

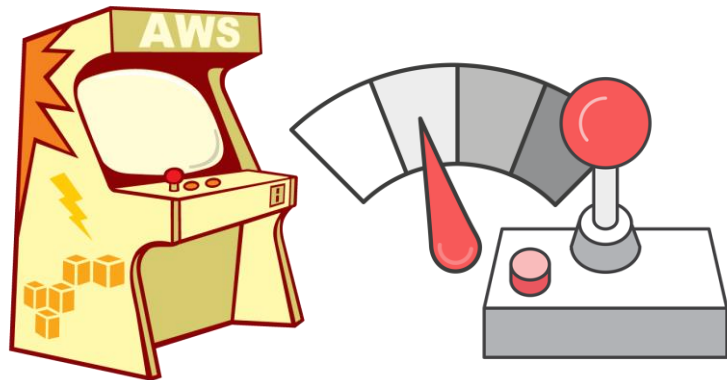


AWS  
**Black Belt**  
Online Seminar

# 【AWS Black Belt Online Seminar】

## AWS 上でのスピードと高可用性を両立した ゲームインフラの構築と事例

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 森 祐孝  
2016.12.20



# 自己紹介

## 森 祐孝（もり ゆうこう）

 ソリューションアーキテクト

 経歴

- メーカー系SIer、電子マネー運営（サーバソフト開発）
- 独立系SIer
  - 情報通信キャリア向けのSIでPL、PM
- コンソール系ゲーム会社（テクニカルディレクター）
  - ブラウザソーシャルゲーム、スマートフォン向けソーシャルゲーム

 担当

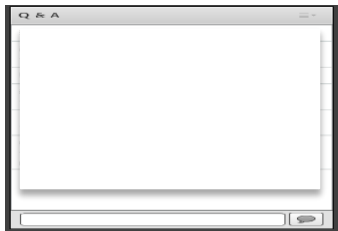
- ソーシャルゲーム、コンソールゲーム系のお客様
- Amazon Lumberyard !

# AWS Black Belt Online Seminar へようこそ！

## 質問を投げることができます！

- Adobe ConnectのQ&Aウィンドウから、質問を書き込んでください。  
(書き込んだ質問は、主催者にしか見えません)
- 今後のロードマップに関するご質問はお答えできませんのでご了承ください。
- Twitterへツイートする際はハッシュタグ **#awsblackbelt** をご利用ください。

①Q&Aウィンドウ  
右下のフォームに  
質問を書き込んで  
ください



②吹き出しマークで  
送信してください

# AWS Black Belt Online Seminar とは

AWSJのTechメンバがAWSに関する様々な事を紹介するオンラインセミナーです

## 【火曜 12:00~13:00】

主にAWSのソリューションや  
業界カッタでの使いどころなどを紹介  
(例: IoT、金融業界向け etc.)

## 【水曜 18:00~19:00】

主にAWSサービスの紹介や  
アップデートの解説  
(例: EC2、RDS、Lambda etc.)



※最新の情報は下記をご確認下さい。

オンラインセミナーのスケジュール&申し込みサイト

- <http://aws.amazon.com/jp/about-aws/events/#webinar>



# Amazon Lumberyard Engine



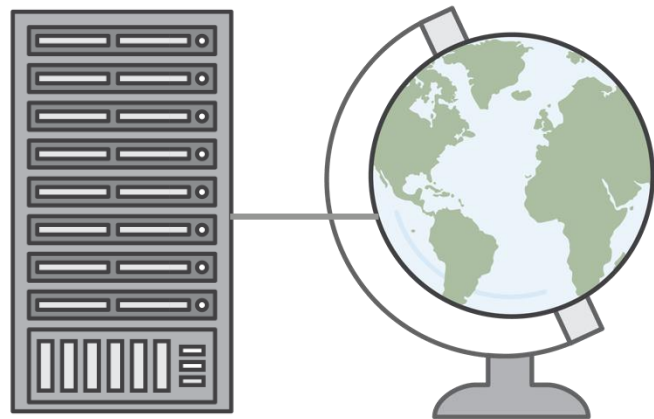
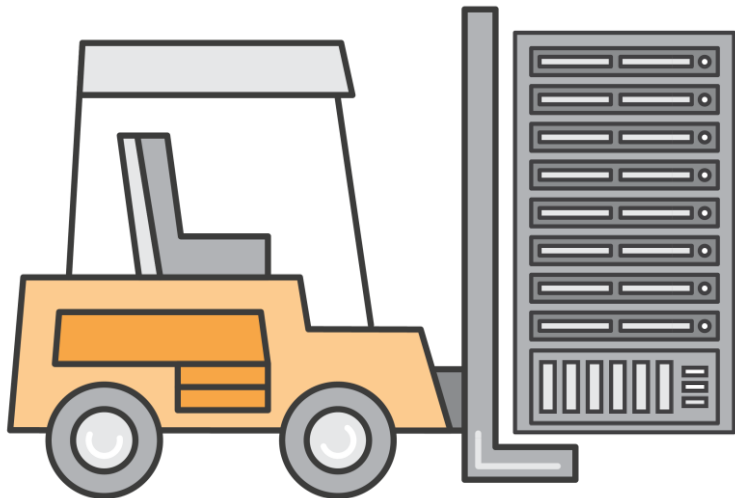
無料なので使ってください！

lumberyard  
BY amazon



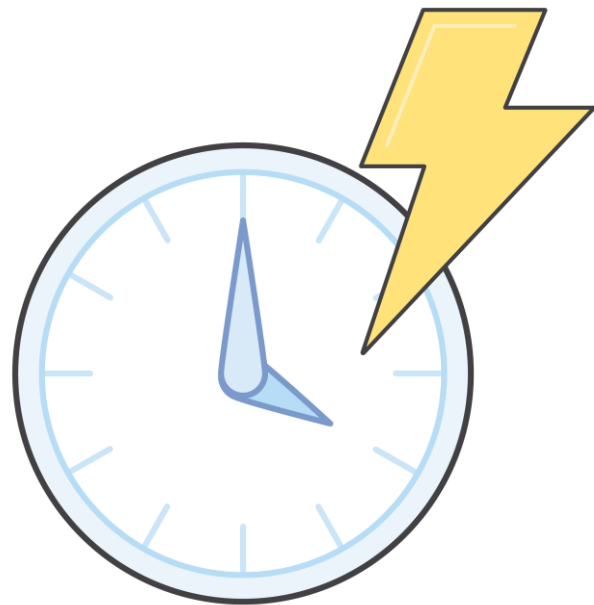
なぜゲームインフラでAWSを選ぶのでしょうか？

# 重いものを持ち上げることはビジネスの中核ではありません



# Game Life Cycle – 高速

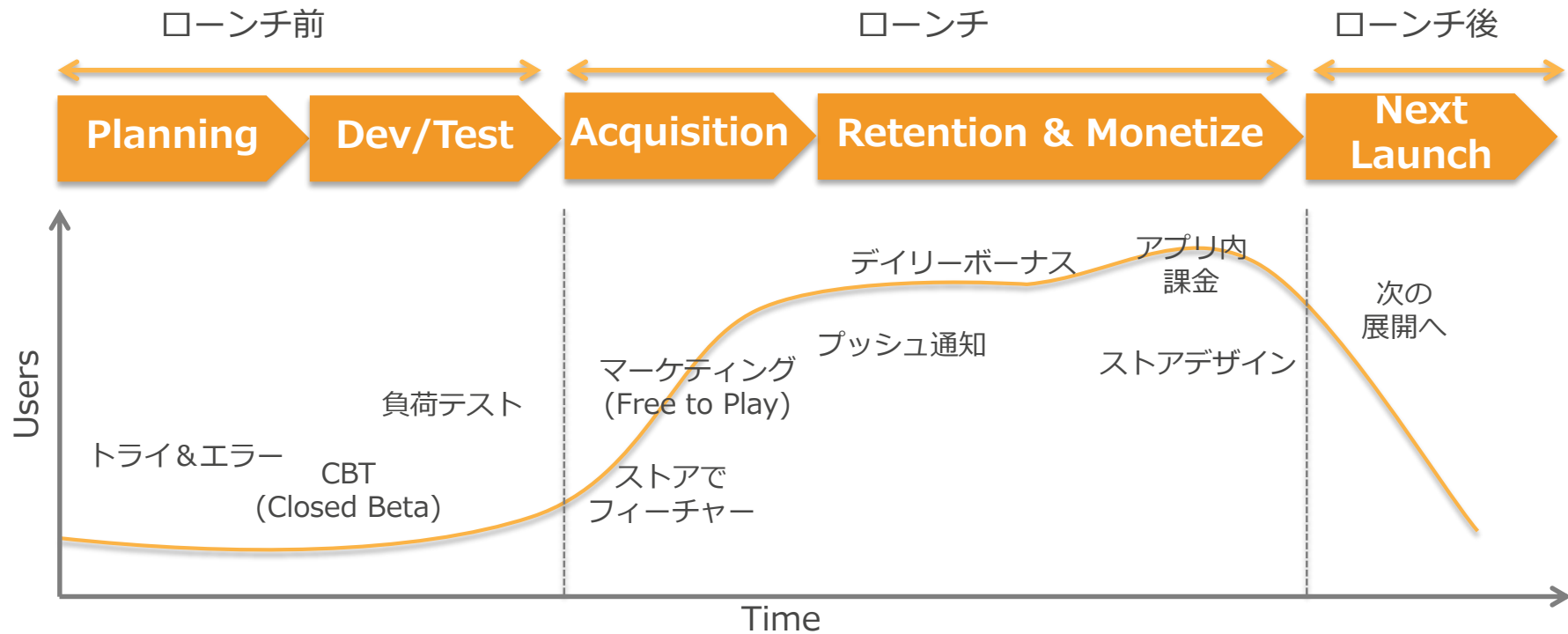
- ゲーム業界は日々高速に進歩しています
- アジャイル開発
- 数時間、数日ではなく数分単位でキャパシティとサービスが変動



