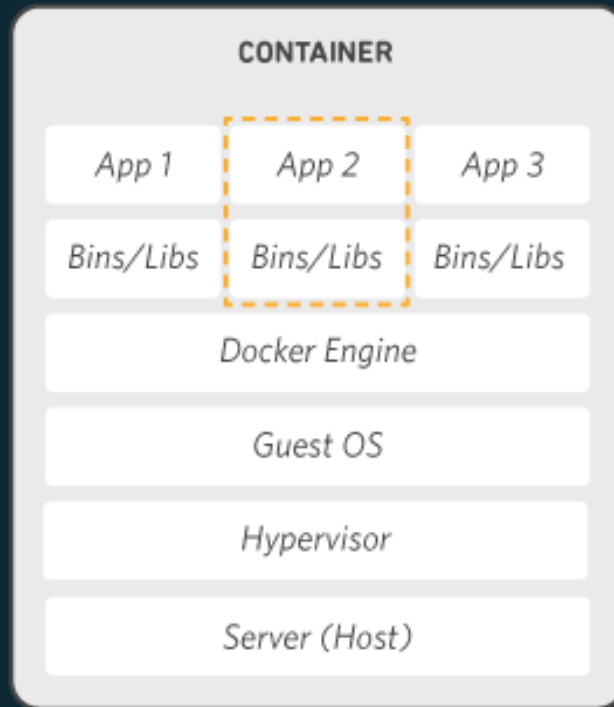
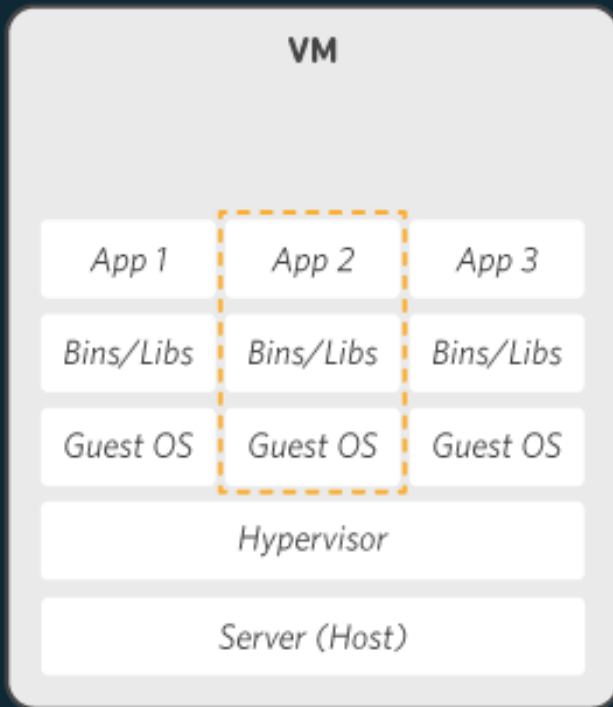


