

AWS

S U M M I T

AWS のコンテナ管理入門

(Amazon EC2 Container Service)

アマゾン ウェブ サービス ジャパン株式会社
ソリューションアーキテクト 川村誠
2017年6月1日



自己紹介

- 名前

川村 誠 （かわむら まこと）

- 所属

アマゾン ウェブ サービス ジャパン 株式会社
技術統括本部 ストラテジックソリューション部
ソリューション アーキテクト

- 好きなAWSサービス

Amazon EMR, Amazon ECS



本日お持ち帰りいただきたいこと

- ❑ なぜコンテナなのか？
- ❑ Amazon EC2 Container Service概要

セッションの内容

□ なぜコンテナなのか？

- ❖ コンテナとは
- ❖ コンテナの利点
- ❖ ベストプラクティス・アンチパターン
- ❖ 課題／クラスタ管理

□ Amazon EC2 Container Service 概要

- ❖ 利点
- ❖ アーキテクチャ
- ❖ デモ
- ❖ 事例

なぜコンテナなのか？

なぜコンテナなのか？

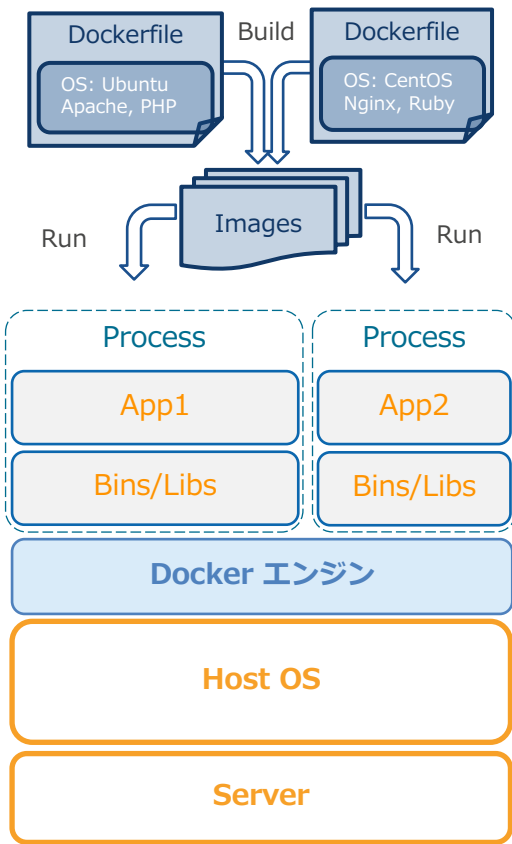
- ❑ コンテナは、真新しい技術ではない
- ❑ コンテナは、リソース隔離が元々の由来
- ❑ コンテナは、最近DevOpsのために再発見された
- ❑ 今や、コンテナはスタートアップからエンタープライズまで支持を得ている

現場の課題

- ❑ 開発に専念したい
- ❑ インフラの運用管理を効率化したい
- ❑ 計算リソースを集約してコストを削減したい
- ❑ もっと速くデプロイ／スケールする仕組みを実現したい

コンテナとは

- ❑ イメージ
- ❑ プロセス隔離
- ❑ 自動化

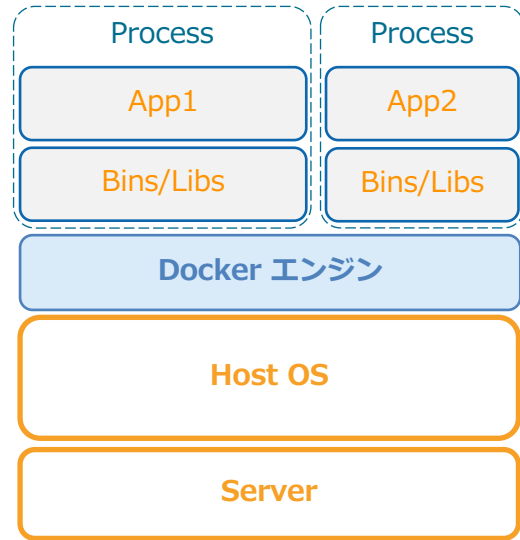


VMと比較すると

- ❑ コンテナはリソースのオーバーヘッドが少ない
- ❑ コンテナイメージもコンパクト



仮想マシン型



コンテナ型

コンテナの利点

❑ 可搬性

- ❖ 不変なイメージ

❑ 柔軟性

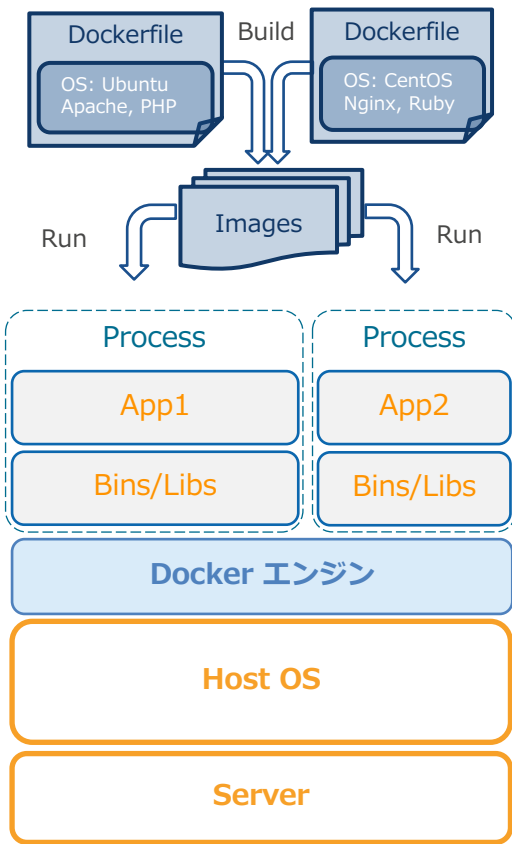
- ❖ 削除、再生成が容易

❑ 速度

- ❖ OSは既に起動済

❑ 効率

- ❖ オーバーヘッド少ない
- ❖ 粒度を細かく利用率を上げられる



ベストプラクティス / アンチパターン

ベストプラクティス

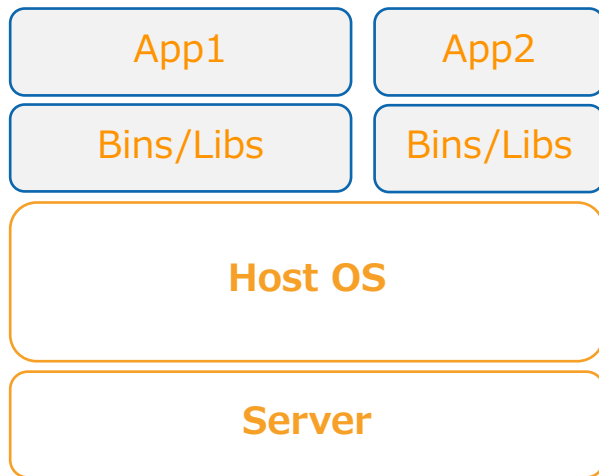
- ❑ アプリをコンテナに適応させる
 - ❖ 12-Factor app
<http://12factor.net/>
- ❑ 複雑さを避ける
 - ❖ シンプルに保つ
- ❑ タスクに集中する
 - ❖ タスク = ジョブの単位
 - ❖ タスク = 1つ以上のコンテナ
 - ❖ ポータブルに

アンチパターン

- ❑ VMベースの処理やワークフロー
 - ❖ コンテナをVMの様に考える
- ❑ 複雑さを上げてしまう
 - ❖ 多すぎる技術が複雑さを増す
- ❑ ホスト単位で何かを実行させる
 - ❖ 出来る限りホストのことは忘れる

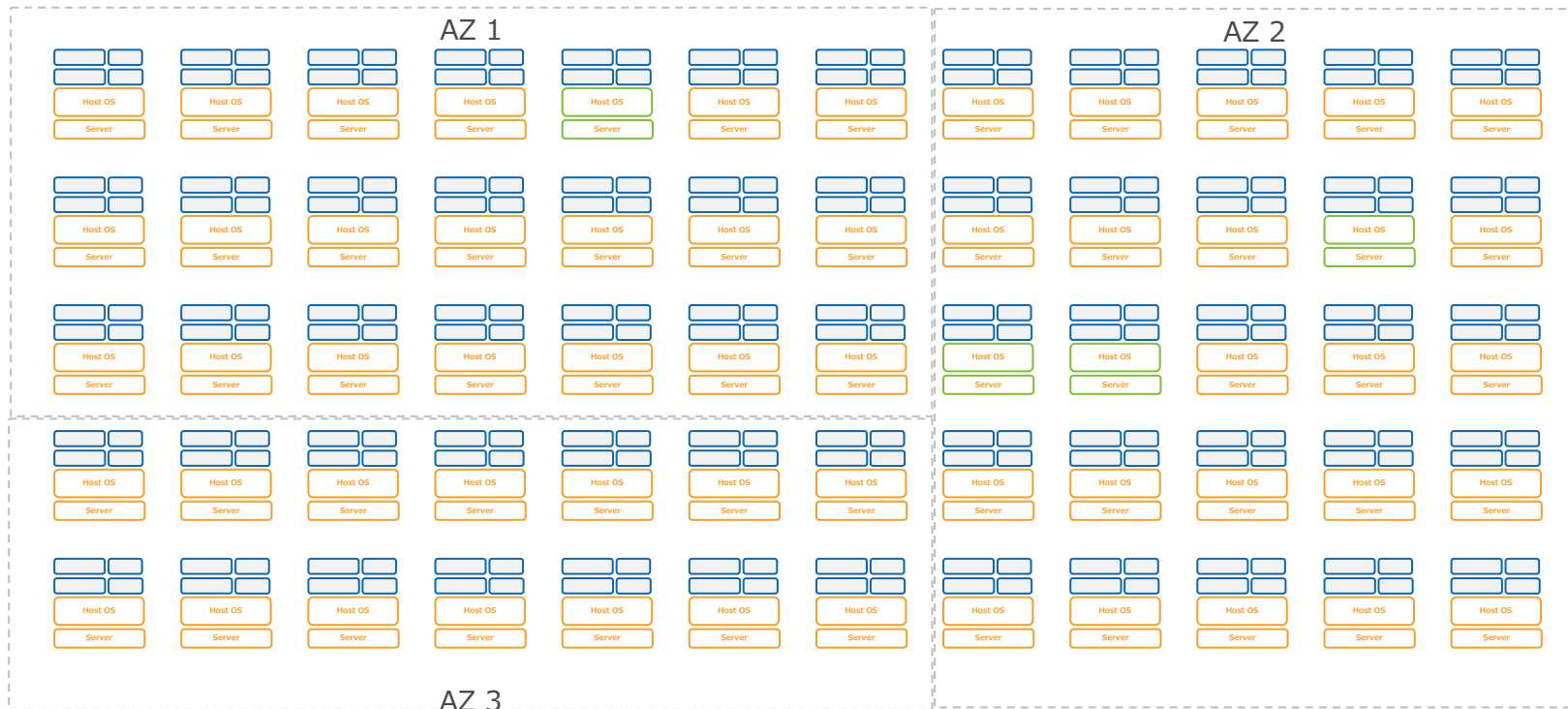
課題／クラスタ管理

1台のサーバ上でコンテナを扱うのは簡単



課題／クラスタ管理

しかし、複数台のクラスタ上で管理するのは困難



課題／クラスタ管理

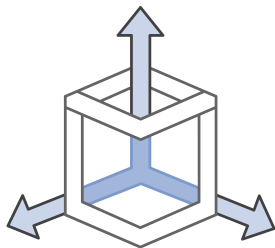
- ❑ 複数ホスト上でのコンテナ管理は非常に難しい
 - ❖ 配置、状態の追跡、監視、自動化等
 - ❖ 想像以上に、たくさんのことを考慮する必要がある
- ❑ 多くのお客様はアプリケーションの開発に専念したいはず
 - ❖ 内製のコンテナ管理システムは、車輪の再発明
 - ❖ ビジネスの差別化にはつながらない