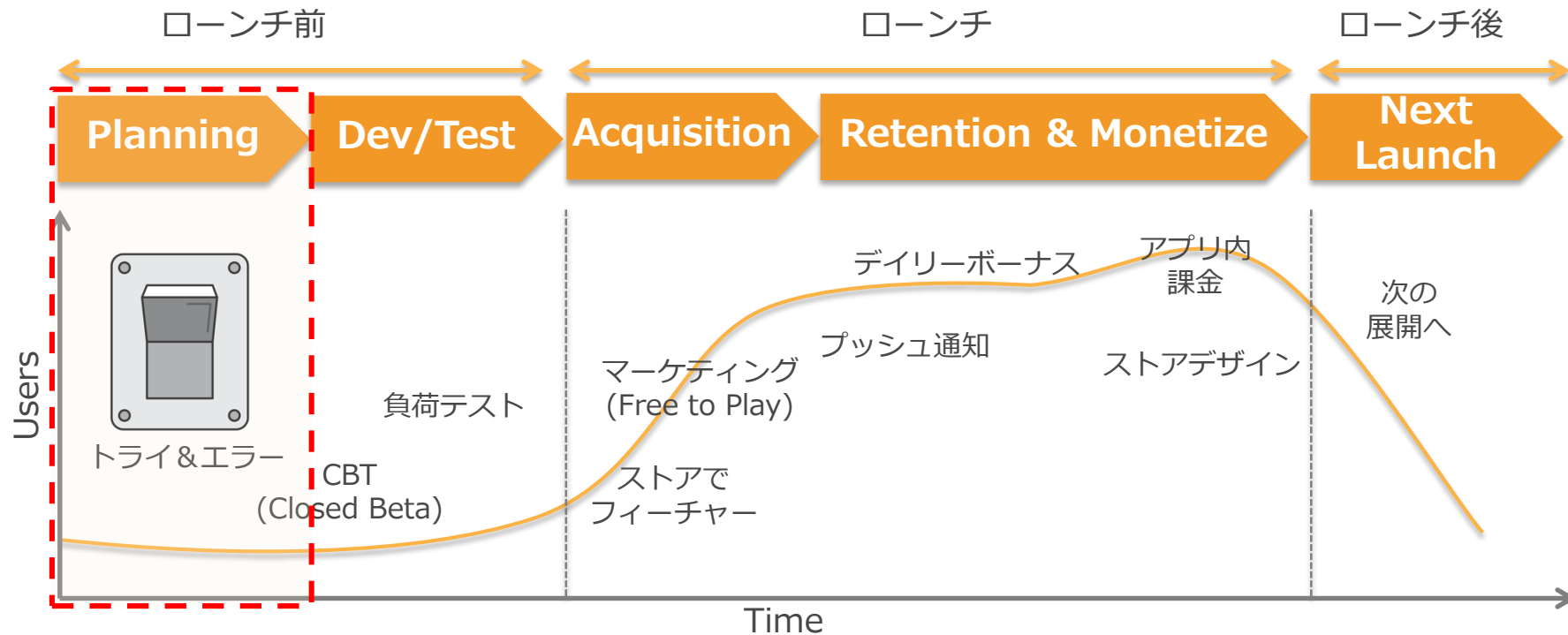


## Game Life Cycle – AWSはどこをお手伝いできますか？

# Game Life Cycle



# Game Life Cycle

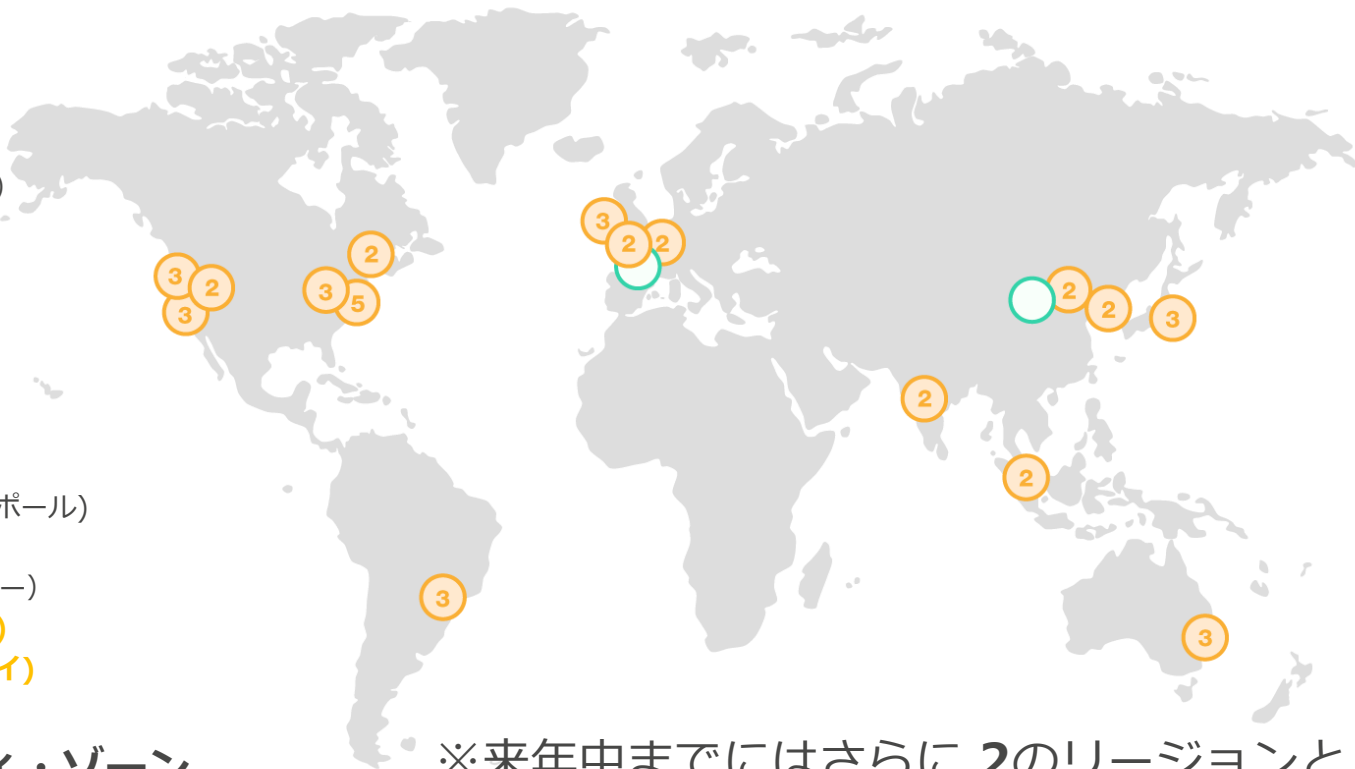




# ゲーム配信を世界中のリージョンがから選択可能

## 16のリージョン

1. 米国東部 (バージニア北部)
2. **米国東部(オハイオ)**
3. 米国西部 (北カリフォルニア)
4. 米国西部 (オレゴン)
5. AWS GovCloud (米国)
6. **カナダ(中部)**
7. 南米 (サンパウロ)
8. 欧州 (アイルランド)
9. 欧州 (フランクフルト)
10. **欧州 (ロンドン)**
11. アジアパシフィック (シンガポール)
12. **アジアパシフィック (東京)**
13. アジアパシフィック (シドニー)
14. **アジアパシフィック (ソウル)**
15. **アジアパシフィック (ムンバイ)**
16. 中国 (北京)



42のアベイラビリティ・ゾーン  
60以上のエッジロケーション

※来年中までにはさらに **2**のリージョンと  
**4**のアベイラビリティゾーン予定

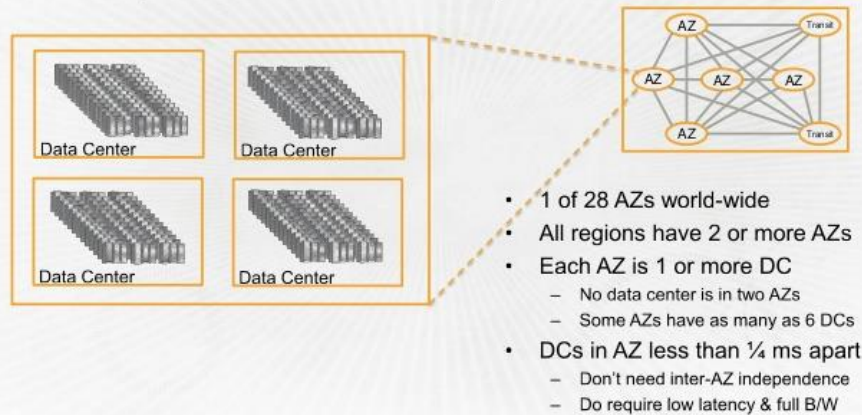
# リージョンとAvailability Zone (AZ)