ローカルラップトップ

ステージング / QA

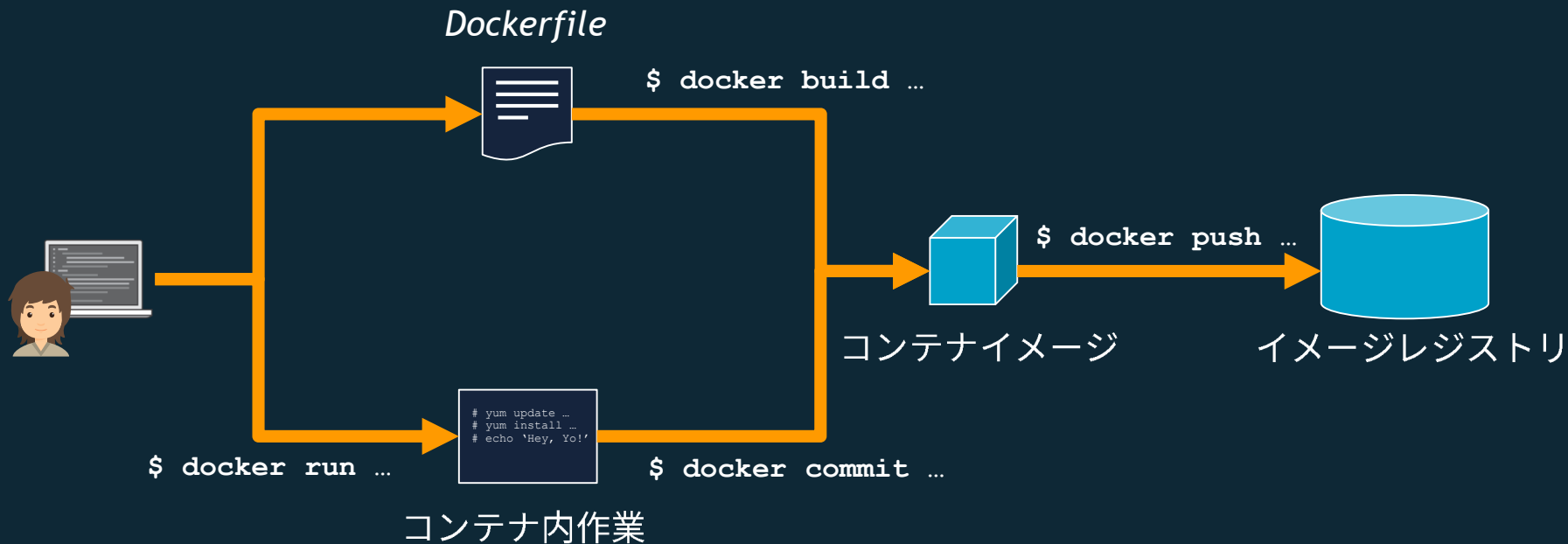
本番

仮想マシンとコンテナ



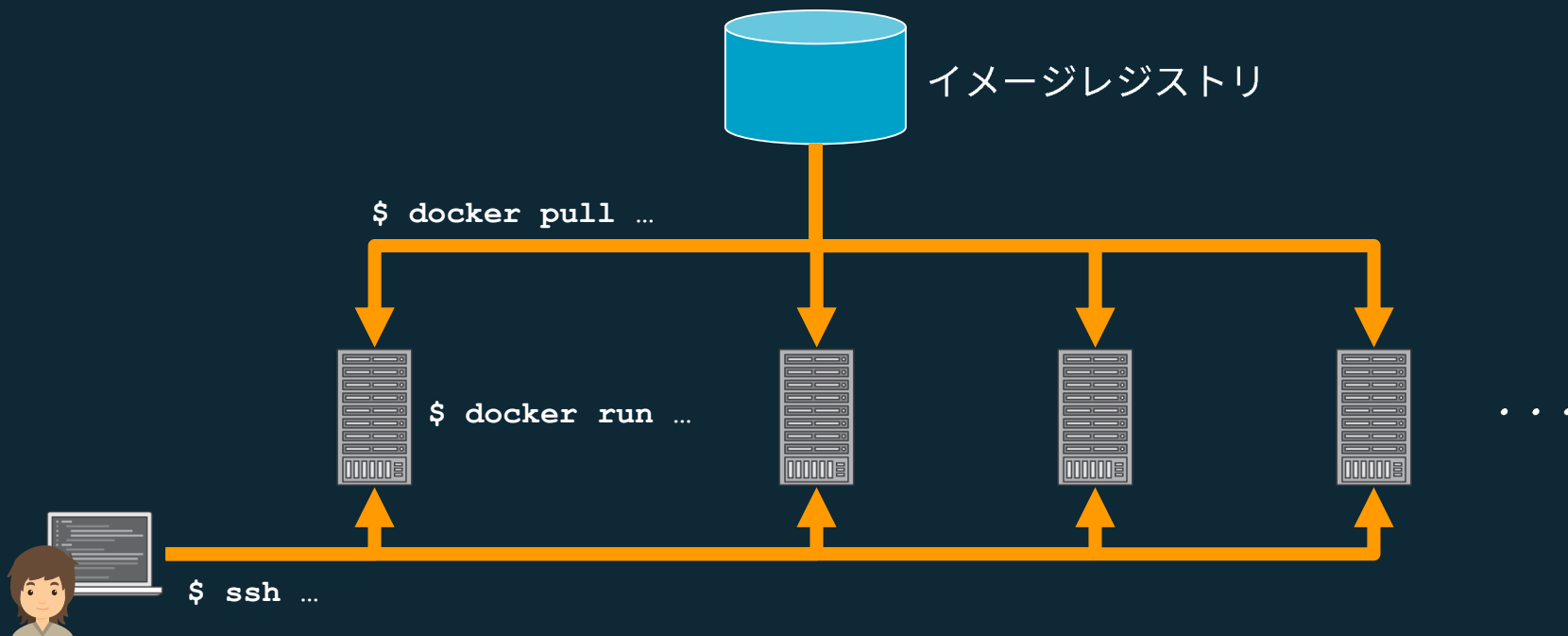
Docker を利用した基本的ワークフロー

- コンテナイメージ作成 -



Docker を利用した基本的ワークフロー

- コンテナ実行 -



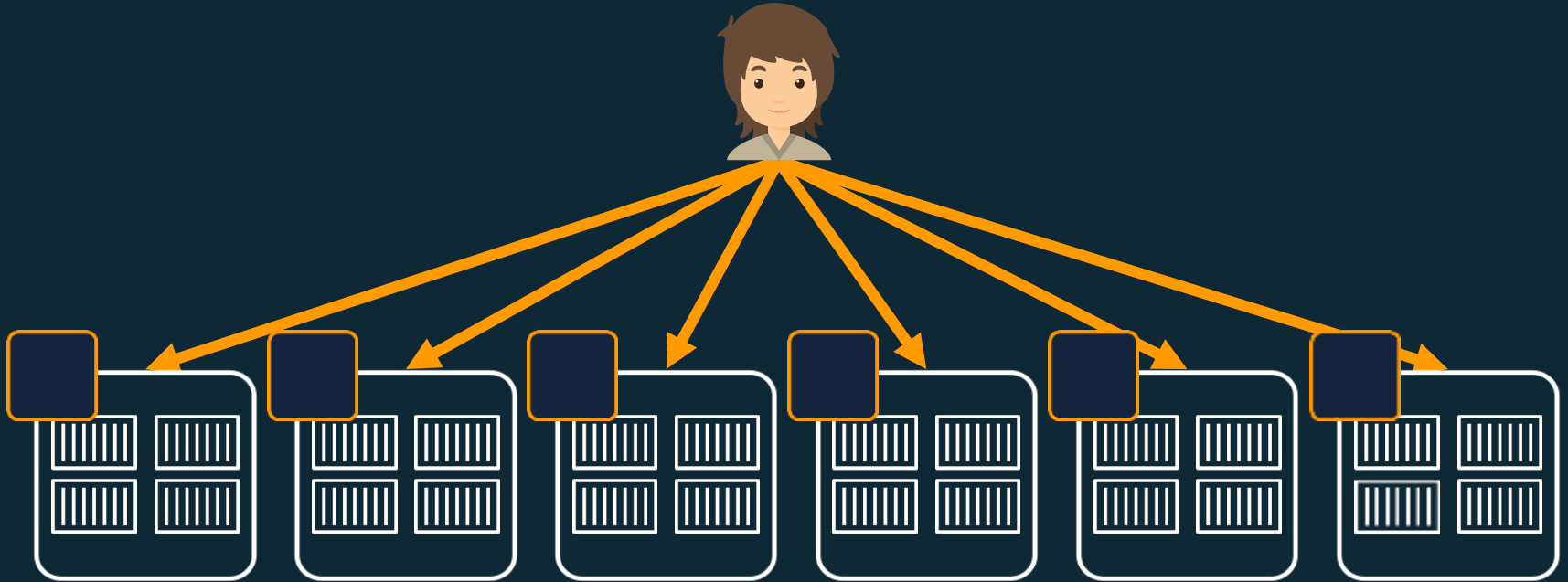
思っていたよりも手作業？



- Docker の責務は**同一サーバ上**のコンテナライフサイクル管理
- **複数サーバやコンテナを束ねた概念**に対するオペレーションはスコープ外

コンテナオーケストレーション

手作業でのコンテナイメージダウンロードと実行は 非効率かつミスオペレーションを招く



「このEC2インスタンスの
クラスターでコンテナを
実行したいです」



Amazon
ECS

「このコンテナを3つのAZに
分散させて10個デプロイして、
このロードバランサーに
つないでください」



Amazon Elastic Container Service (ECS)

- クラウドでコンテナを本番環境利用するためのオーケストレーター
- 各種 AWS サービスとの高度な連携
- 数億コンテナ/週、数百万 EC2 インスタンスを管理するスケーラビリティ
- 多様なワークロードをサポートする「タスク」「サービス」というシンプルなリソース表現
- Linux/Windows サポート