Amazon EC2 Container Service 概要

Amazon EC2 Container Service の利点

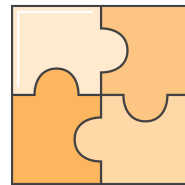
簡単に、どんなスケールのクラスタも管理できる

クラスタ管理はお任せ
状態管理、操作、監視
スケーラブル



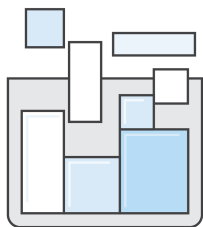
他のAWSサービスとの連携がデザインされている

Elastic Load Balancing, Amazon VPC, AWS IAM, AWS CloudTrail



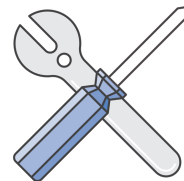
柔軟なコンテナの配置

アプリケーション
バッチジョブ
複数のスケジューラ



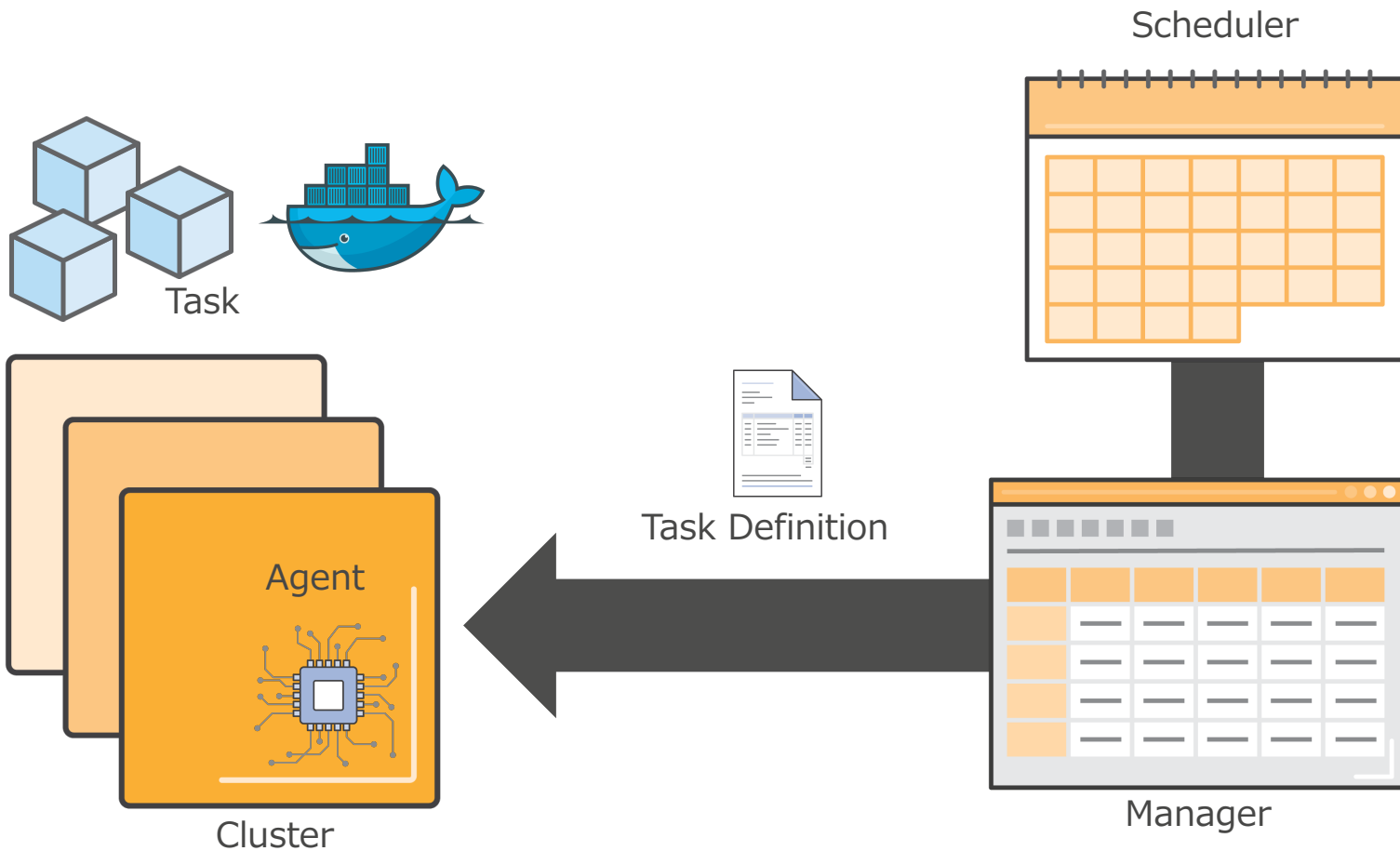
拡張性の高さ

わかりやすいAPI
オープンソースエージェント
カスタムスケジューラ

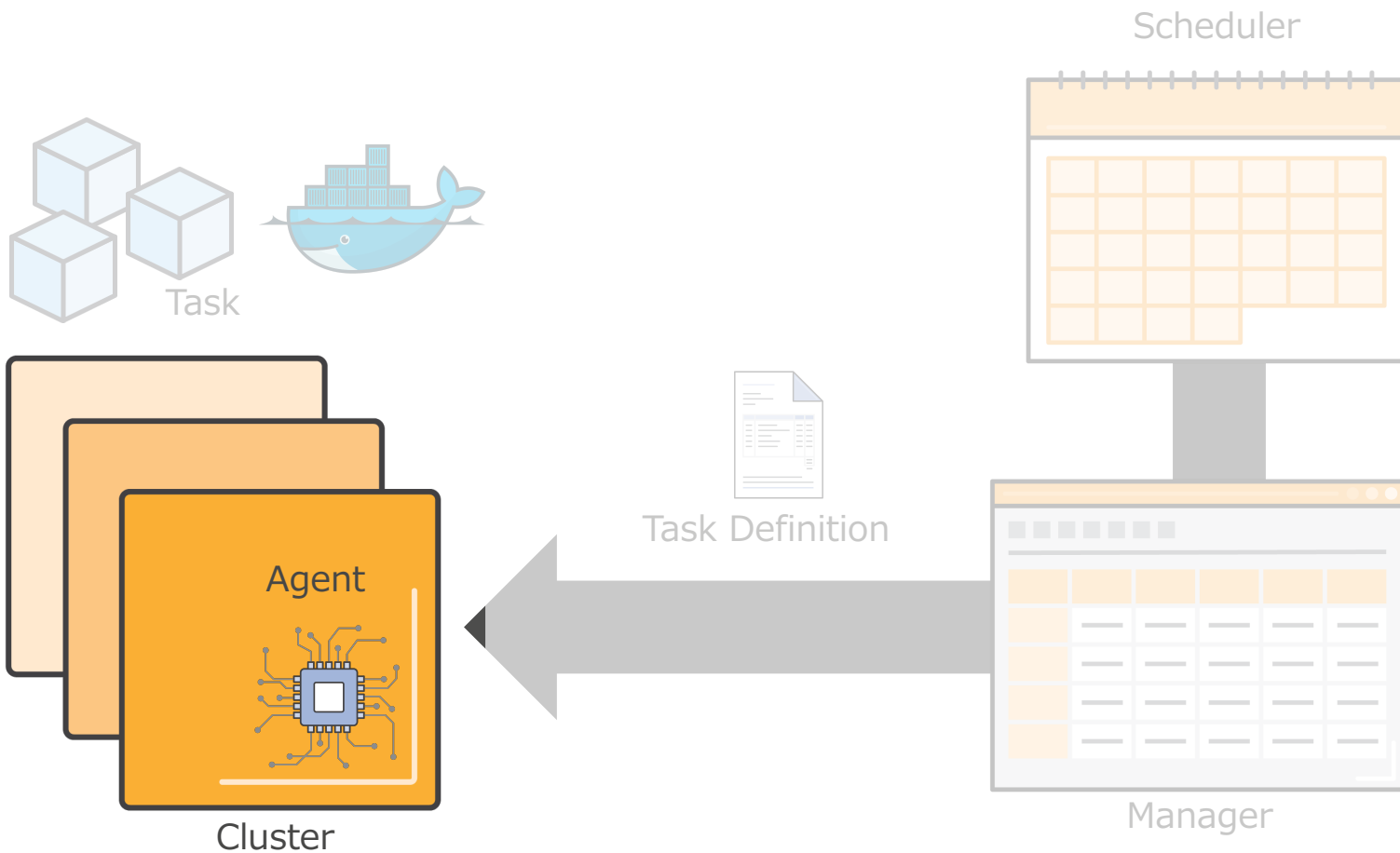


アーキテクチャ

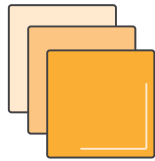
アーキテクチャ



アーキテクチャ

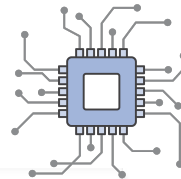


主要コンポーネント: Clusters