各リージョン内に複数の独立したAZ(データセンター群)が存在

各AZは互いに影響を受けないように、地理的・電源的・ネットワーク的に分離するように設計されている

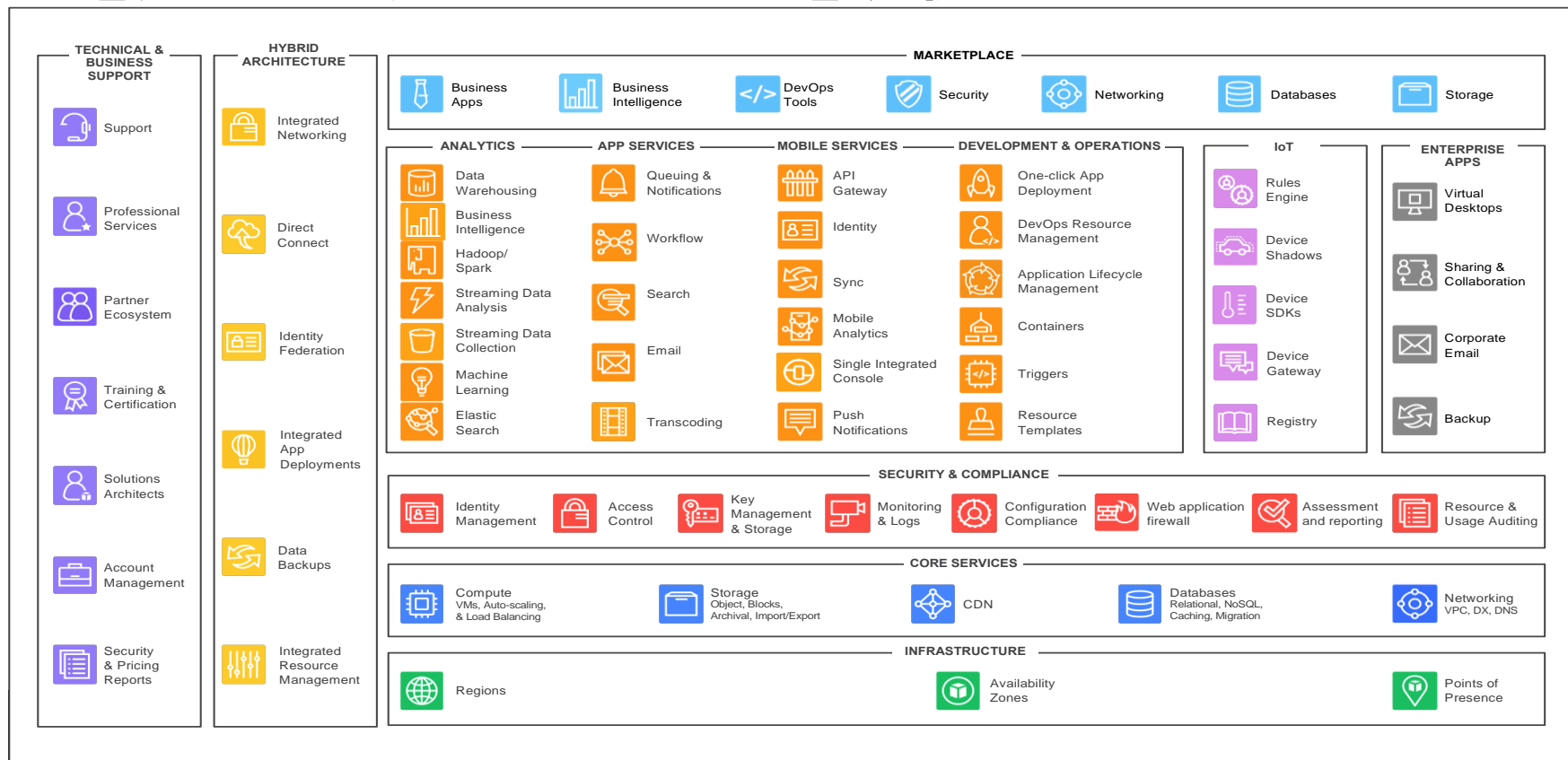
AZ間は低レイテンシーの高速専用線で接続されている

## Example AWS Availability Zone



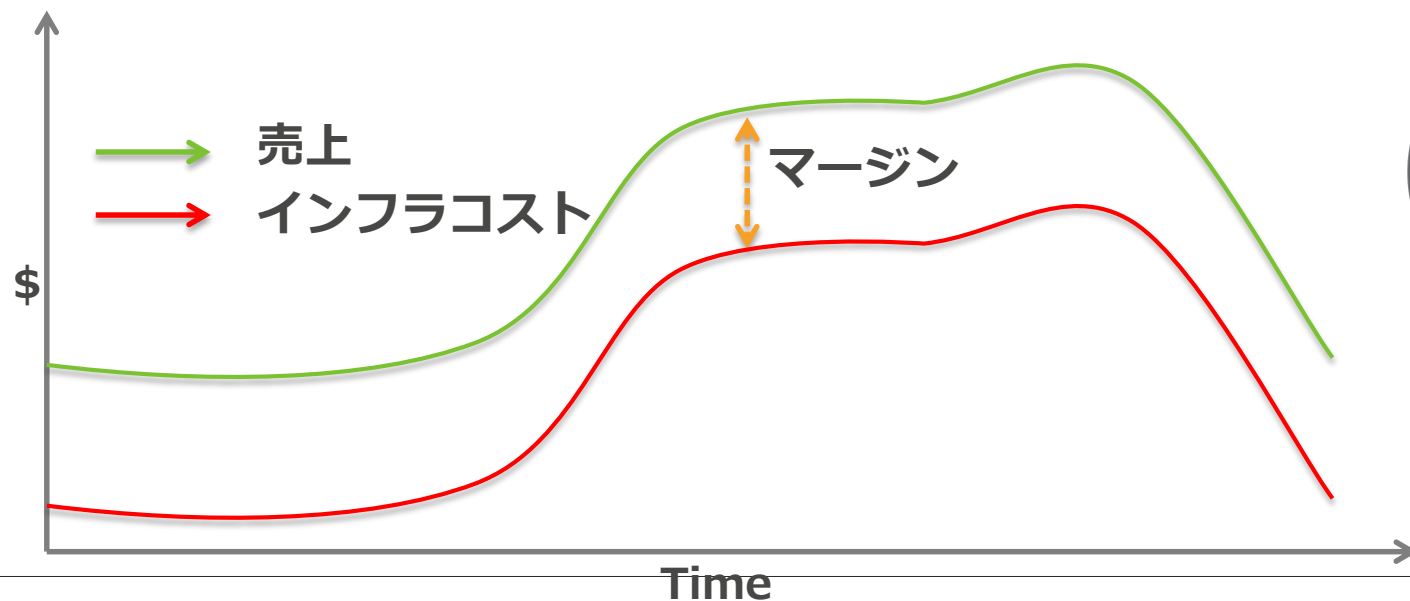
<http://www.slideshare.net/AmazonWebServices/spot301-aws-innovation-at-scale-aws-reinvent-2014>

# 70を超えるクラウドサービスを提供

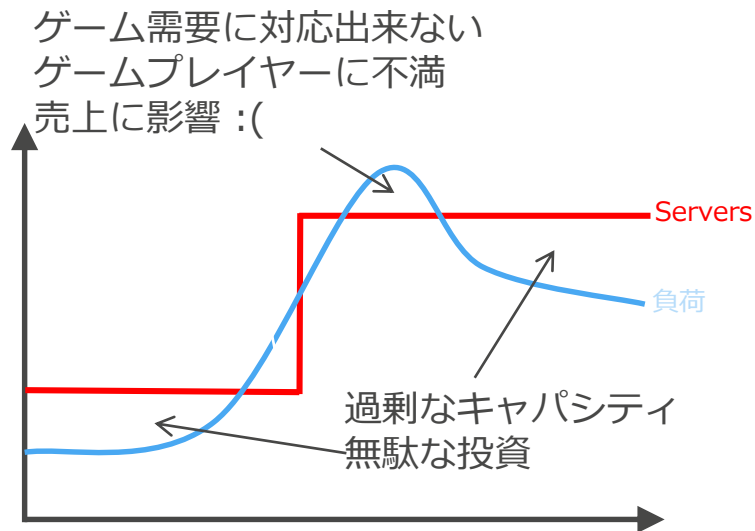


# 実際の利用分だけのコスト

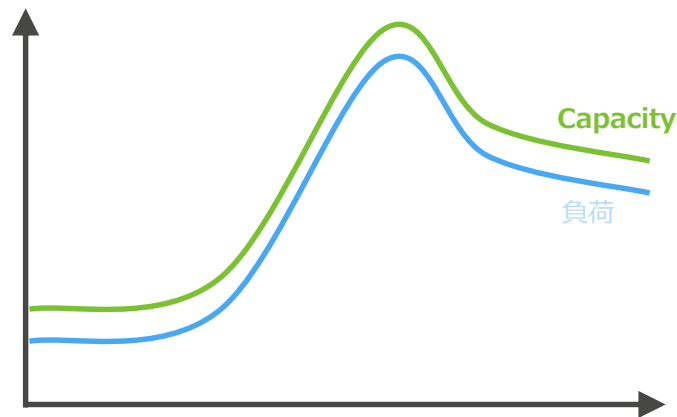
インフラコストと収益モデルを揃える



# ゲームユーザーの需要に合わせてサーバをスケール

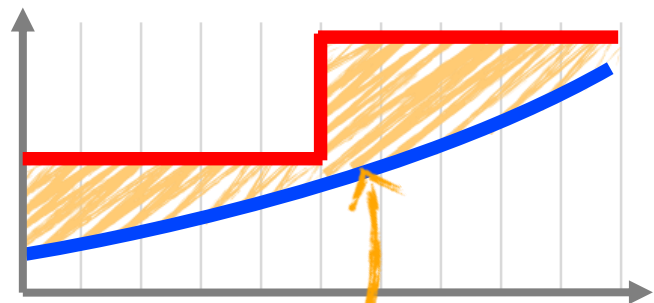


Traditional Rigid(柔軟ではない)

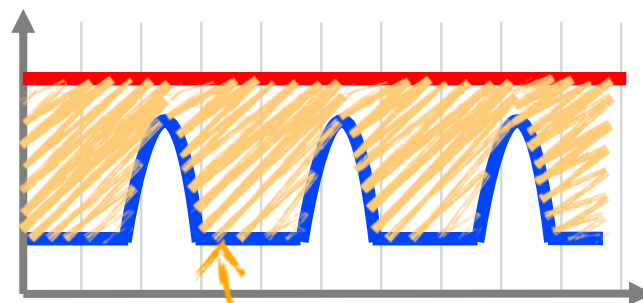


AWS Elastic(柔軟)

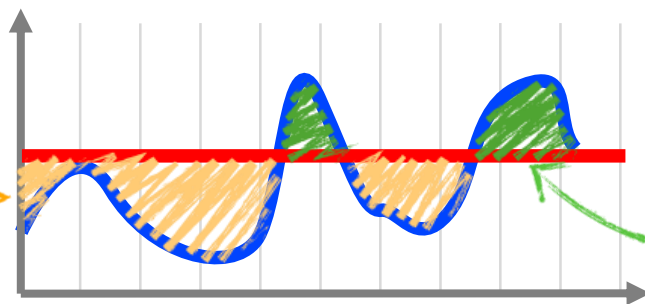
# ゲームの成長でコントロール



無駄！

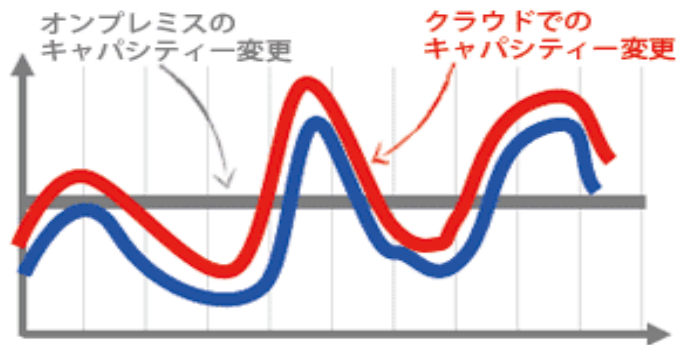
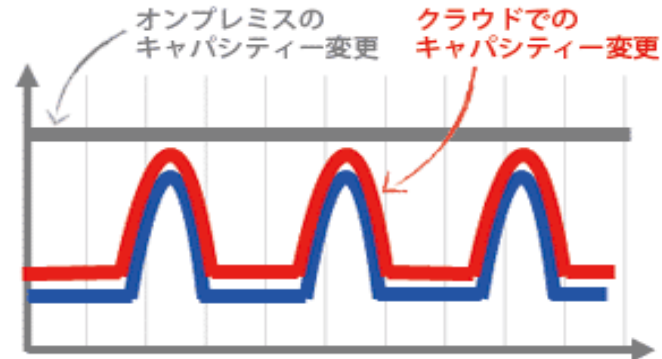
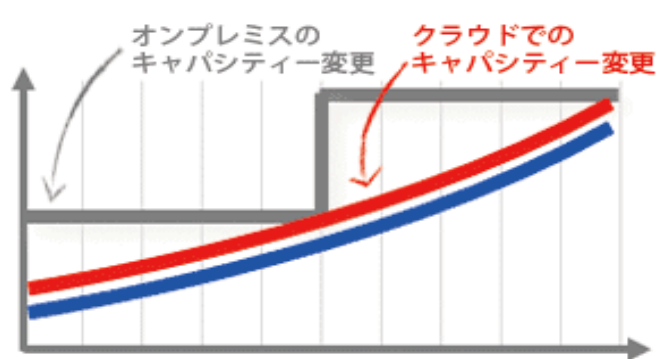