Amazon ECS customers - from Startups to Fortune 500 companies





kubernetes



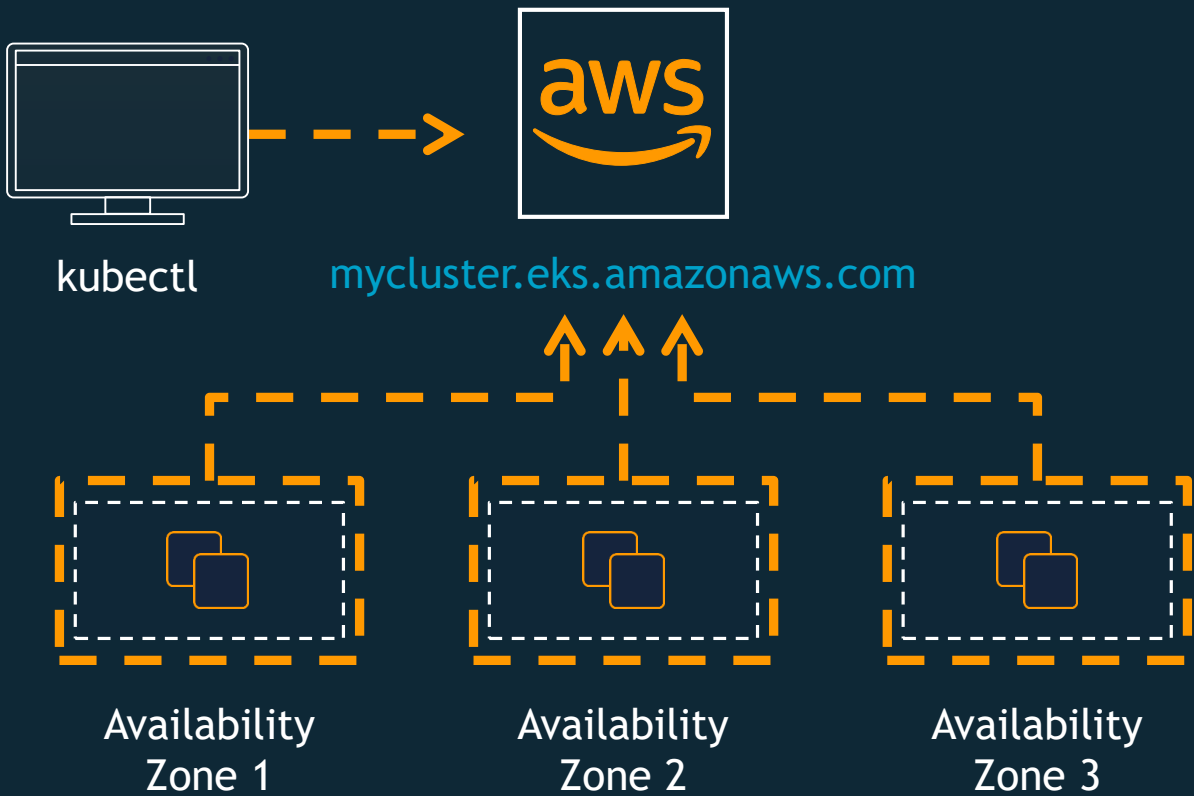
51%

の Kubernetes ユーザーが AWS 上で
ワークロードを動かしている

—CNCF survey

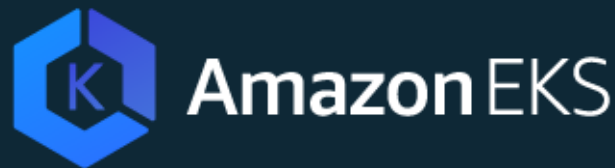


Amazon Elastic Kubernetes Service

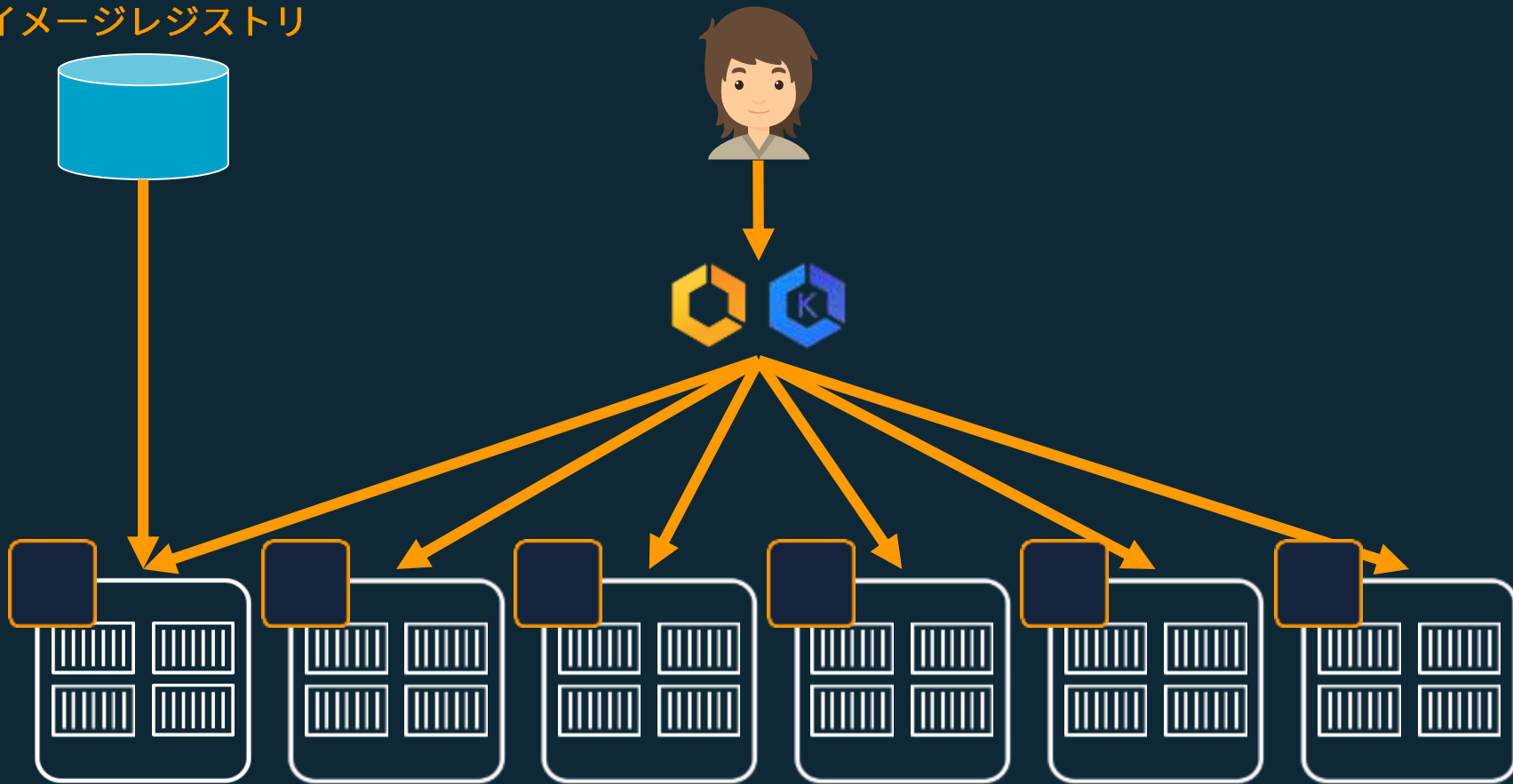


Amazon EKS, Kubernetes

- 運用難易度の高い Kubernetes マスターをマネージドで提供
- エコシステムの OSS やツールを利用できる, CNCF certified
- 各種 AWS サービスとの連携
- EKS サービスチーム、OSS チームによる Kubernetes コミュニティへの貢献
- オープンソース
- “Pod”, ”Deployment”, “Service”, “Job” などのリソースに代表される高い表現力



イメージレジストリ



イメージレジストリ

Amazon Elastic Container Registry (ECR)

- フルマネージドなプライベートコンテナイメージレジストリ
- セキュア – 保管イメージの自動的な暗号化、IAM 連携
- スケーラブルかつ高い可用性
- Docker CLI からの利用
- ECS/EKS/Kubernetes だけでなく、その他コンテナオーケストレーターからも利用可能