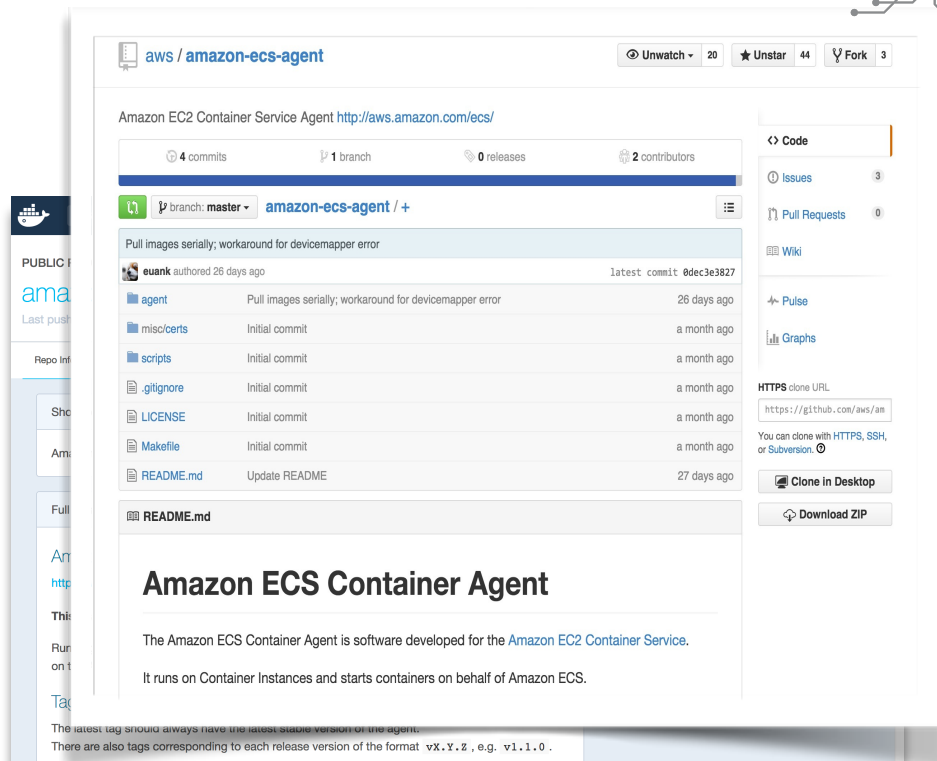


- ❑ Taskが実行されるコンテナインスタンス群で構成されるリソースプール
- ❑ VPC 内の任意のEC2インスタンスをコンテナインスタンスとして利用可能
- ❑ コンテナインスタンス上では、Docker daemon と Amazon ECS agent が動いている必要がある
 - ❖ Amazon ECS-optimized AMI

主要コンポーネント: Amazon ECS Agent

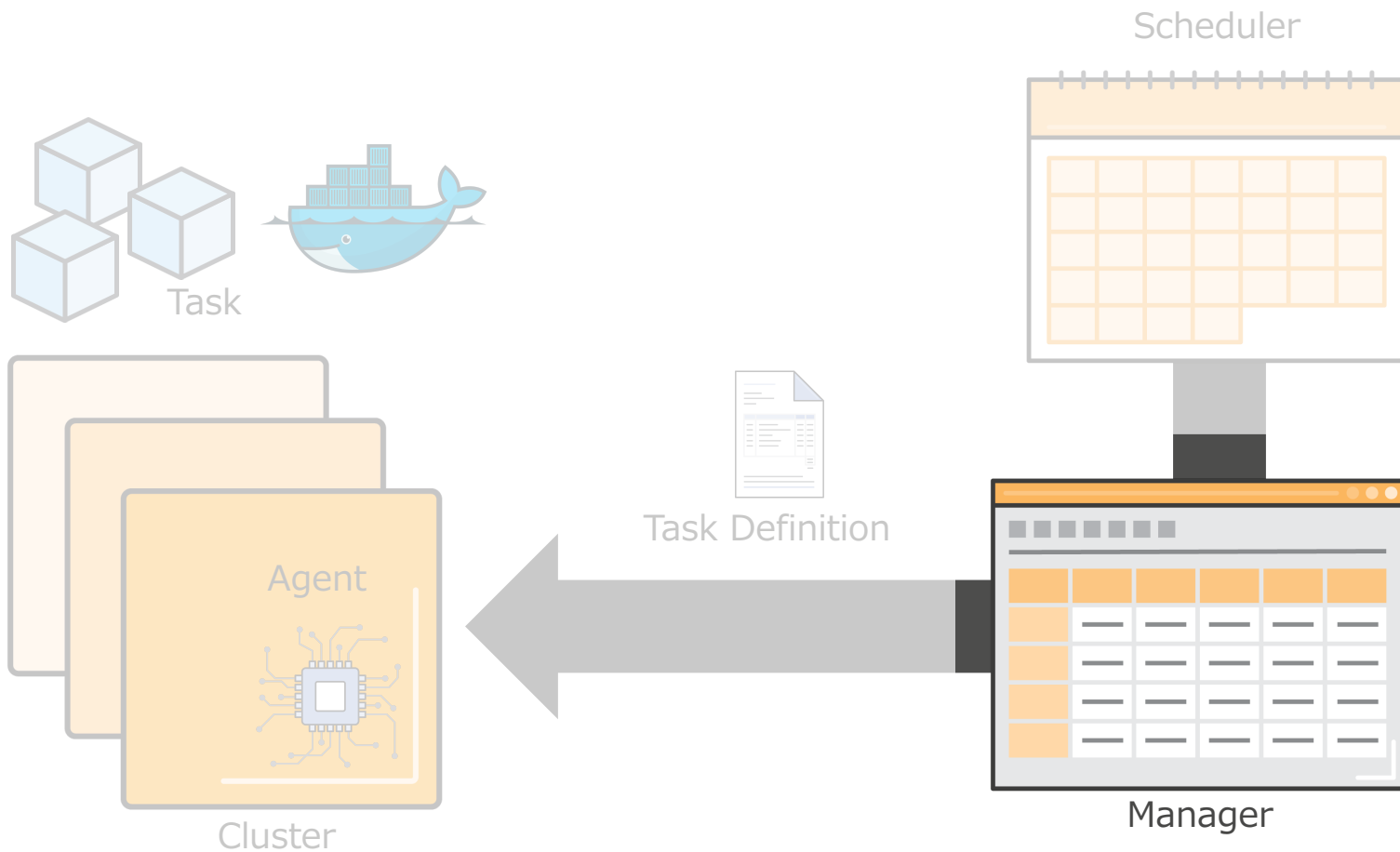


- ❑ コンテナインスタンスと Manager の連携を司る
- ❑ GitHub/Dockerhubで OSSとして公開されています



<https://github.com/aws/amazon-ecs-agent>
<https://hub.docker.com/r/amazon/amazon-ecs-agent/>

アーキテクチャ

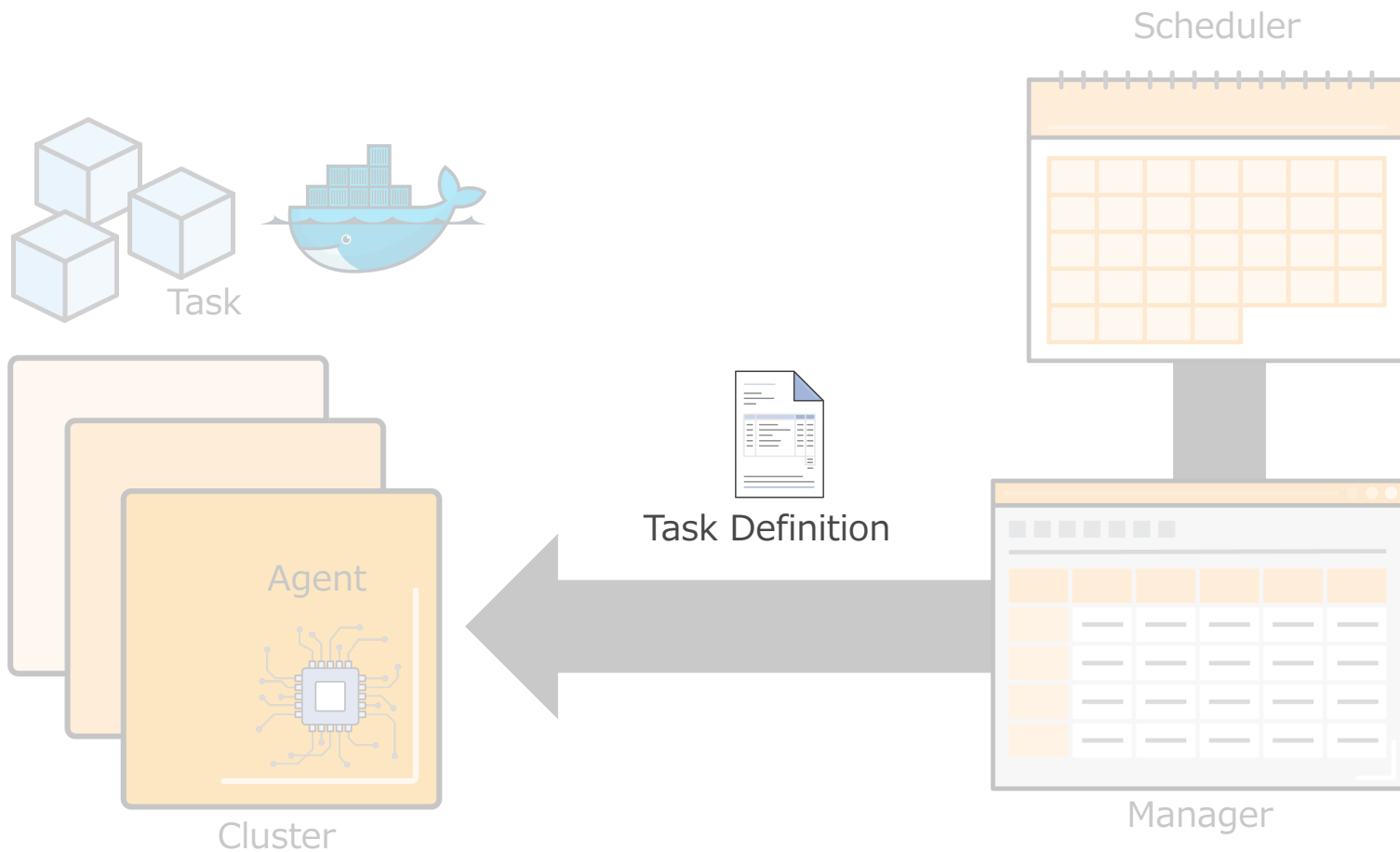


主要コンポーネント: Manager



- ❑ 利用可能なClusterのリソースとTaskの状態を管理
- ❑ リソースの変化を追跡
- ❑ リソースへのリクエストを受け付ける
- ❑ 正確性と一貫性を保証する