無駄！

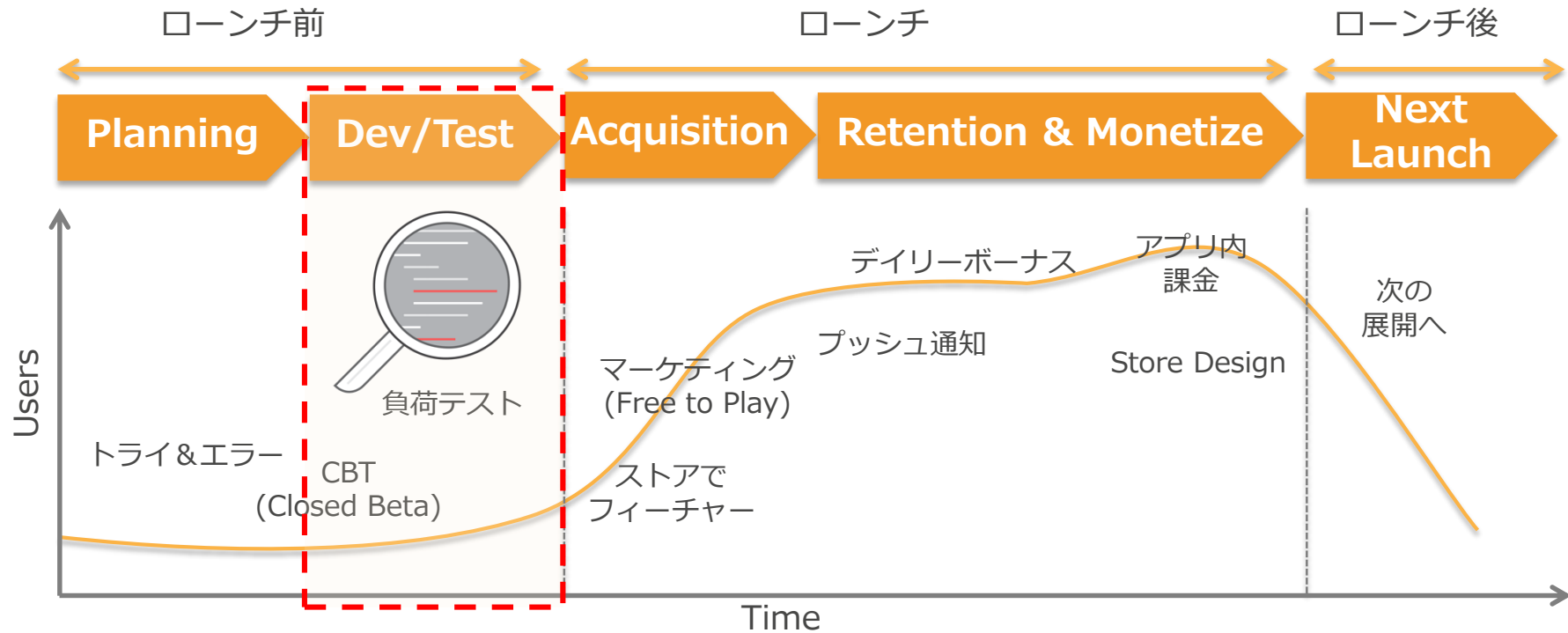


機会損失！

# ゲームの成長でコントロール



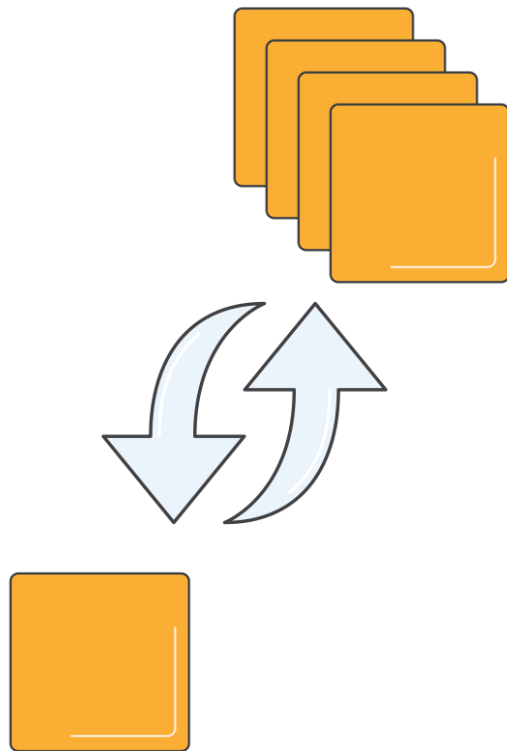
# Game Life Cycle



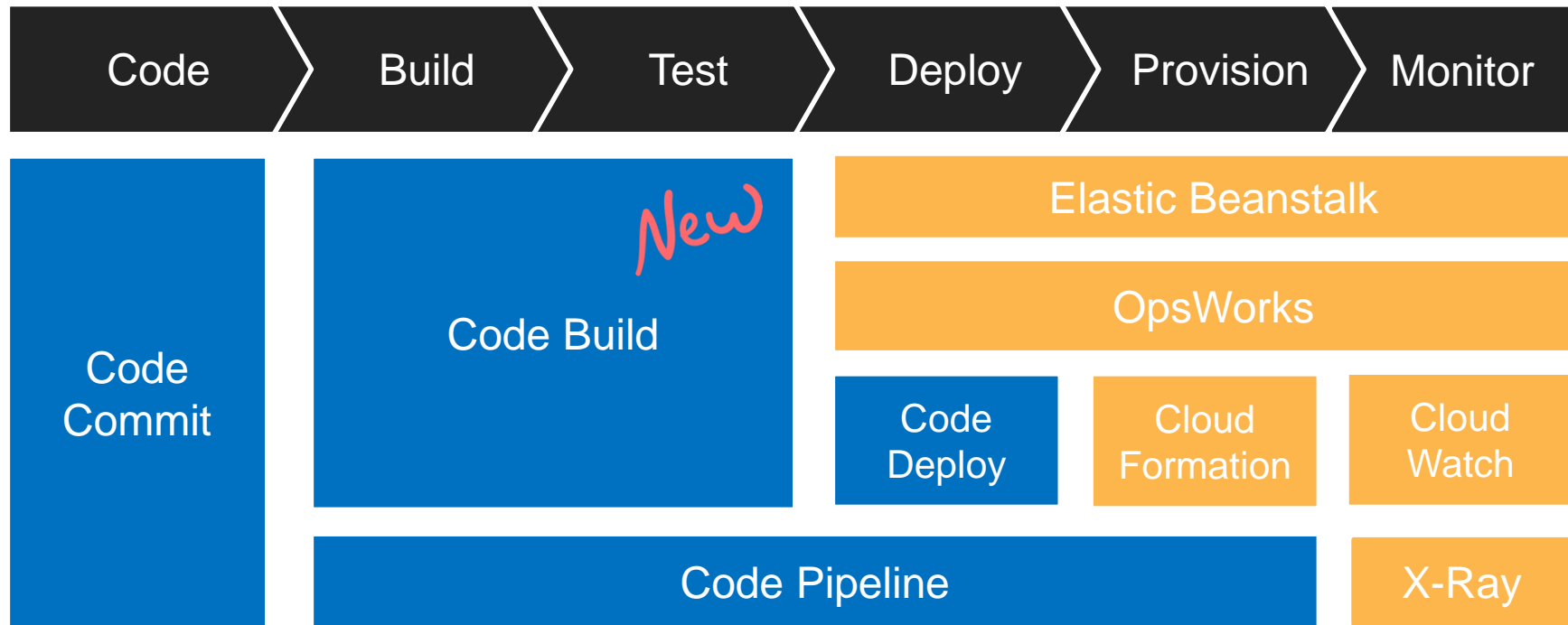


# 開発とテスト

- アジャイルリソースー いつでも必要なサービスやキャパシティを確保
- 変更可能なインフラストラクチャ
- 数分以内に必要なリソースを作成、削除可能
  - テスト環境、負荷環境



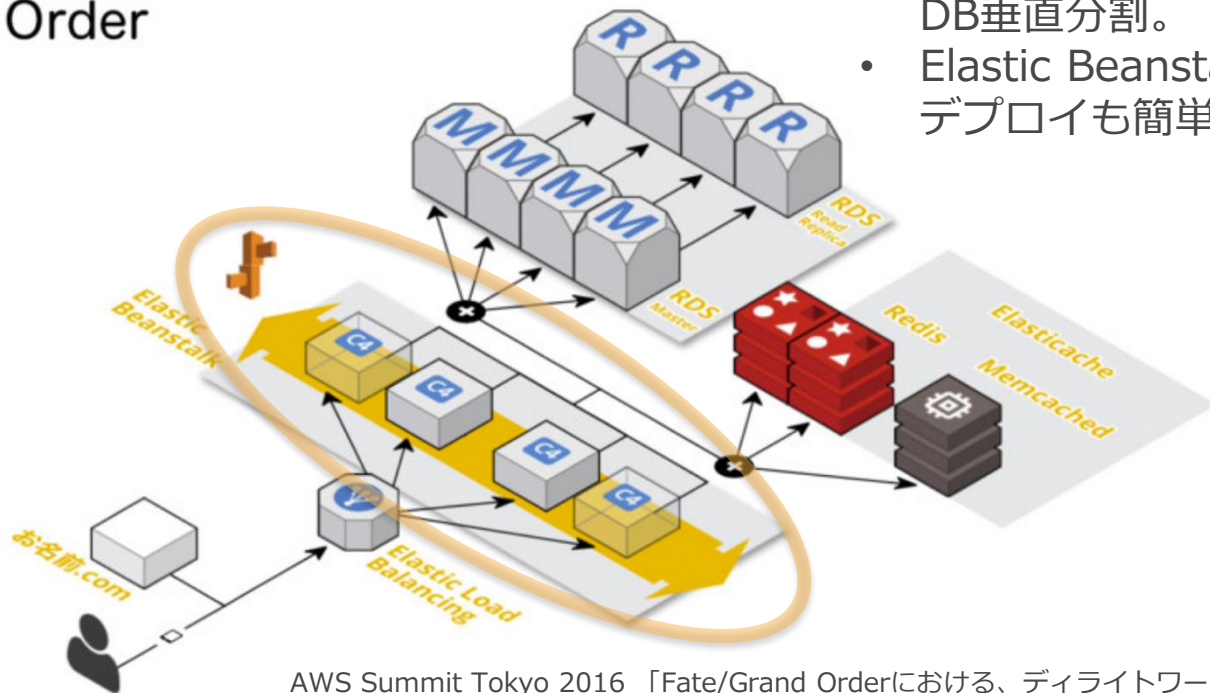
# AWS から提供されているデプロイ & マネージメント 関連サービス



# AWS 導入事例:ディライトワークス株式会社

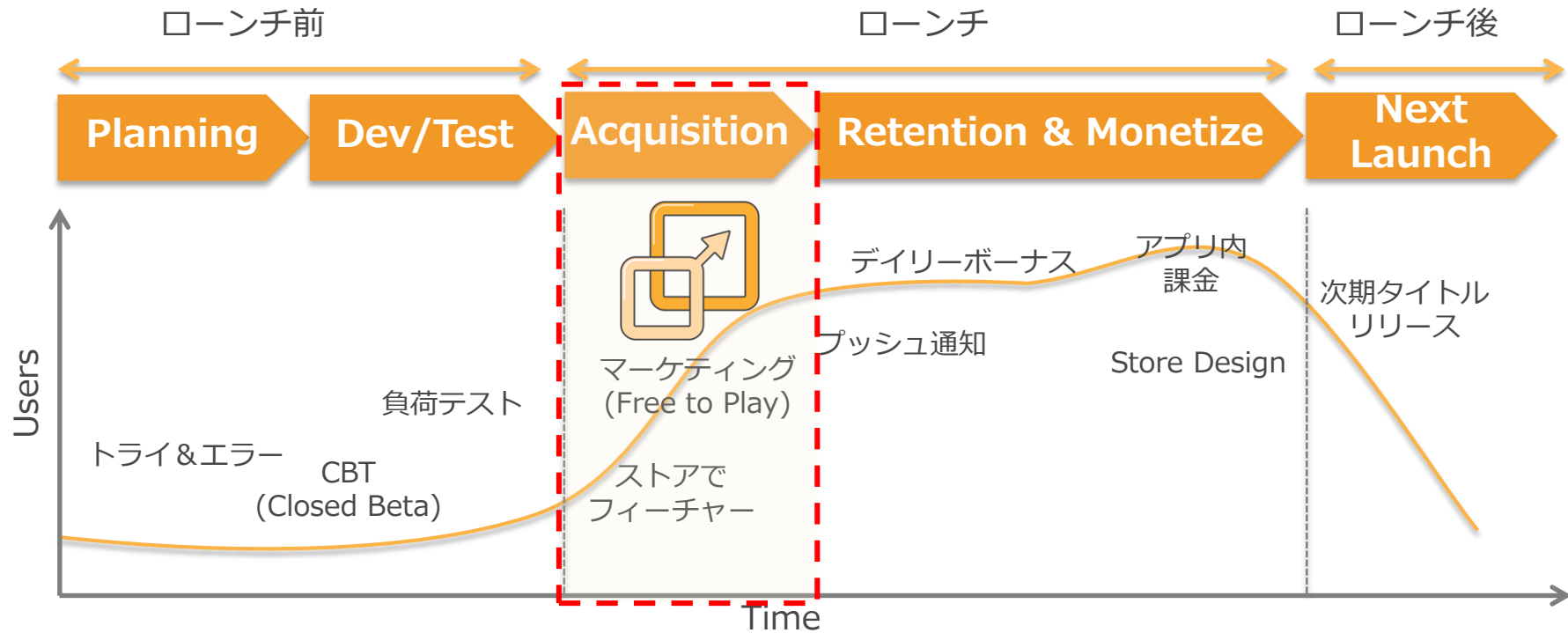
Architecture:  
Fate/Grand Order

- TopゲームのFate/GOでの活用
- RDS for Auroraを活用し、DB垂直分割。
- Elastic Beanstalkでゲームサーバのデプロイも簡単に実現