コンテナ実行環境



Amazon

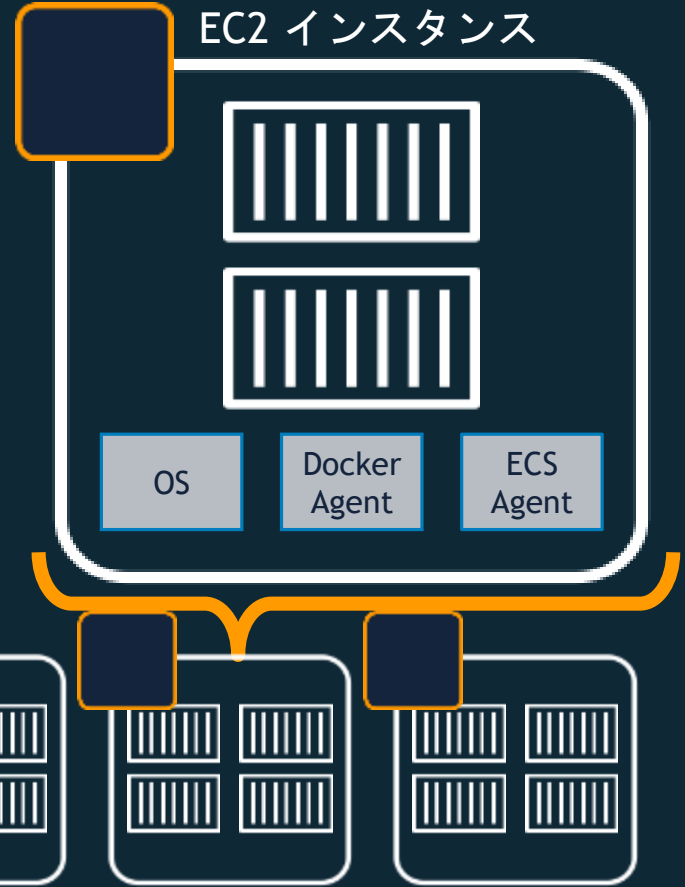
コンテナ実行環境



実行環境 EC2 インスタンスの運用業務

OS やエージェント類へのパッチ当て・更新

実行中のコンテナ数に基づく、最適なリソース
使用率を保つための EC2 インスタンス数のス
ケーリング



AWS Fargate



AWS マネージド

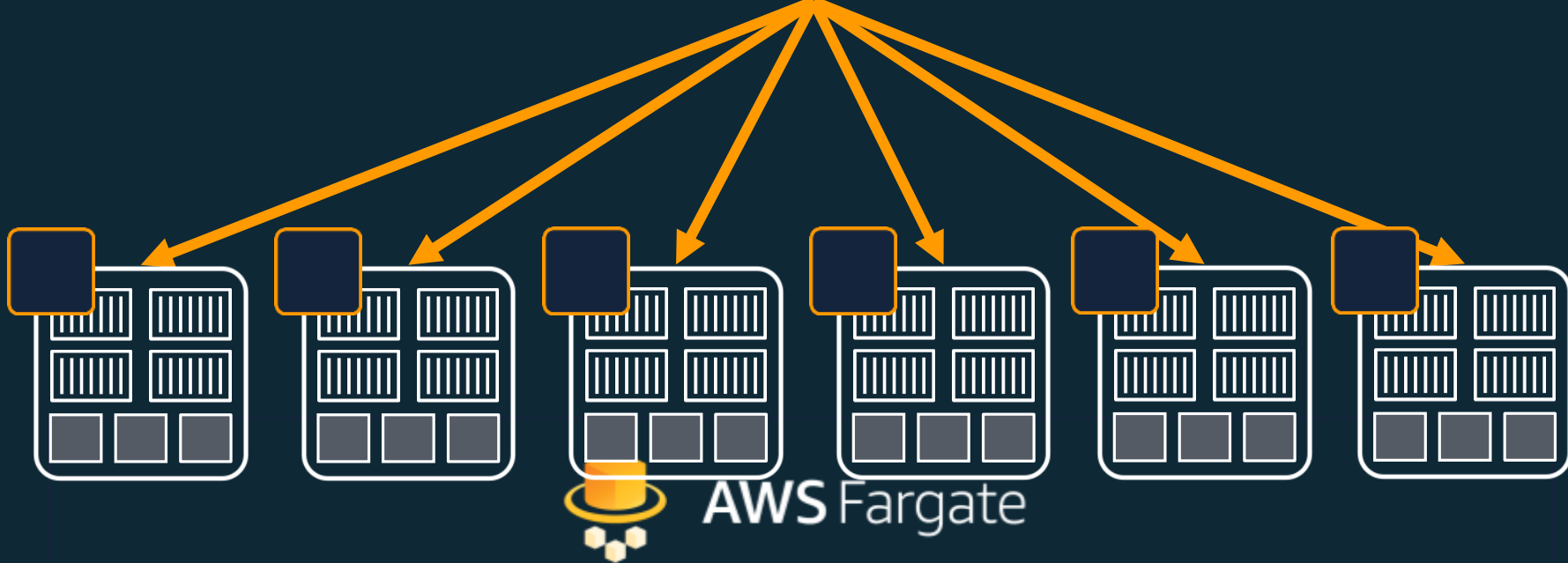
EC2 インスタンスのプロビジョン、スケール、管理不要

コンテナネイティブ

仮想マシンを意識しないシームレスなスケーリング
コンテナの起動時間・使用リソースに応じた料金設定

AWS サービスとの連携

VPC ネットワーキング、Elastic Load Balancing、IAM、
CloudWatch、etc.