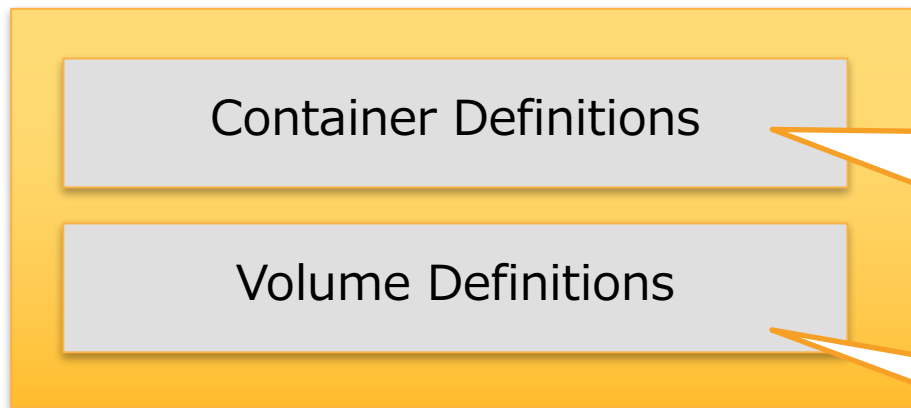
アーキテクチャ



主要コンポーネント: Task Definitions



□ Taskで使用するコンテナ、及び周辺設定の定義

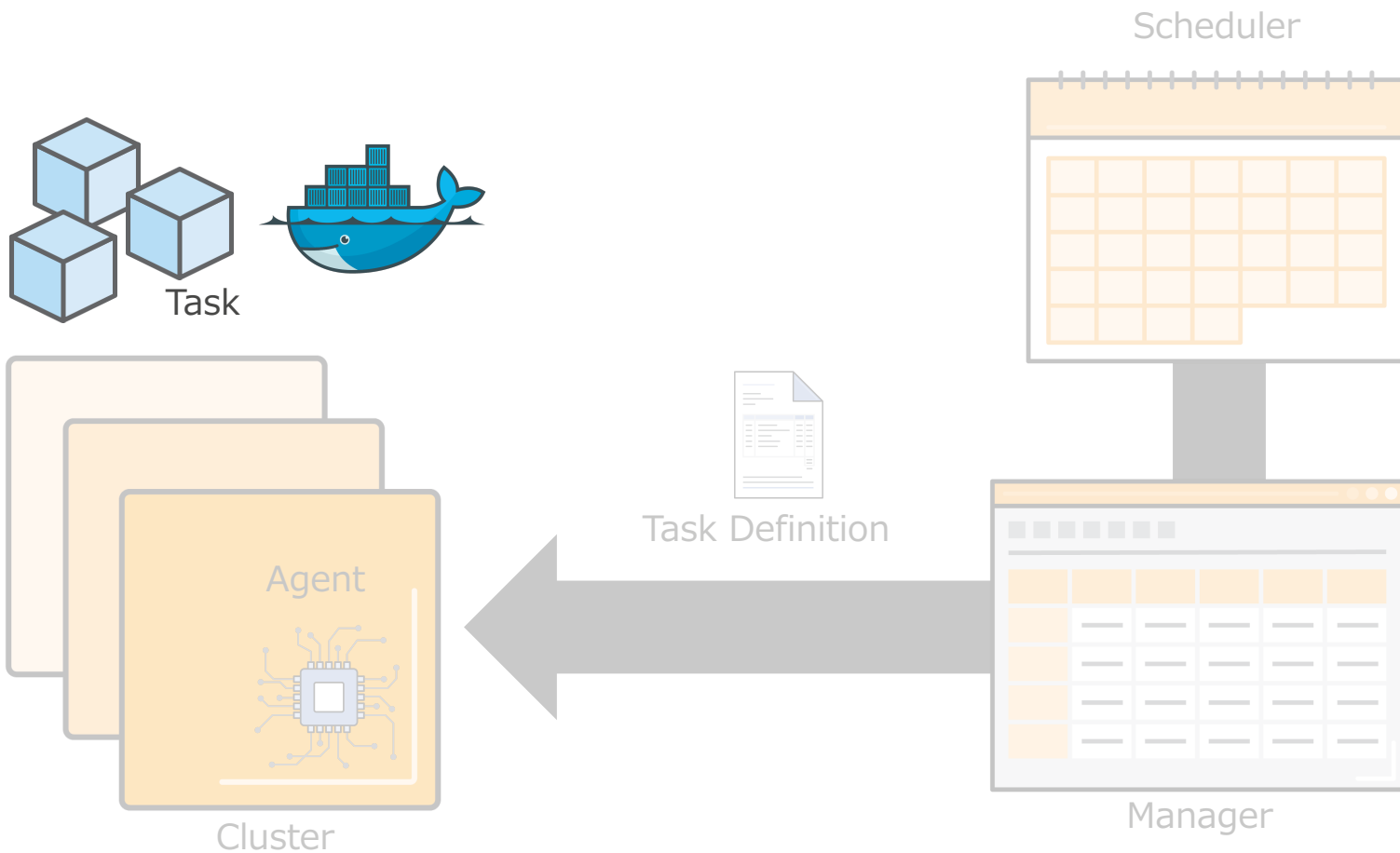


稼働させるContainerの情報を定義する

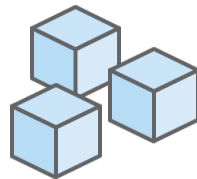
- イメージ定義
- リソース制限(CPU/メモリ)
- ポートマッピング
- …等

Container Instanceのファイルシステムをマウントする場合、定義する

アーキテクチャ

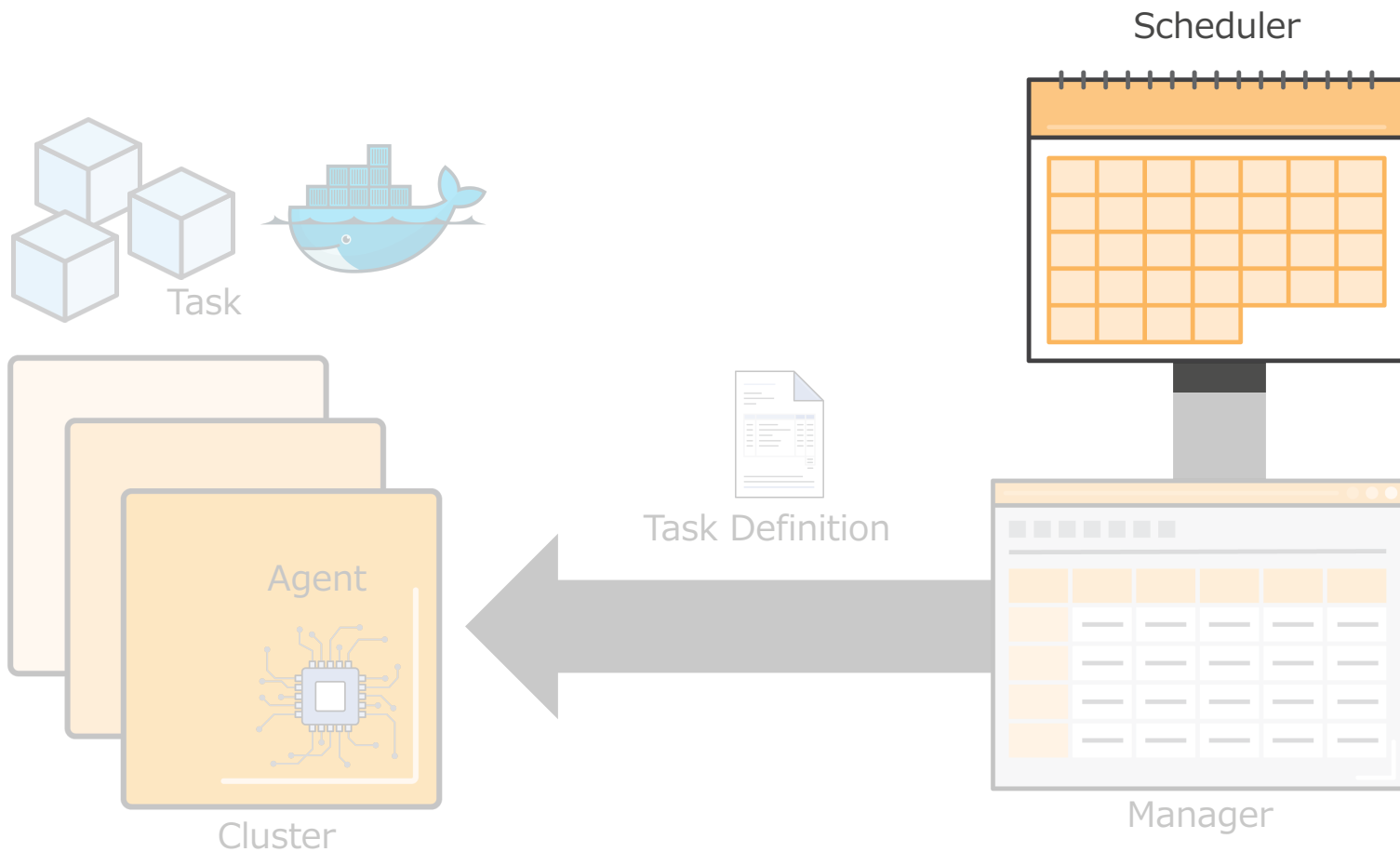


主要コンポーネント: Tasks



- ❑ コンテナ実行の 1 単位
- ❑ 関連するコンテナでグループ化される
- ❑ Task Definitionが実体化されたもの
 - ❖ 1つのTask Definitionから複数のTaskが生成
- ❑ リソースに余裕のあるContainer Instanceで実行される