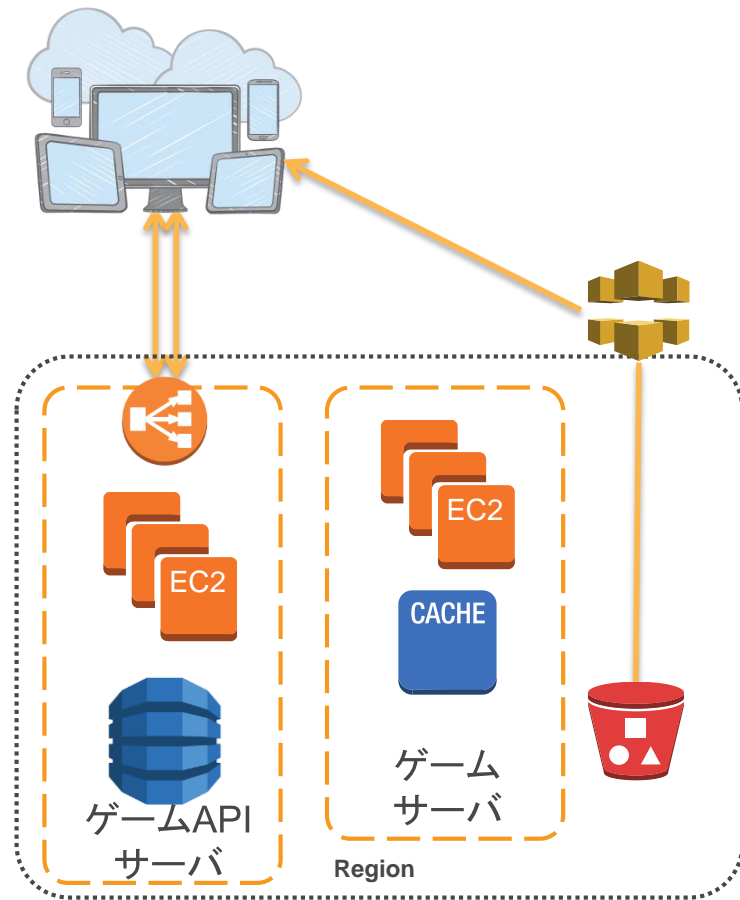
AWS Summit Tokyo 2016 「Fate/Grand Orderにおける、ディライトワークス流 AWS 導入&活用術」より  
<https://speakerdeck.com/isoparametric/grand-orderniokeru-deiraitowakusuluu-aws-dao-ru-and-huo-yong-shu>

# Game Life Cycle



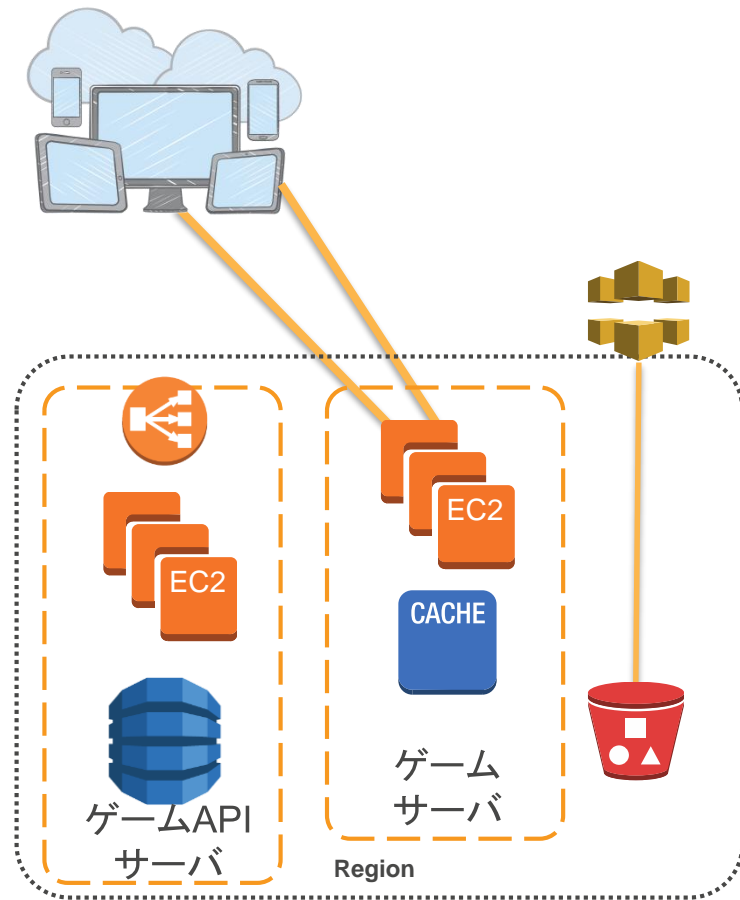
# オンラインゲームのフロー

- ① HTTP APIを用いてログイン
- ② Game Assetsをダウンロード
- ③ Game Serverのマッチメイク



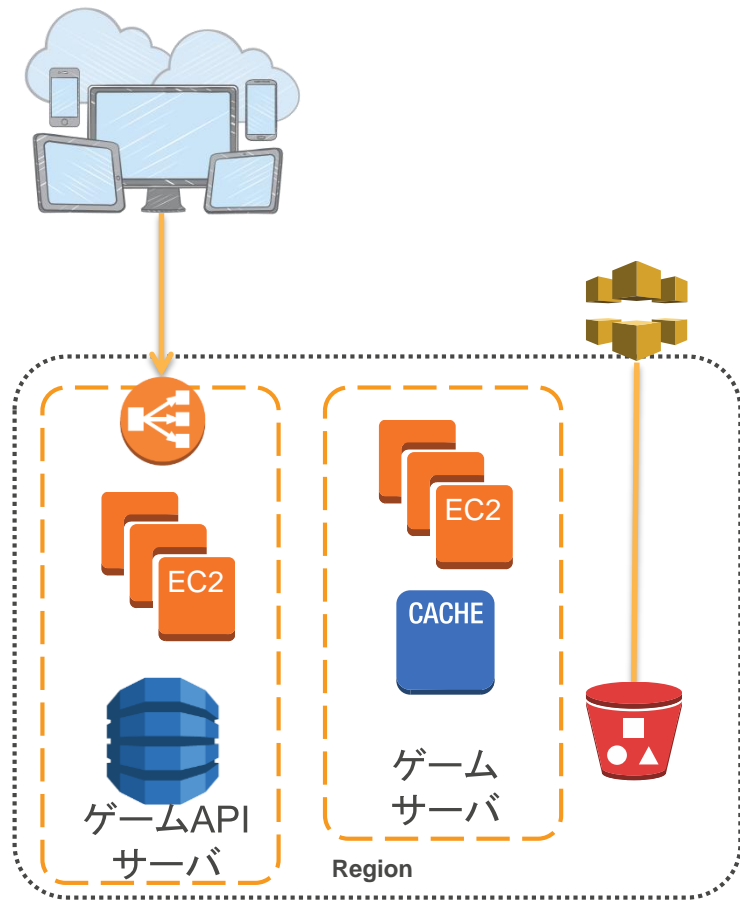
# オンラインゲームのフロー

- ① HTTP APIを用いてログイン
- ② Game Assetsをダウンロード
- ③ Game Serverのマッチメイク
- ④ Serverへ接続（TCP, UDP, WebSocket等）
- ⑤ 離れた仲間と対戦
- ⑥ Game Over



# オンラインゲームのフロー

- ① HTTP APIを用いてログイン
- ② Game Assetsをダウンロード
- ③ Game Serverのマッチメイク
- ④ Serverへ接続（TCP, UDP, WebSocket等）
- ⑤ 離れた仲間と対戦
- ⑥ Game Over
- ⑦ HTTP APIを使って結果の書き込み

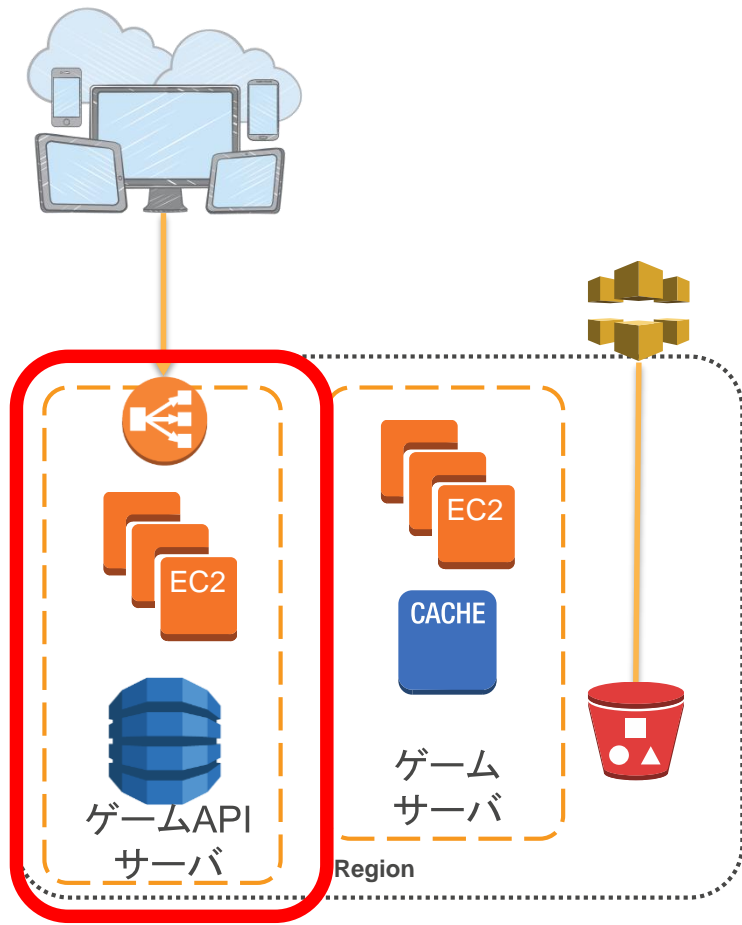


# ゲームAPI サーバ



# オンラインゲームのフロー

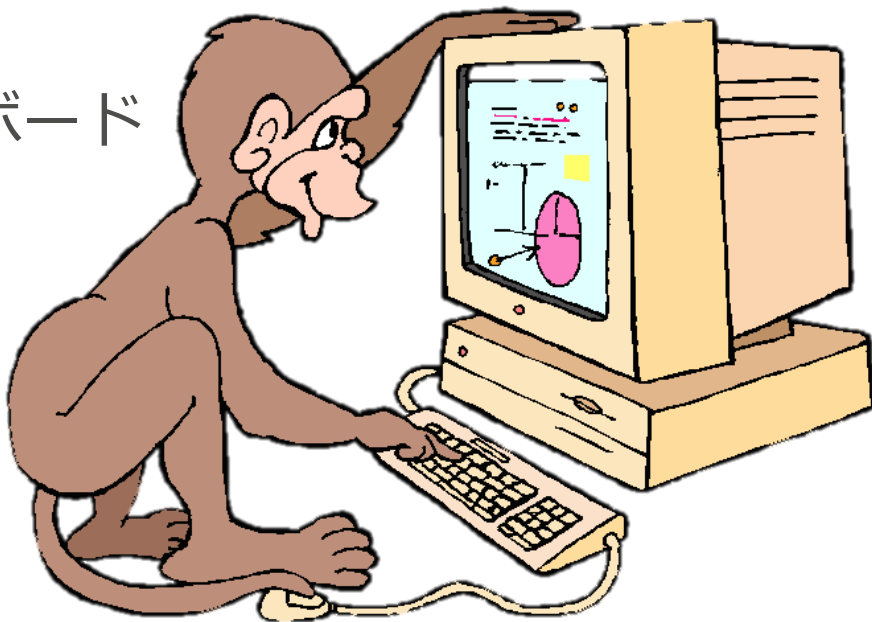
- ① HTTP APIを用いてログイン
- ② Game Assetsをダウンロード
- ③ Game Serverのマッチメイク
- ④ Serverへ接続（TCP, UDP, WebSocket等）
- ⑤ 離れた仲間と対戦
- ⑥ Game Over
- ⑦ HTTP APIを使って結果の書き込み