Amazon
ECS



AWS Fargate



「このコンテナを
3つのAZに分散させて
10個デプロイして、
このロードバランサーに
つないでください」

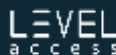
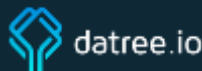
AWS Fargate customers

“インフラの管理をせずにスケールでき、かつネットワークのきめ細やかな制御ができることが私たちにとって必要であり、それこそが Fargate に移行した理由です。”

Product Hunt

“クラスタの子守はしたくない。私たちにとってはクラスタ管理から生まれる価値は何もない。”

Shimon Tolts
CTO, DATREE



その他のコンテナ関連サービス

AWS Cloud Map

New!



AWS
Cloud
Map

クラウドリソースのためのサービスディスカバリ

各リソースに対する継続的なヘルスチェック

ディスカバリ対象サービスの変更に合わせた動的な更新

開発生産性の向上

全アプリケーションリソースをディスカバリ可能な単一のレジストリ
ユーザーフレンドリーな名前の設定

AWS コンテナサービスとの連携

AWS Fargate

Amazon ECS

Amazon EKS

AWS App Mesh



New!

アプリケーションレベルのネットワーク

ログ・メトリクス・トレース情報の容易な出力

クライアントサイドのトラフィック・ルーティングポリシー

単一のクラスタやコンテナサービスを超えて動作

Amazon ECS

Amazon EKS

Kubernetes on EC2

AWS Fargate

Amazon EC2

マネージド

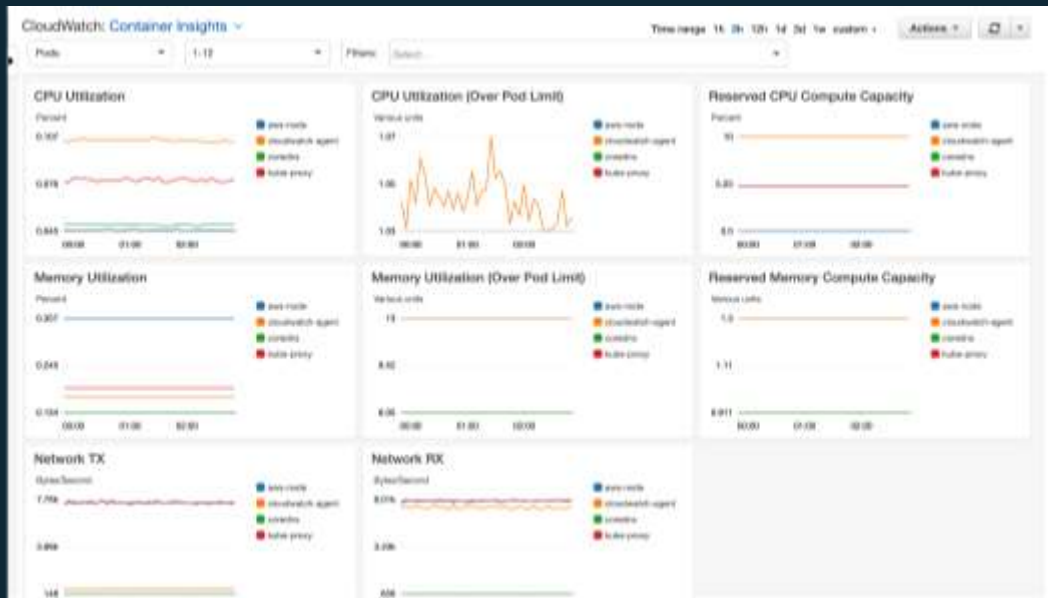
コントロールプレーンの管理不要

容易なオペレーション

高いスケーラビリティ

Amazon CloudWatch Container Insights (パブリックプレビュー)