アーキテクチャ



主要コンポーネント: Scheduler



Run Task

バッチジョブのようなワークロードに最適

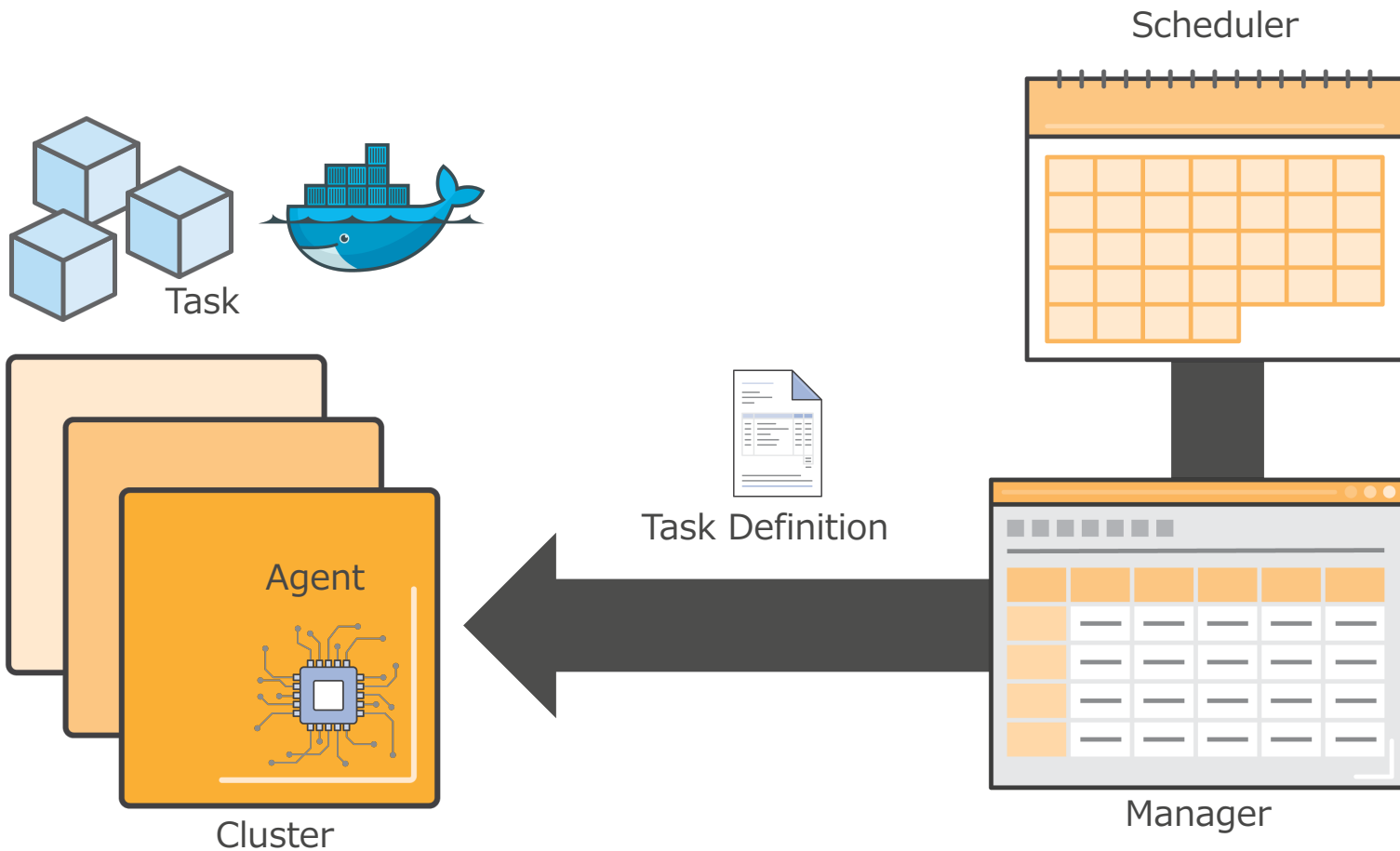
- ❑ Task Definition から指定した数の Task を実行する
- ❑ コンテナインスタンスは自動的に選択される

Service

Web/APIの様に、長期稼働するワークロードに最適

- ❑ Task を必要数維持するスケジューラ
 - ❖ 自動復旧 / Auto Scaling にも対応
- ❑ 新しいコンテナをデプロイしつつの切り替えが可能
- ❑ Elastic Load Balancing との連携も可能