# 一般的なゲームAPIのコンセプト

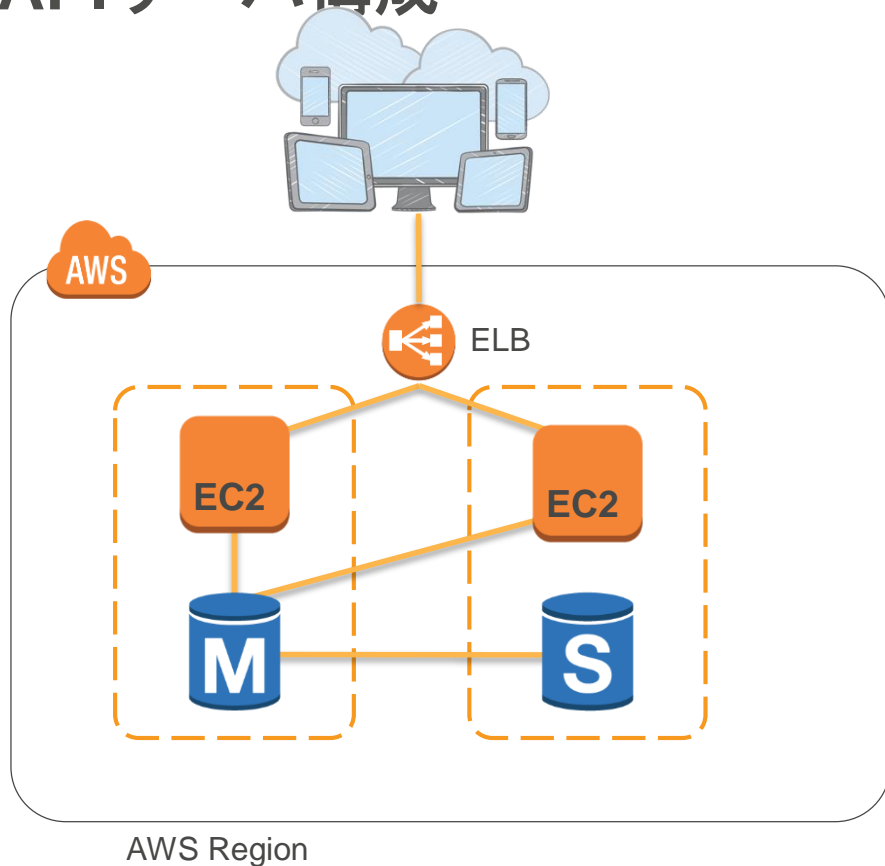
ゲームから見てAPIとして

- HTTP + JSON
- フレンドリスト, スコアボード
- バイナリアセット配信
- 高可用性
- 拡張性



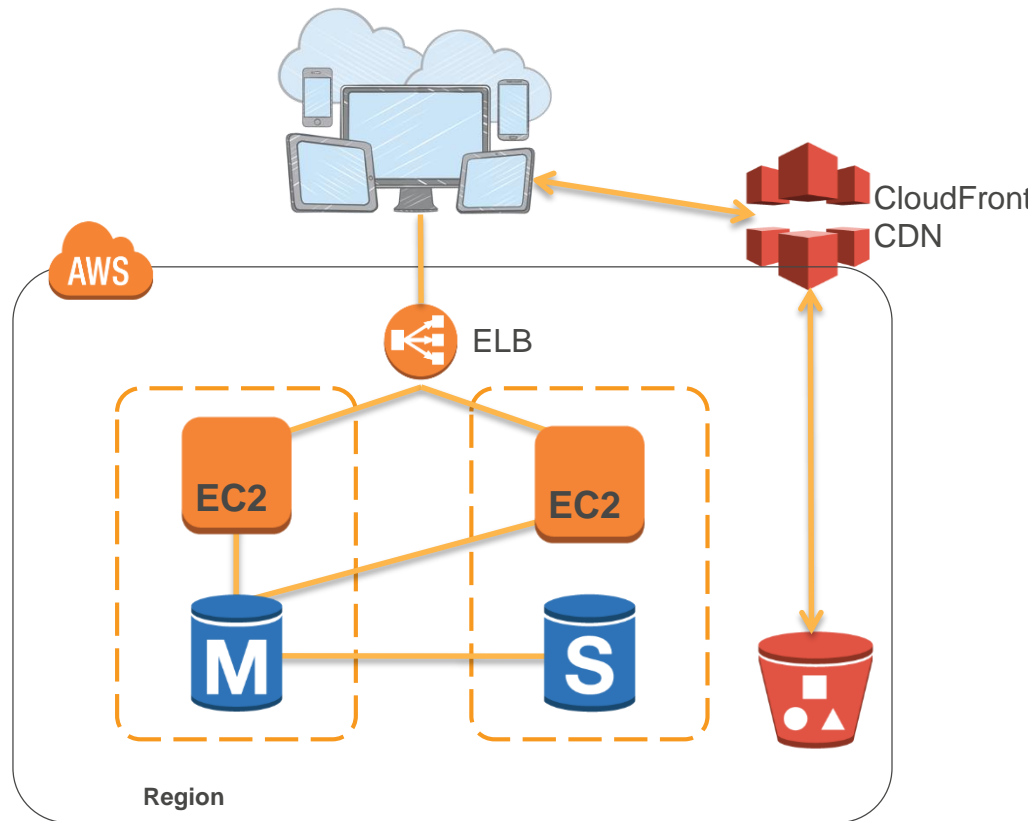
# 一般的なAWS上でのゲームAPIサーバ構成

- WEB + DBゲームサーバ
  - ユーザログイン
  - ロビーサーバ
  - スコアボード
  - プレゼントリスト
- 最適なリージョンの選択
- 2つ以上のAZで構築
- アプリ用Amazon EC2
- Elastic Load Balancing
- Amazon RDS
  - Multi-AZ



# 静的コンテンツの配信

- Amazon S3 + CloudFront
  - ゲームアセットファイル
  - ユーザフォト
  - カスタムマップ
- Amazon CloudFront(CDN)

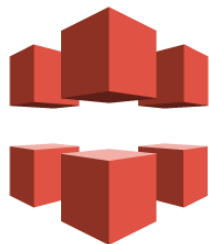


# Amazon S3 & Amazon CloudFront

- Amazon S3
  - 高可用性**オブジェクトストレージ**
  - **99.999999999%** の耐久性に設計
  - オブジェクトを**HTTP / HTTPS** のエンドポイントで取得
- Amazon CloudFront
  - 簡単にサイトの**高速化**が実現できると共に、サーバの負荷も軽減
  - 様々な規模のアクセスを処理することが可能
  - 世界60箇所以上の**エッジロケーション**



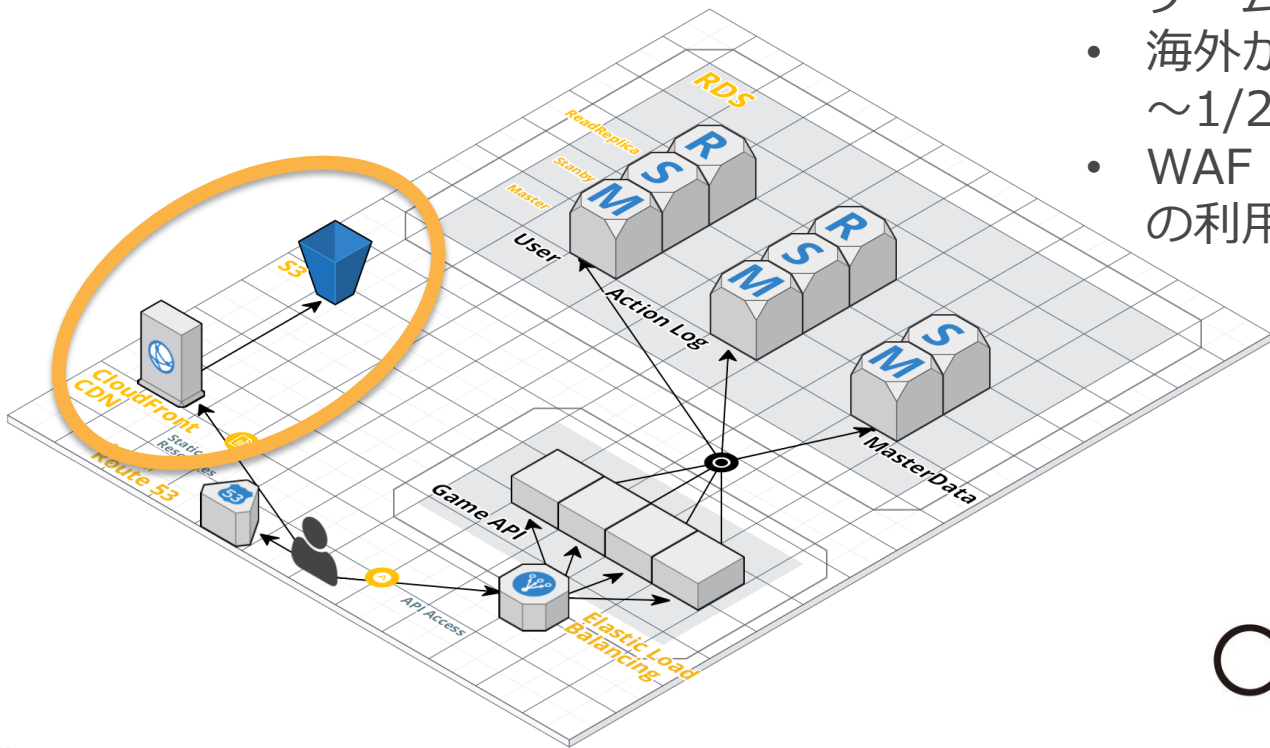
Amazon S3



Amazon CloudFront

# AWS 導入事例:株式会社サイバード (BFB 2016)

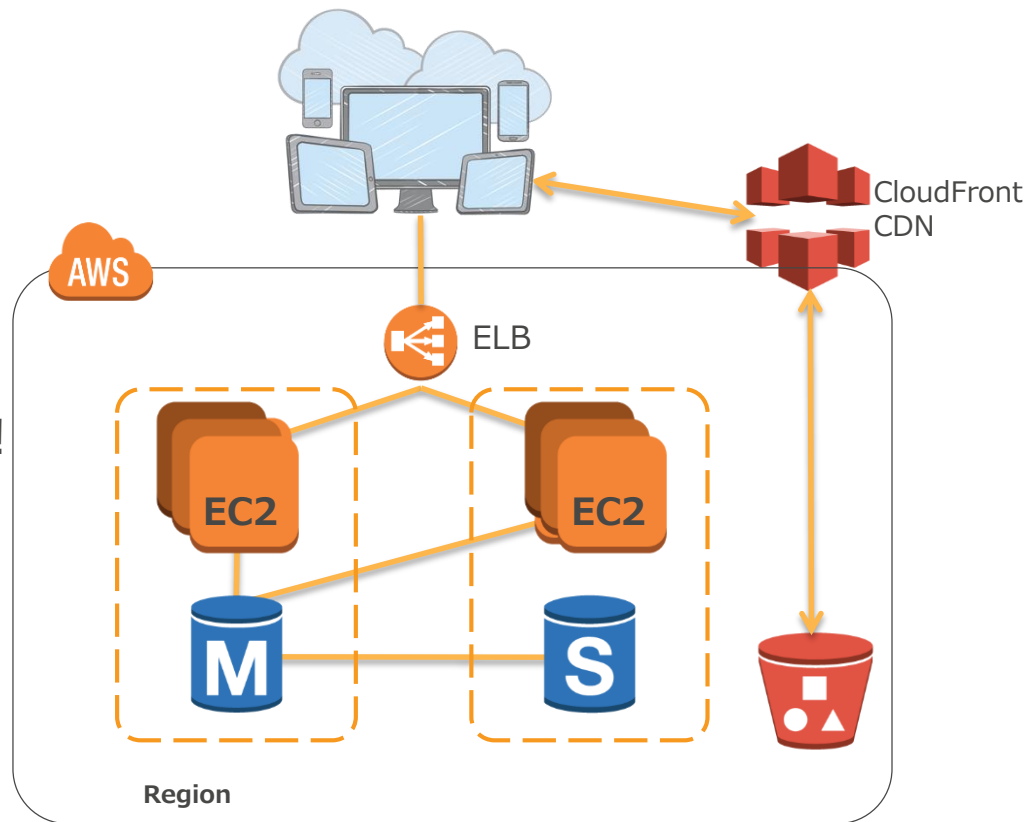
- ゲームのアセットファイルを配信
- 海外からのアクセスでも従来の1/3  
～1/2ダウンロード時間短縮
- WAF (web application firewall)  
の利用も

C  B I R D<sup>®</sup>

モバイルでスマイル!