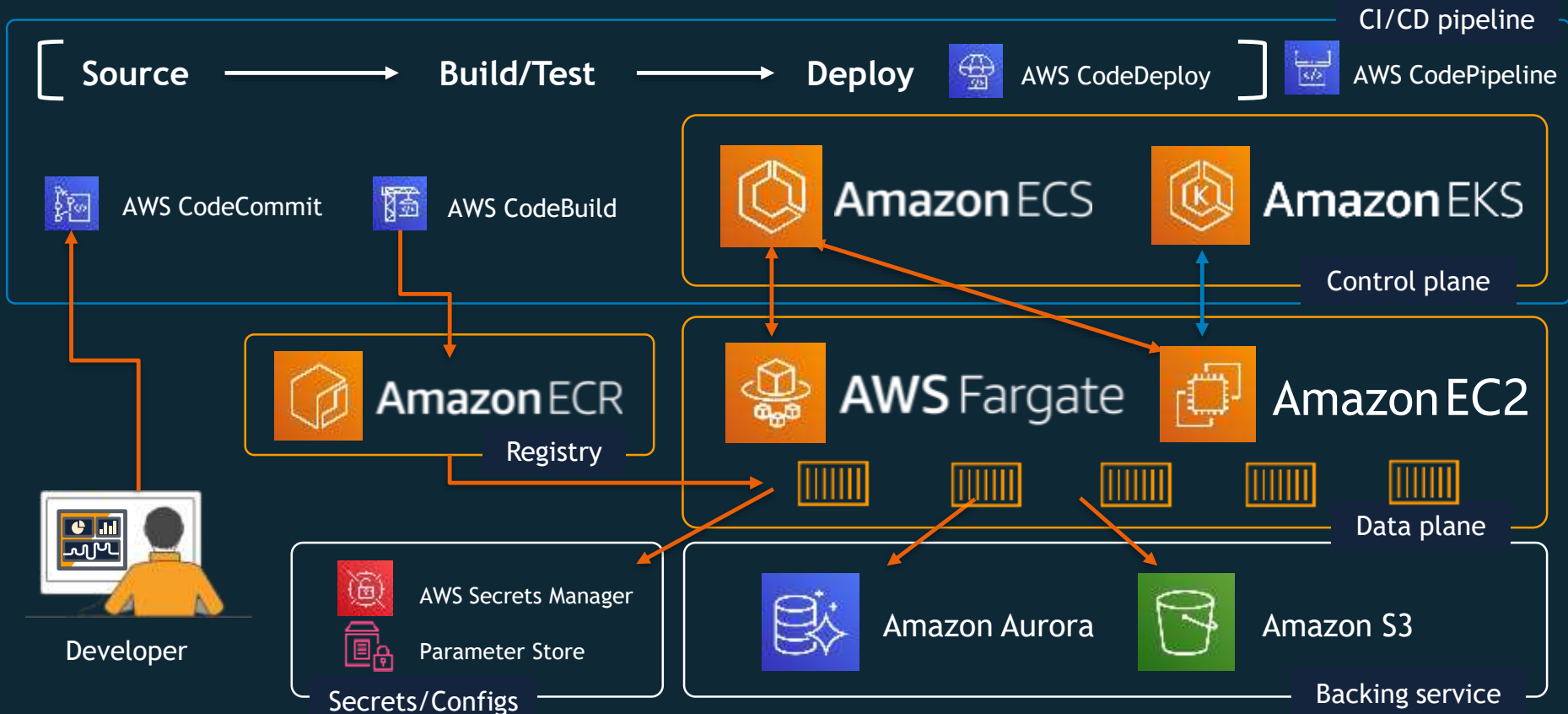
Preview



- コンテナワークロードのためのモニタリングサービス
- 自動ダッシュボードによる可視化、フィルタリング
- CloudWatch Logs Insights 併用による高度な分析
- Amazon ECS, AWS Fargate, Amazon EKS, Kubernetes をサポート

現実世界のコンテナワークロード

周辺を支えるサービス群



コンピューティングの多様な選択肢

運用コスト低

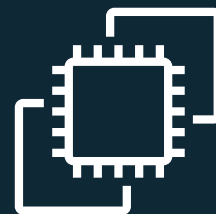
コントロール範囲



Functions



Containers



Virtual Machines

適切なコンピューティングサービスの選択が重要

まとめ: AWS のコンテナ関連サービス

オーケストレーション

コンテナのデプロイ、スケジューリング、スケーリング



Amazon ECS



Amazon EKS

イメージレジストリ

コンテナイメージの格納



Amazon ECR

ホスティング

コンテナ実行環境



Amazon EC2



AWS Fargate

アプリケーションレベル

サービスメッシュ、サービスディスカバリ



AWS App Mesh



AWS

Cloud Map

Thank you!

Yasuhiro Tori Hara
yshr@amazon.co.jp

 toricls