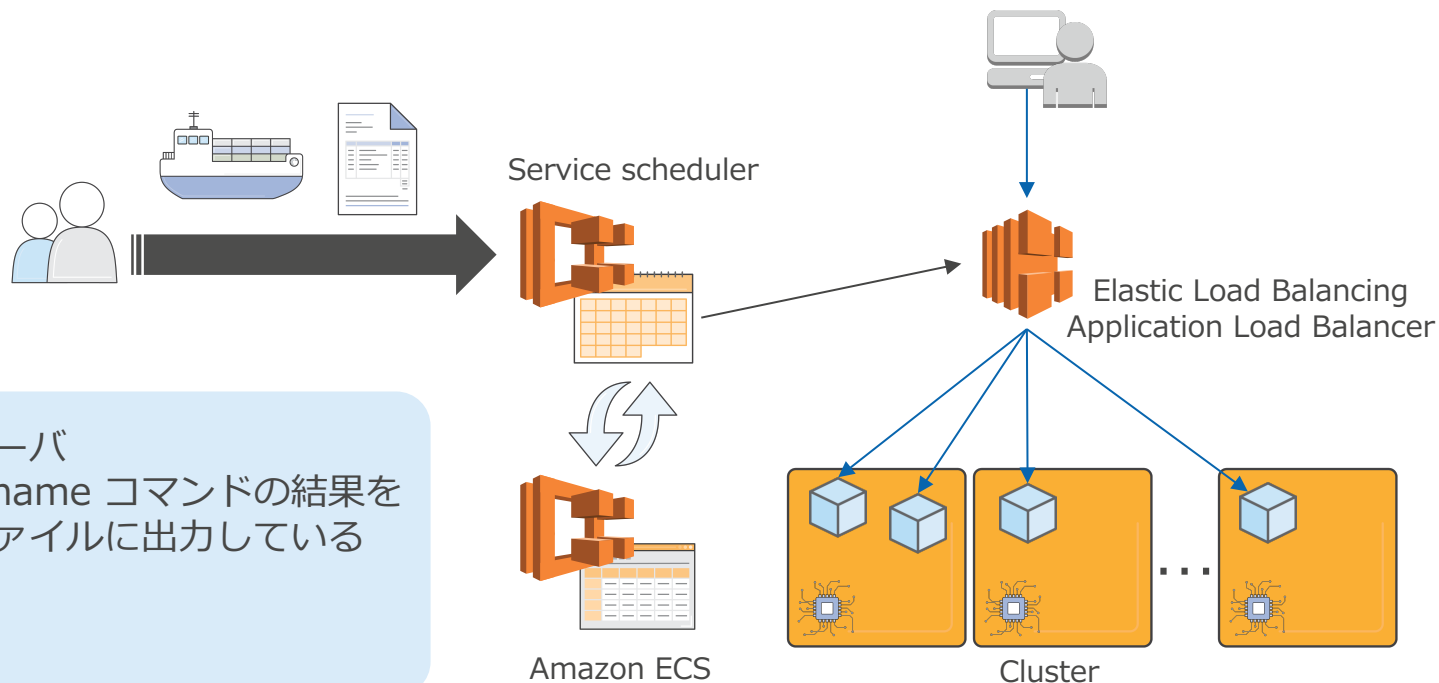
アーキテクチャ



Demo

Demo

❑ 構築するサービス



Demo

❑ 構築の流れ

ステップ 1: リポジトリの設定

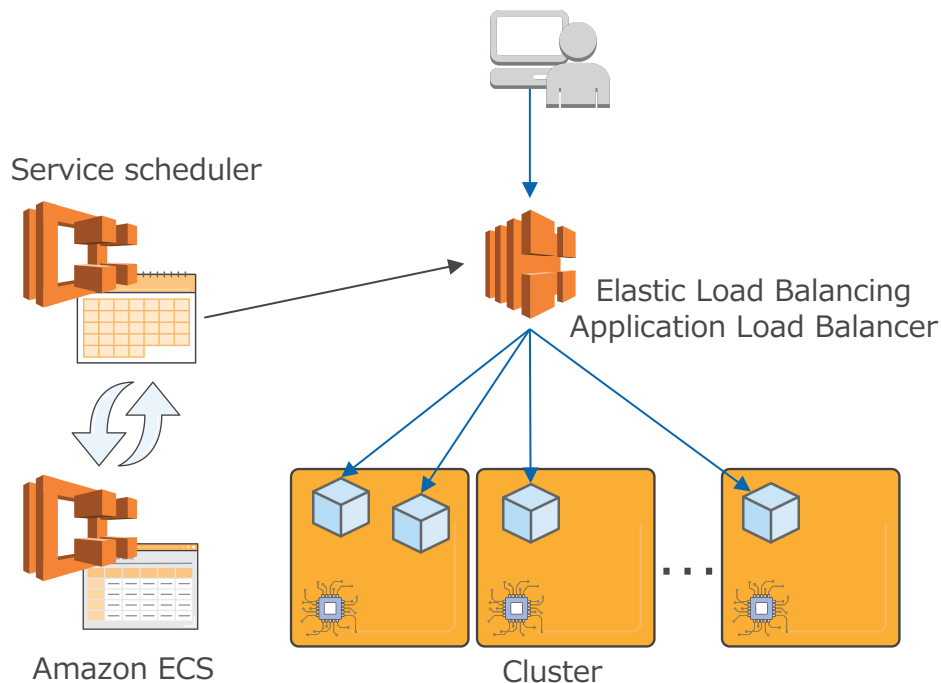
ステップ 2: Docker イメージの構築、
タグ付け、プッシュ

ステップ 3: Task Definition の作成

ステップ 4: Service の設定

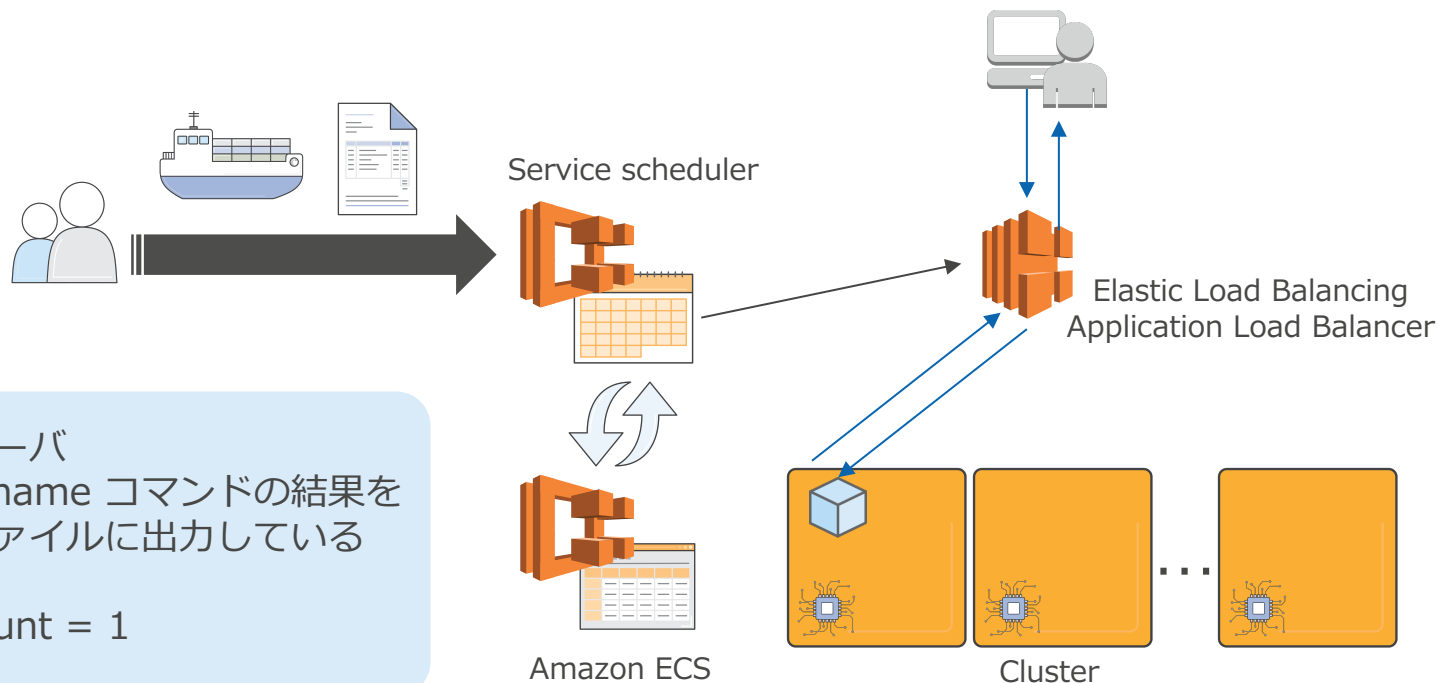
ステップ 5: Cluster の設定

ステップ 6: 確認・実行



Demo

❑ 構築完了後、動作確認を実施

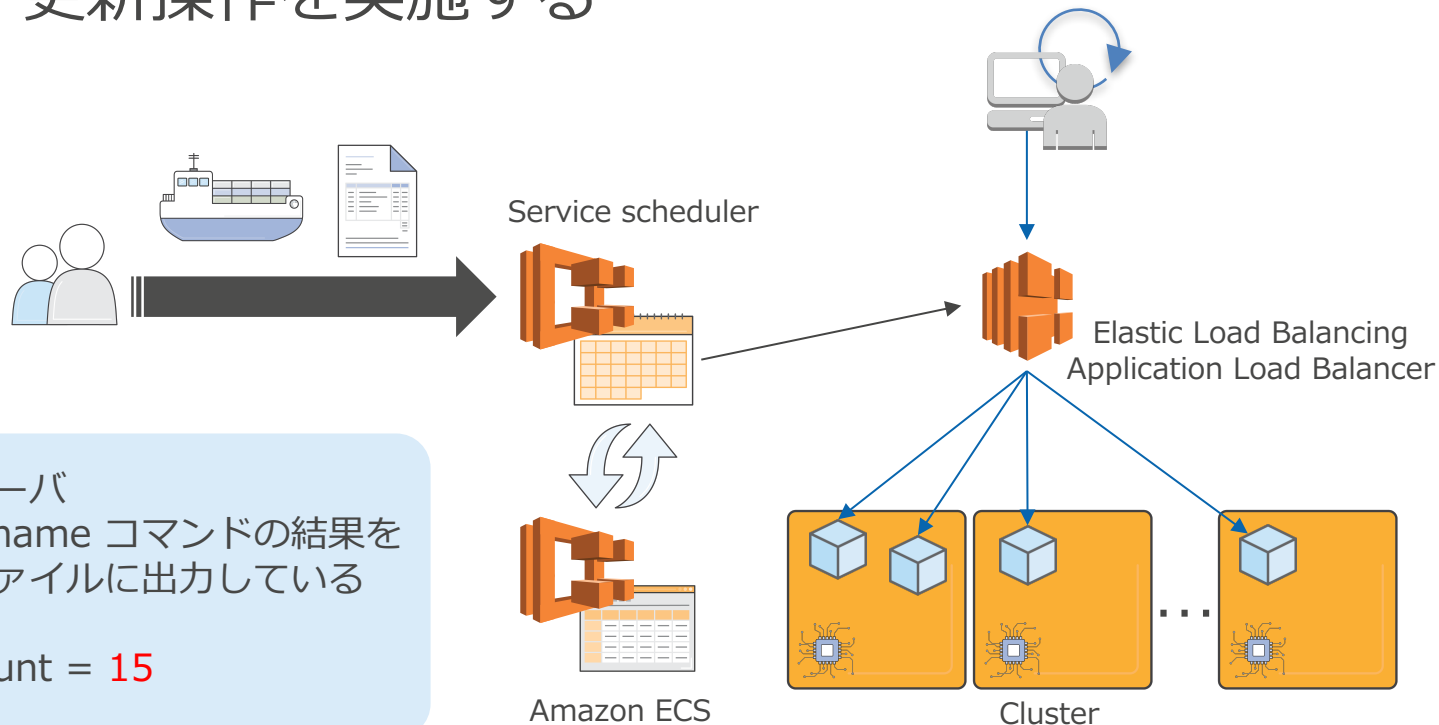


Task  = Webサーバ
起動時に hostname コマンドの結果を
index.html ファイルに出力している

Desired Task Count = 1

Demo

□ その後、更新操作を実施する



Task  = Webサーバ
起動時に hostname コマンドの結果を
index.html ファイルに出力している

Desired Task Count = 15

Amazon EC2 Container Service (ECS)



Amazon ECS makes it easy to deploy, manage, and scale Docker containers running applications, services, and batch processes. Amazon ECS places containers across your cluster based on your resource needs and is integrated with familiar features like Elastic Load Balancing, EC2 security groups, EBS volumes and IAM roles.

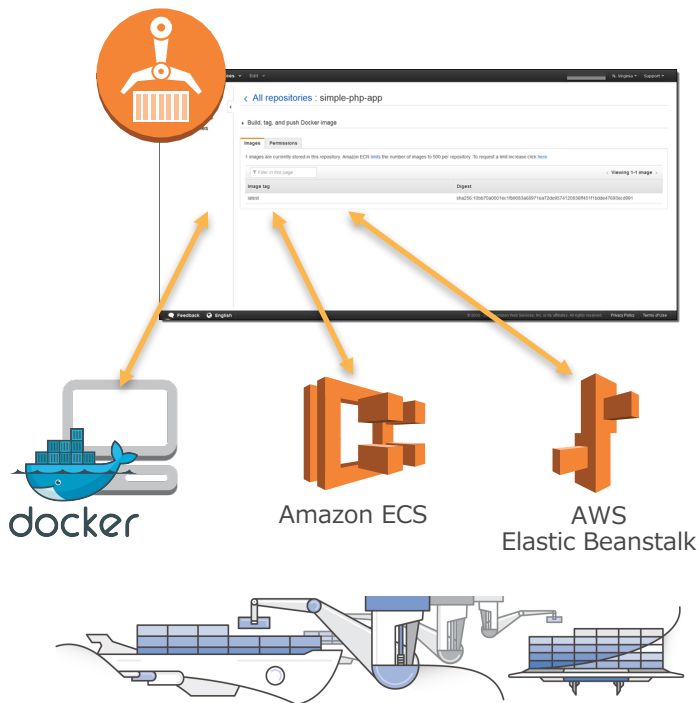
[Get started](#)

[Learn more about Amazon ECS](#)

Amazon EC2 Container Registry



フルマネージドで使えるDockerレジストリサービス



特徴 (<https://aws.amazon.com/ecr/>)

- 高可用性、スケラブル、IAM連携、暗号化
- Dockerコマンドからシームレスに利用可能
- Amazon ECSなども連携済で簡単にデプロイ
- 多くのパートナーが既に連携済のレジストリ

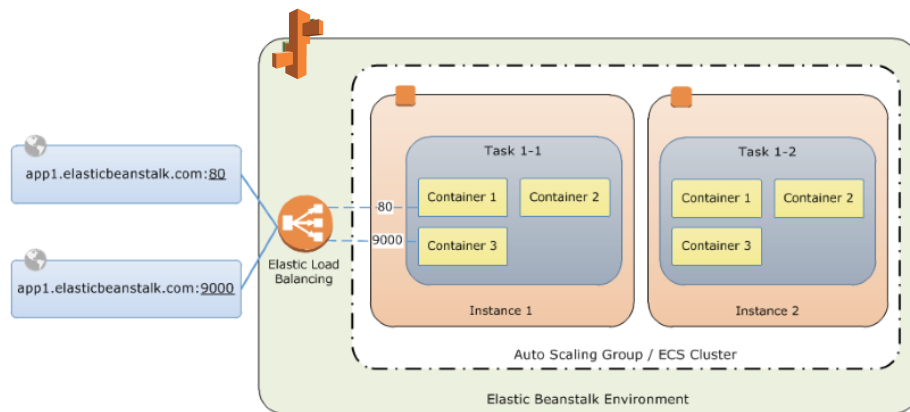
価格体系 (<https://aws.amazon.com/ecr/pricing/>)

- イメージの容量に対して課金 (\$0.1/GB/月)
- 転送量課金(他のAWSサービス同様)
 - 無料: 全てのINと同一リージョンへのOUT
 - 他リージョン、オンプレ等へのOUTは転送量に応じて

ECSを利用するAWSサービス : AWS Elastic Beanstalk

インフラ構成の構築・アプリデプロイの自動化サービス

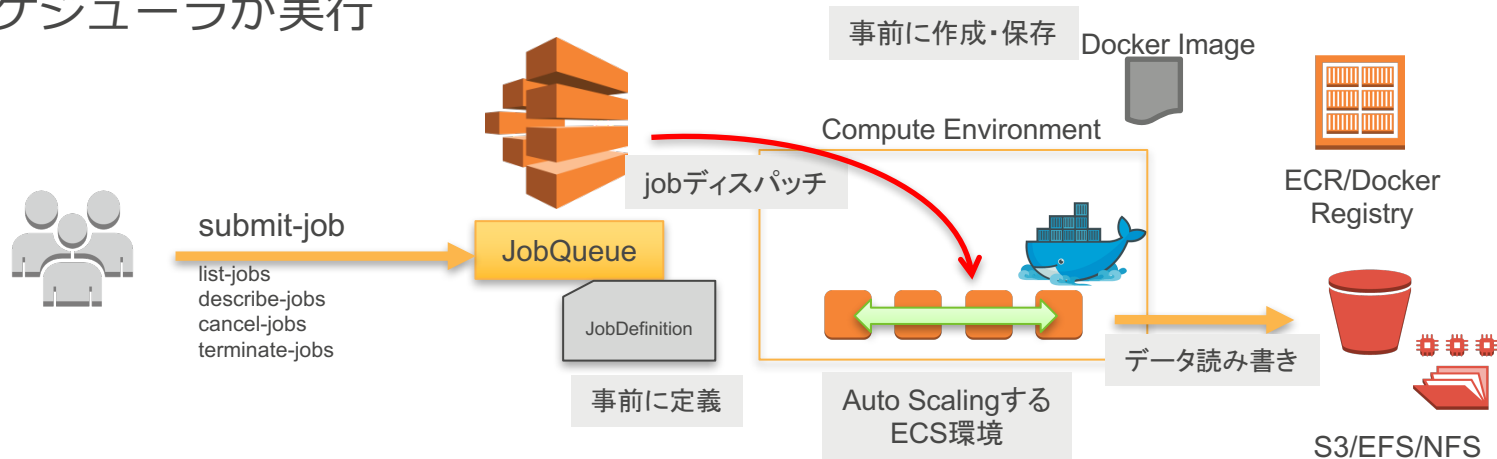
- ❑ 速く簡単にアプリケーションをデプロイ可能
- ❑ インフラの準備&運営からアプリケーションスタックの管理まで自動化
- ❑ Docker コンテナを使用することで、独自のランタイム環境が定義可能
 - ❖ 複数コンテナを利用する場合にECSを使用
 - ❖ Dockerrun.aws.jsonを使ったより柔軟な構成が可能