# ここからスケール

- Amazon S3 + CloudFront
  - ゲームアセットファイル
  - ユーザフォト
  - カスタムマップ
  - Amazon CloudFrontを追加！
- Auto Scaling group
  - 動的キャパシティ
  - ユーザの増加に合わせて
  - オートヒーリング

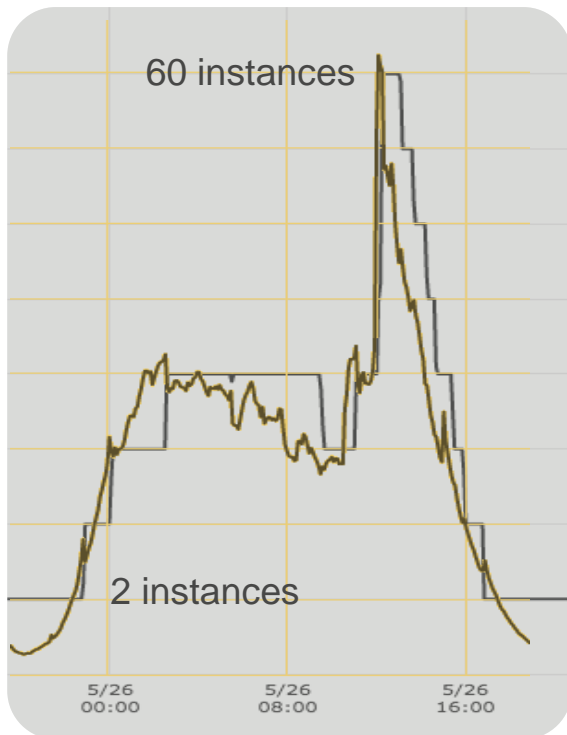


# Adjust dynamically - 쿠키러ん



DEV SISTERS™

Case Study



# instances vs. CPU utilization

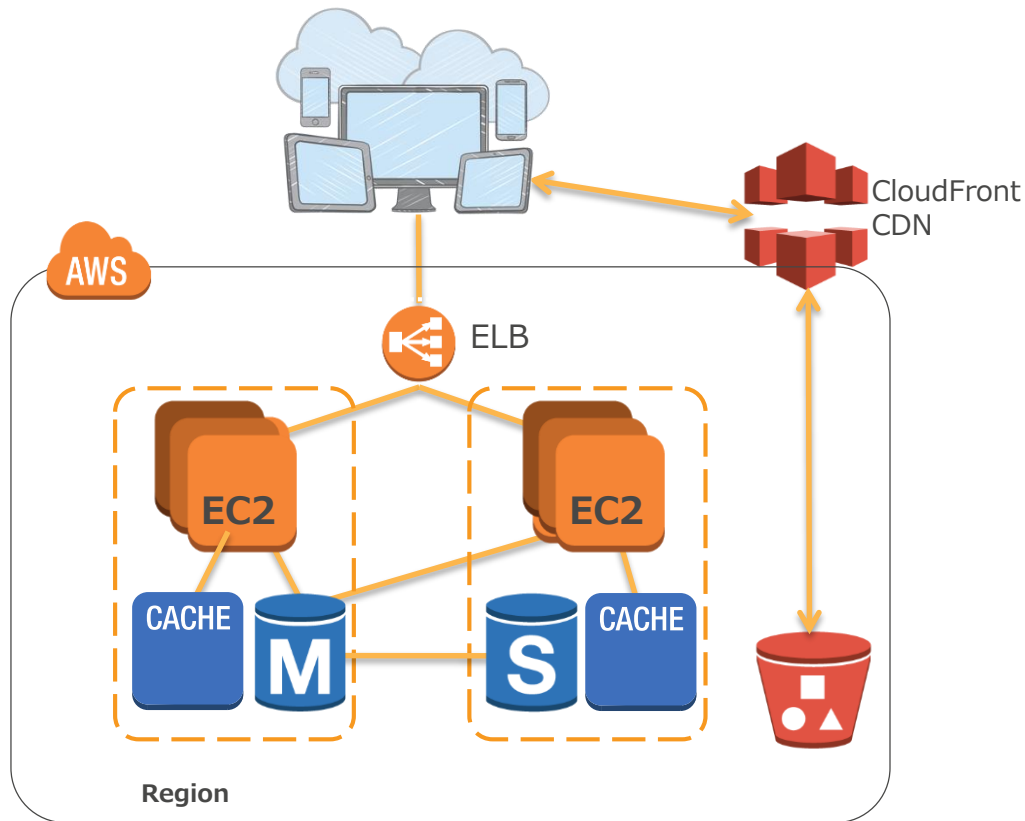


24時間の間にサービスの変化に基づいて、2~60インスタンスをオートスケーリングさせている



# DBリードが増加！

- Amazon S3 + CloudFront
  - Game Assets
  - ユーザフォト
  - Custom Maps
- Auto Scaling group
  - 動的キャパシティ
  - ユーザの増加に合わせて
  - オートヒーリング
- Amazon ElastiCache
  - フルマネージド
  - Memcached
  - Redis