ECSを利用するAWSサービス : AWS Batch

フルマネージド型のバッチ処理実行サービス

- ❑ 多量のバッチジョブ実行をクラスタの管理を行うことなしに容易に実現できる
- ❑ 利用するインスタンスタイプや数、スポット利用有無などはCompute Environmentとして任意に指定可能
- ❑ ジョブとして登録したアプリケーションやコンテナイメージをスケジューラが実行



事例



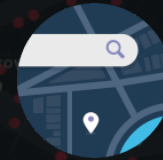
Maps



Developer tools



Directions



Geocoding

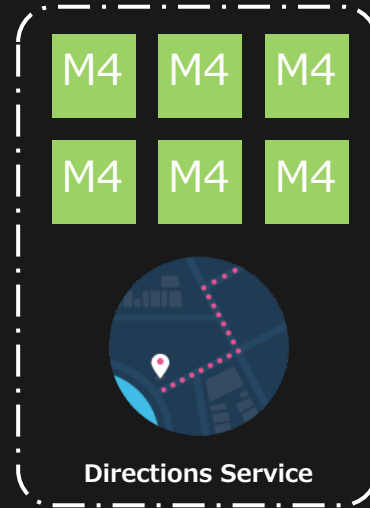
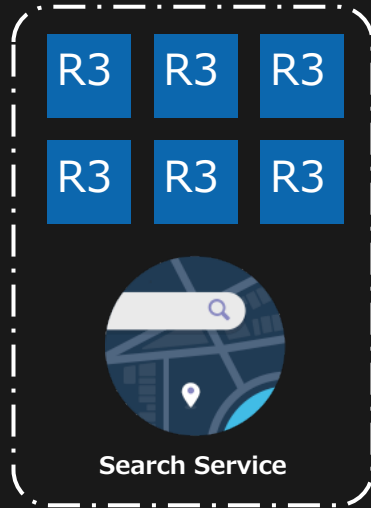


Analysis

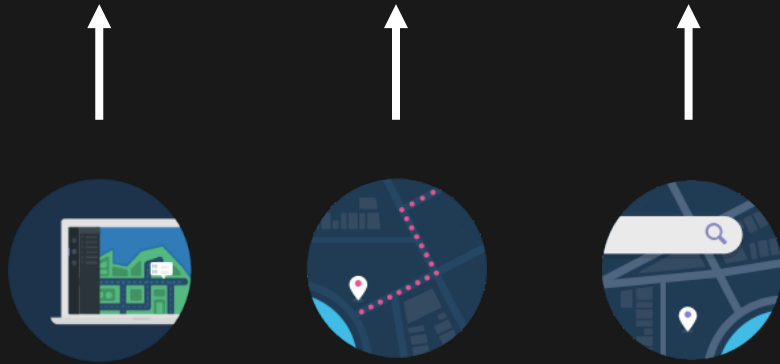
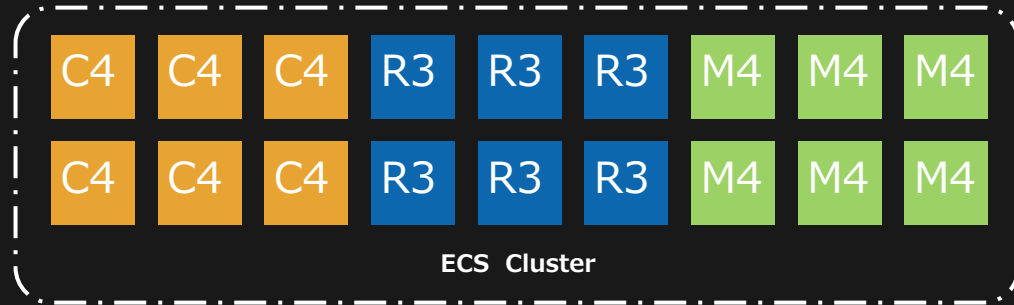


Mobile

<https://www.youtube.com/watch?v=UR8BI2Exkbc>



<https://www.youtube.com/watch?v=UR8BI2Exkbc>



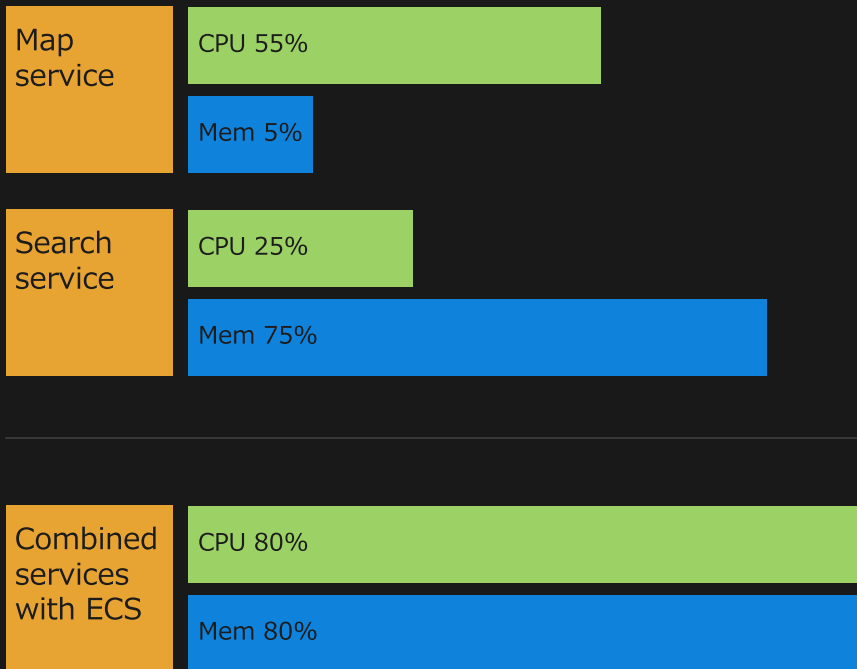
Map Service

Directions Service

Search Service

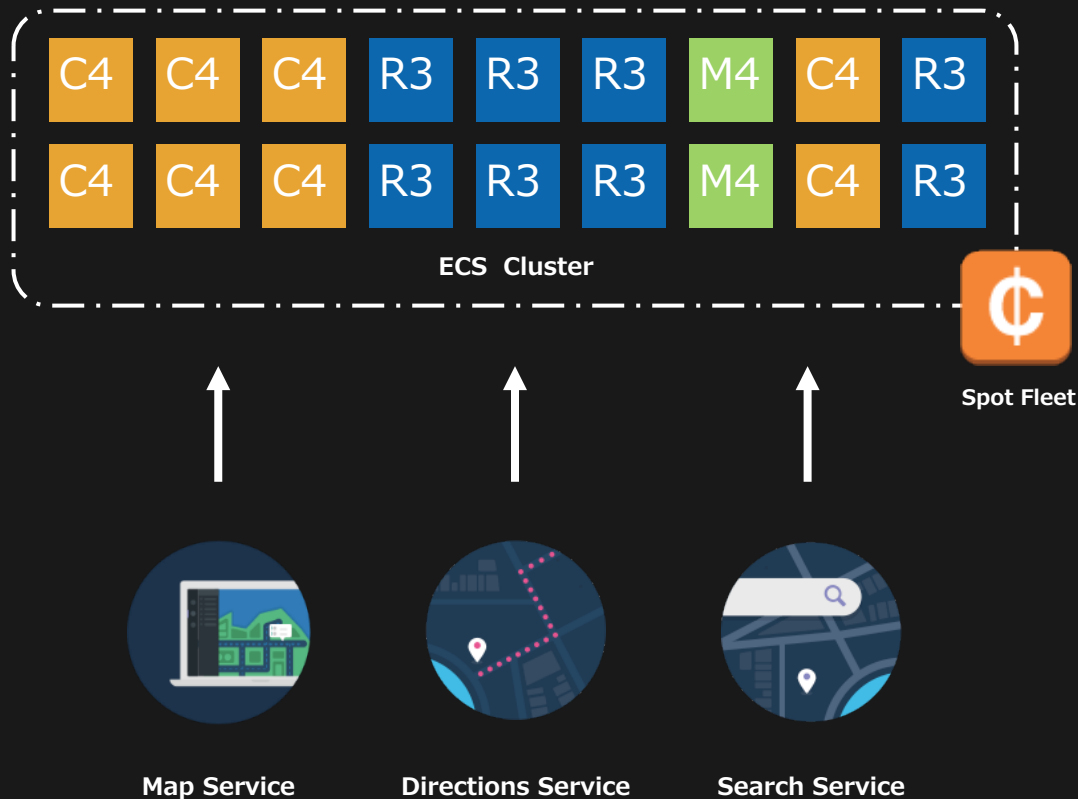
<https://www.youtube.com/watch?v=UR8BI2Exkbc>

インスタンスに
効率良く詰め込
むことで、無駄
を削減する



<https://www.youtube.com/watch?v=UR8BI2Exkbc>

インスタンスを
多様化して使う
ことでSpotの安
定性を獲得



<https://www.youtube.com/watch?v=UR8BI2Exkbc>

25%

より少ないインスタンス数

80-90%

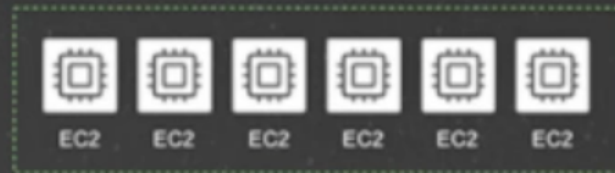
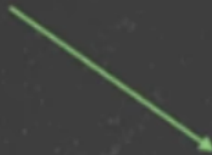
EC2の毎月のコスト削減

<https://www.youtube.com/watch?v=UR8BI2Exkbc>

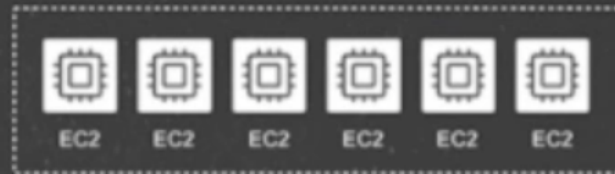
事例: Nextdoor AWS Summit San Francisco 2017 Keynoteより

Red/Blackでゼロダウンタイムデプロイを実現していた

Elastic Load Balancer



Build X+1



Build X

<https://www.youtube.com/watch?v=fqXa0RKJvJE>

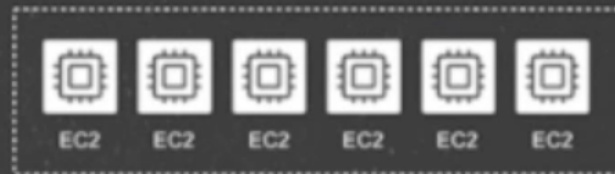
事例: Nextdoor AWS Summit San Francisco 2017 Keynoteより

Red/Blackでゼロダウンタイムデプロイを実現していた

Elastic Load Balancer



Build X+1



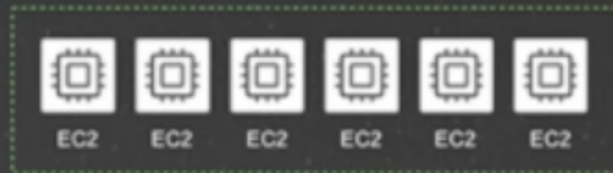
Build X

<https://www.youtube.com/watch?v=fqXa0RKJvJE>

事例: Nextdoor AWS Summit San Francisco 2017 Keynoteより

Red/Blackでゼロダウンタイムデプロイを実現していた

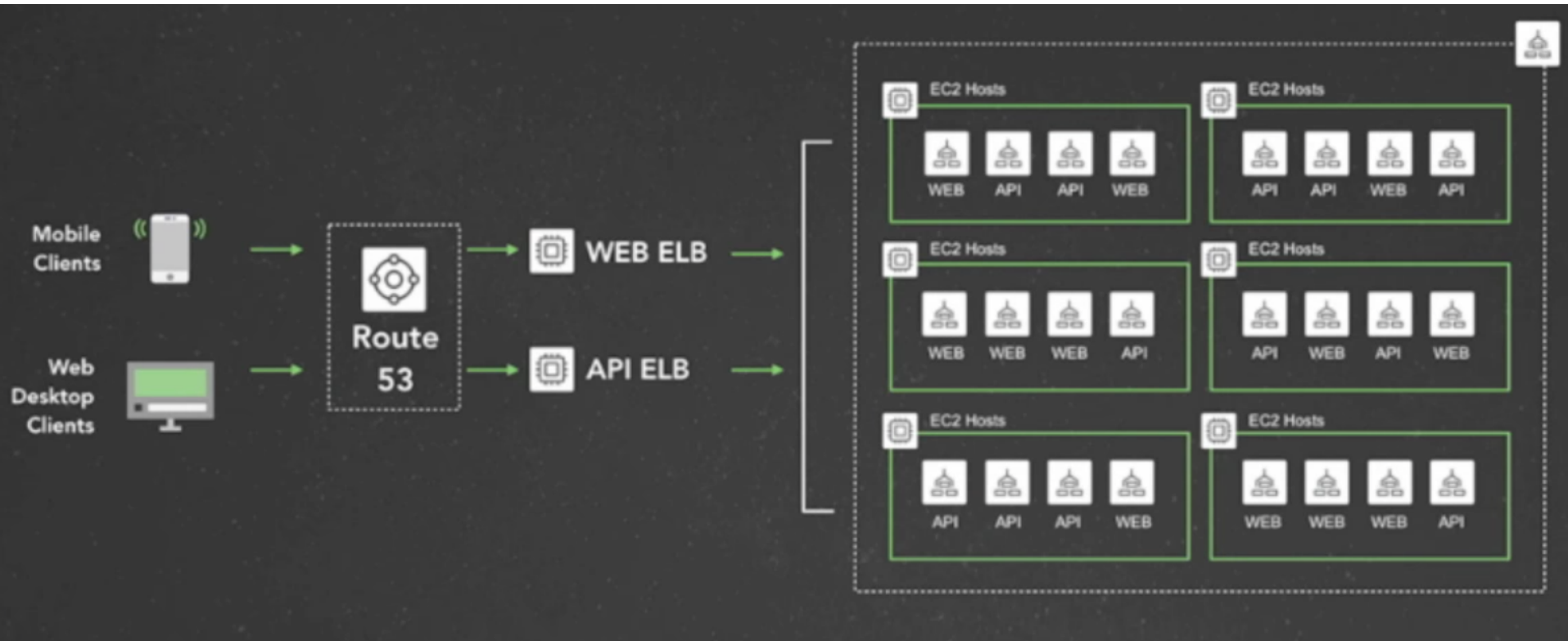
Elastic Load Balancer



Build X+1

<https://www.youtube.com/watch?v=fqXa0RKJvJE>

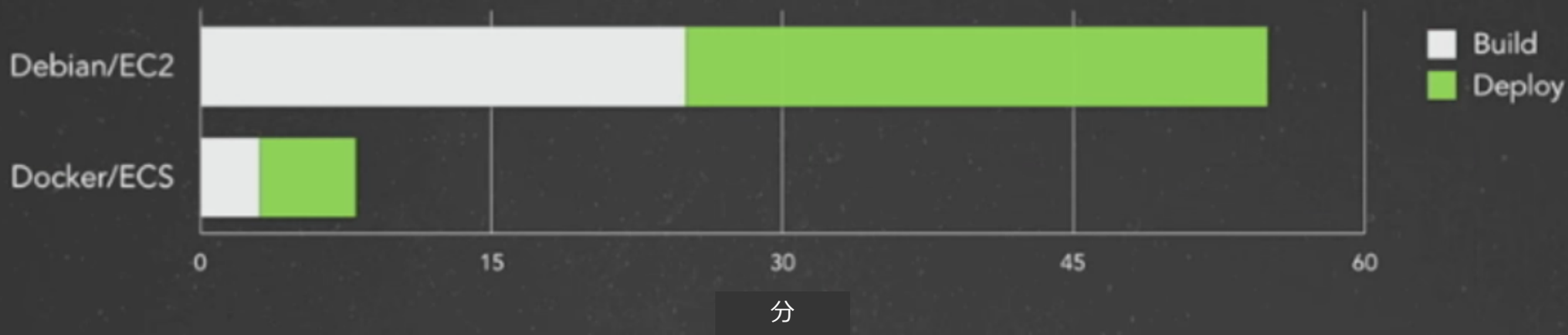
事例: Nextdoor AWS Summit San Francisco 2017 Keynoteより



<https://www.youtube.com/watch?v=fqXa0RKJvJE>

事例: Nextdoor AWS Summit San Francisco 2017 Keynoteより

ビルドとデプロイにかかる時間を約10倍に改善できた

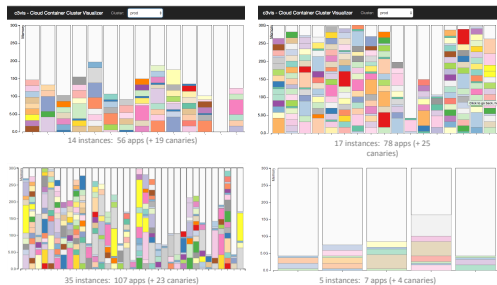


<https://www.youtube.com/watch?v=fqXa0RKJvJE>

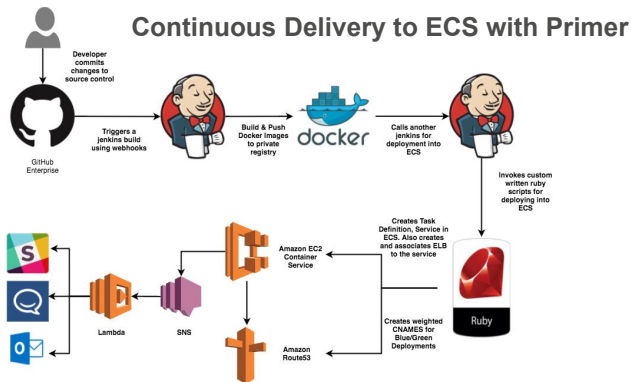
事例：Expedia

世界最大級の旅行会社

ECS Production Clusters – Serving 200 applications



Charts produced with c3vis: github.com/ExpediaDotCom/c3vis



- ❑ Primer - 内部のデプロイツール
 - ❖ 様々なアプリのテンプレートを提供
 - ❖ GitHubレポジトリ作成からパイプライン、監視までワンクリックでサービス作成が可能
- ❑ ECS Optimized AMIをベースにCloudFormationを使って設定
- ❑ ECSと他AWSサービスを統合し、無停止でインスタンスの入替えを実現

たくさんのお客様が、本番環境でECSをご利用中

～スタートアップからエンタープライズまで～

airtime

Atlassian

Aol.



Blackboard

edmunds.com

Expedia



FOURSQUARE

GILT

here

instacart

lyft

meetup

mytaxi

okta



RIOT
GAMES

slack

SUP
ERC
ELL

Trimble

Upserve

ECS事例セッション(1/2) AWS Summit Tokyo 2017

6/1 15:20～ [Amazon ECS と SpotFleet を活用した低コストでスケラブルなジョブワーカーシステム](#)

6/1 17:20～ [\[Intelligence\] オンプレから移行するので、Amazon ECS でコンテナ化と Terraform でインフラコード化した話](#)

6/2 14:20～ [DMM における会員基盤プラットフォームへのAWS導入から活用事例の紹介](#)

ECS事例セッション(2/2) AWS Summit Tokyo 2017

[ナビタイムサービスにおける、Amazon ECS を活用したシステム移行 ～『乗換NAVITIME』での移行事例 ～](#)

[\[リコー\] サービス全断はダメ、ゼツタイ。途切れないテレビ会議システムを目指して ～AWS を最大限活用して可用性を高める秘策～](#)

[\[ABEJA\] IoT / Bigdata / AI 時代におけるスケーラブルな Deep Learning 実行基盤と応用](#)

[Amazon ECS の進化、DevOps と Microservices の実践\(フラー株式会社様事例\)](#)
[クックパッドの機械学習を支える基盤のつくりかた](#)

[\[CyberZ\] OPENREC.tvにおけるライブ動画およびメッセージ配信基盤のリプレース全貌](#)

まとめ

まとめ

- ❑ なぜコンテナなのか？
- ❑ Amazon EC2 Container Service 概要

Amazon EC2 Container Service

- ❑ AWS ネイティブで非常に簡単
- ❑ スタートアップからエンタープライズまで多くの実績

(参考) Black Belt オンラインセミナー資料

Amazon EC2 Container Service

- ❑ Slideshare <http://bit.ly/2qIM9pO>
- ❑ PDF <http://bit.ly/2s247hC>
- ❑ オンデマンドセミナー(動画) <http://bit.ly/2rnJMqM>

本セッションのFeedbackをお願いします

受付でお配りしたアンケートに本セッションの満足度やご感想などをご記入ください
アンケートをご提出いただきました方には、もれなく**素敵なAWSオリジナルグッズ**を
プレゼントさせていただきます



アンケートは受付、パミール3FのEXPO展示会場内にて回収させていただきます

AWS

S U M M I T

Thank you !