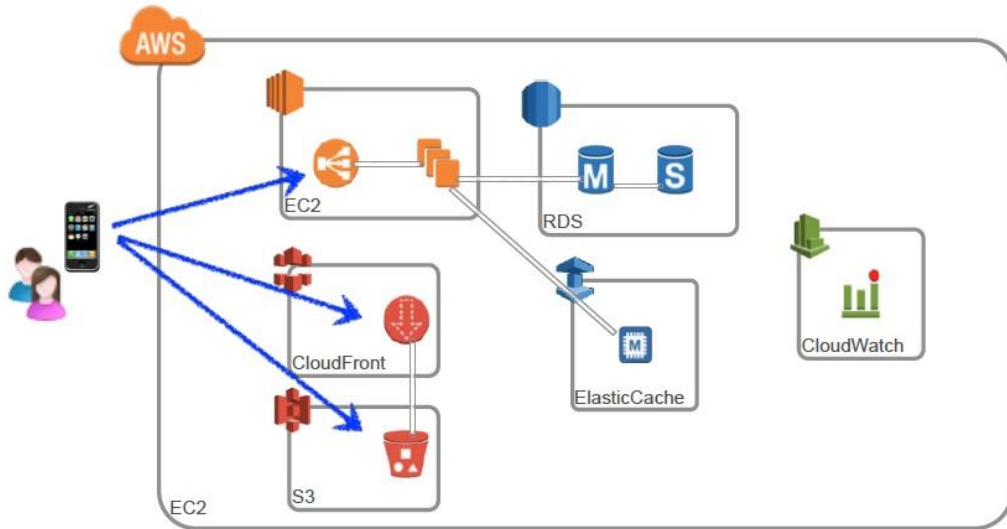


# パズドラも基本的には同じような構成

Case Study

Case Study

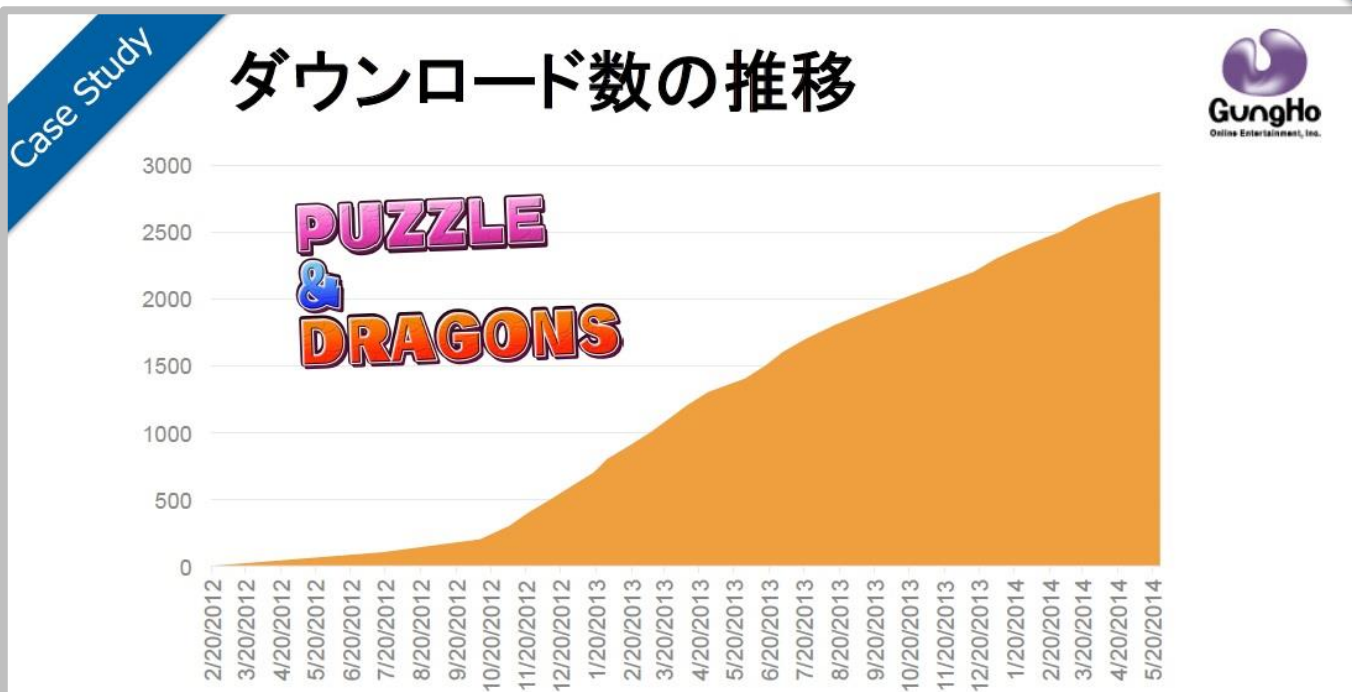
## サービス開始時の構成



Tokyo Game Show 2014の発表資料より抜粋

# 成長に伴い...

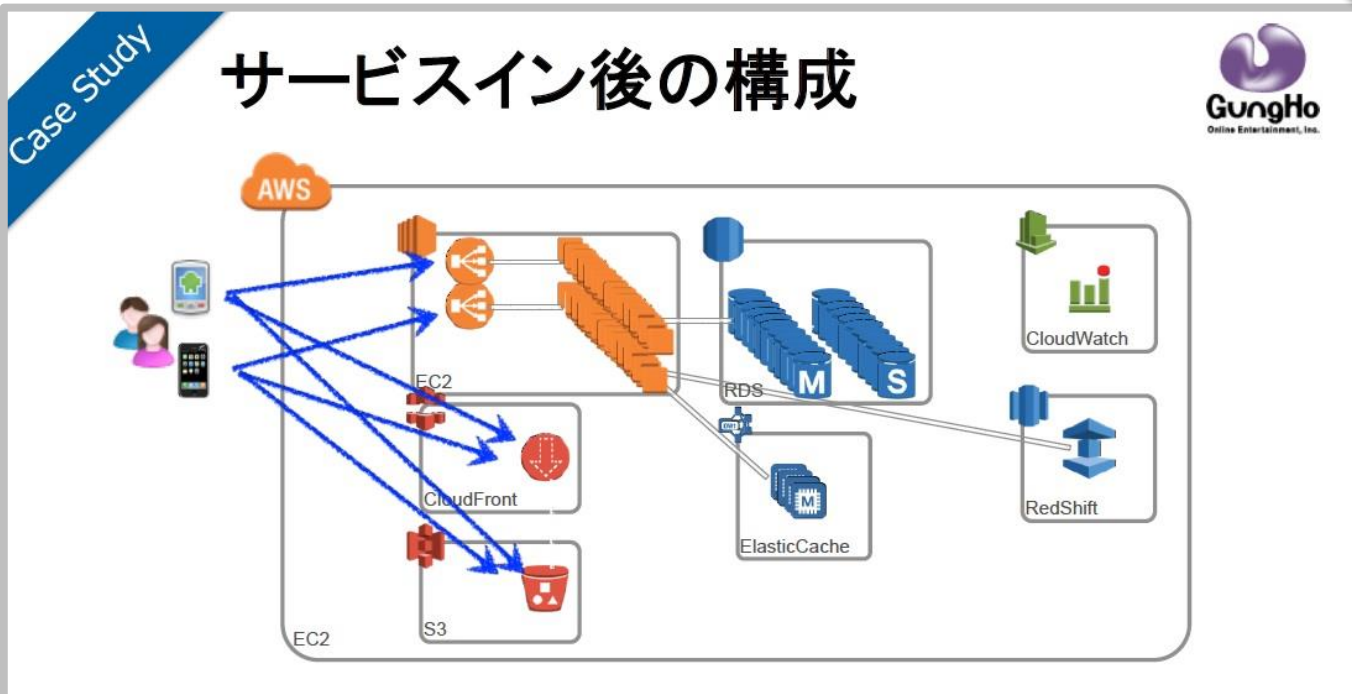
Case Study



Tokyo Game Show 2014の発表資料より抜粋

# サーバ増！

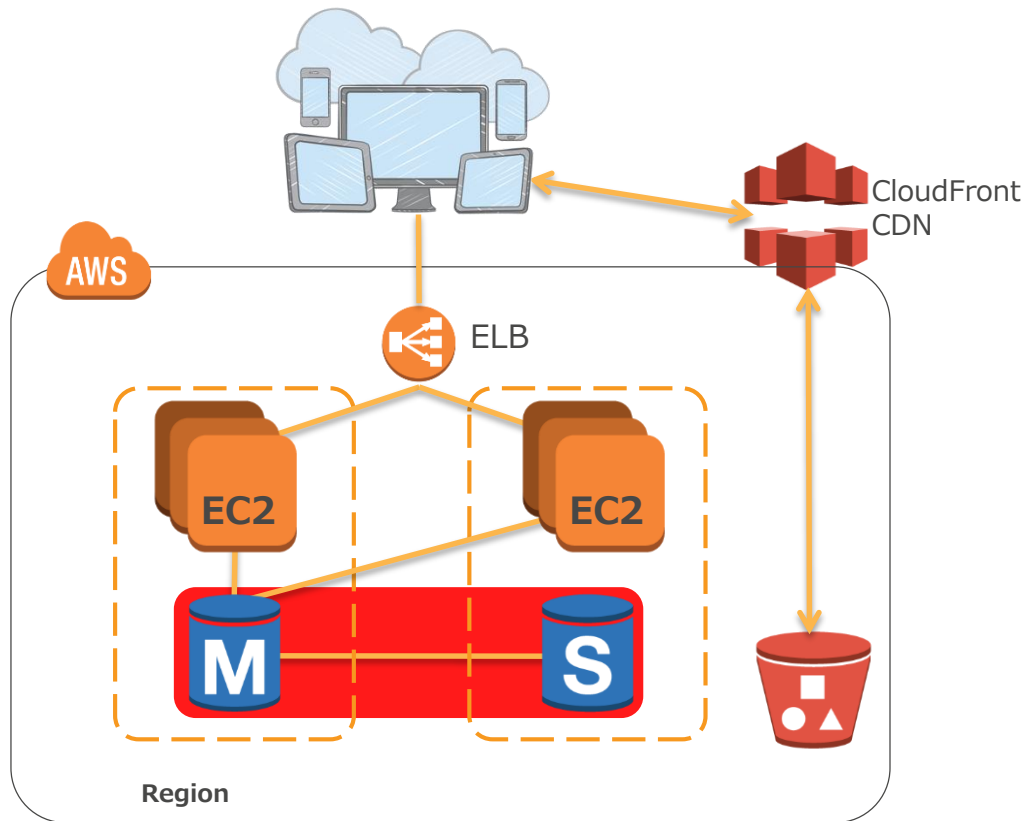
Case Study



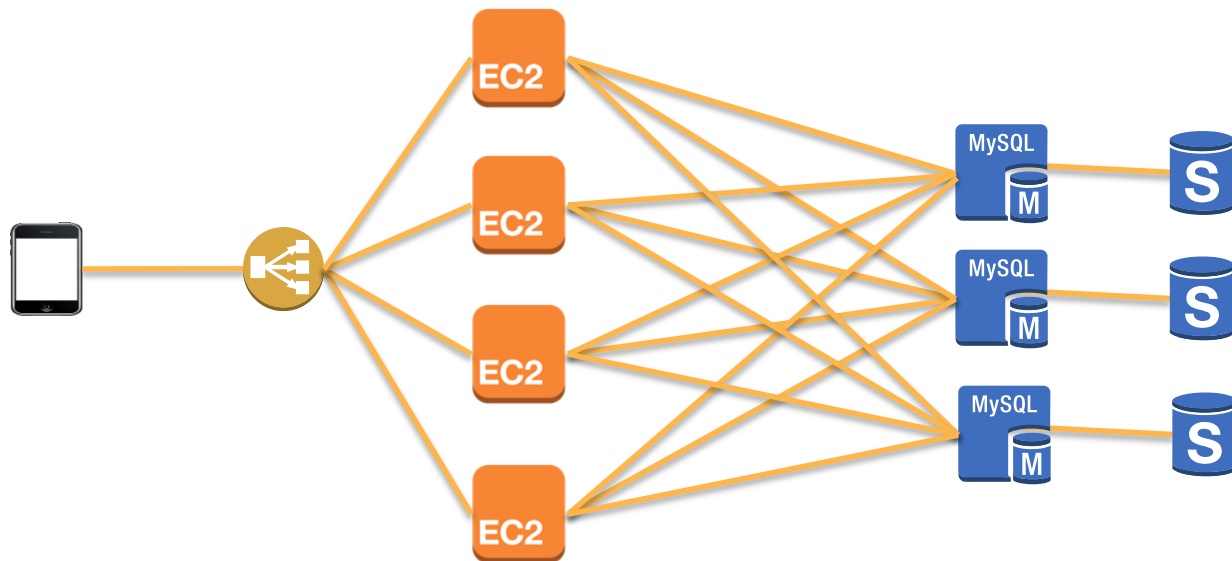
Tokyo Game Show 2014の発表資料より抜粋

# ゲームはDBライトヘビー・・・

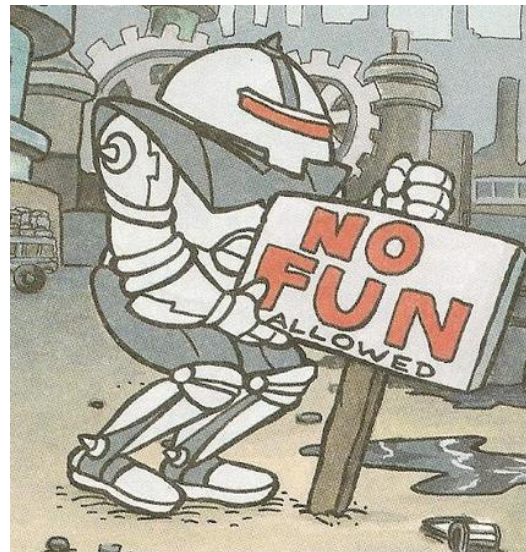
- DBライトヘビー
- 限界までキャッシング
- キーバリュー型へ変更  
(非正規化へ)
- バイナリーストラクチャー
- データベース= ボトルネック



# RDBのシャーディング

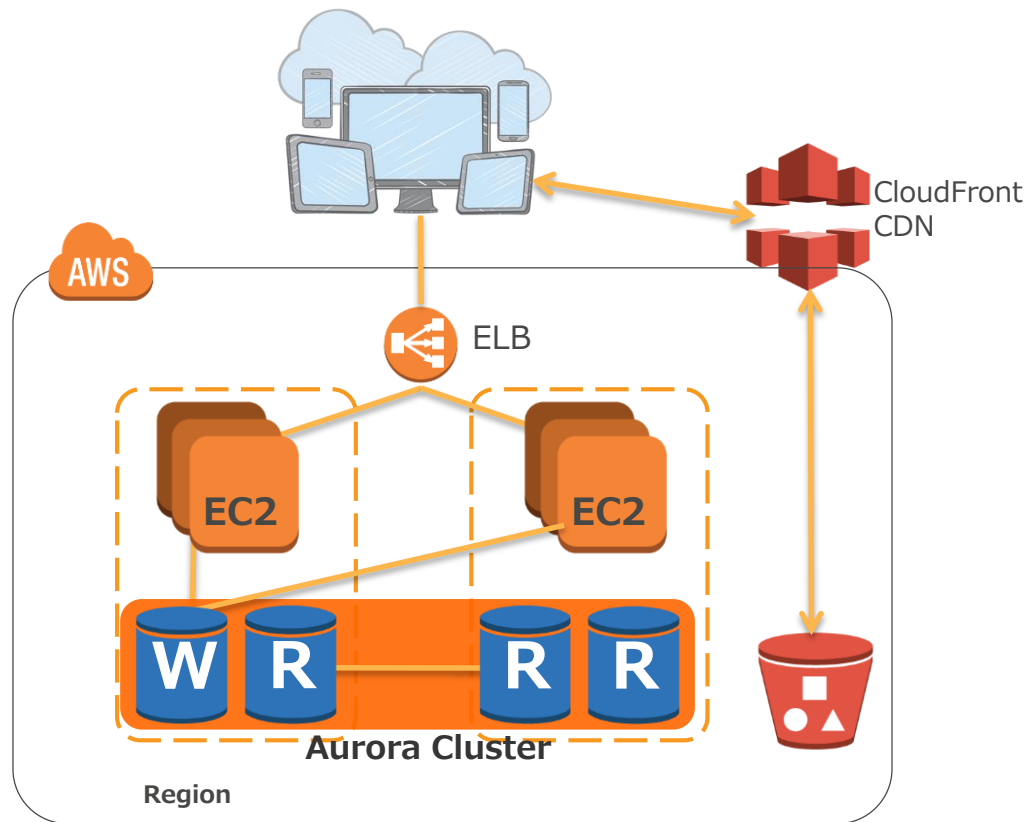


ツライ・・・



# Amazon Aurora !

- Amazonがクラウド時代にあわせて作ったRDBMS
- MySQL5.6互換
- 最大64TBまで自動でスケール
- 15台のレプリカ
- 高速なクラッシュリカバリ
- 高速なフェールオーバー
- 秒単位のポイントインタイムリカバリ

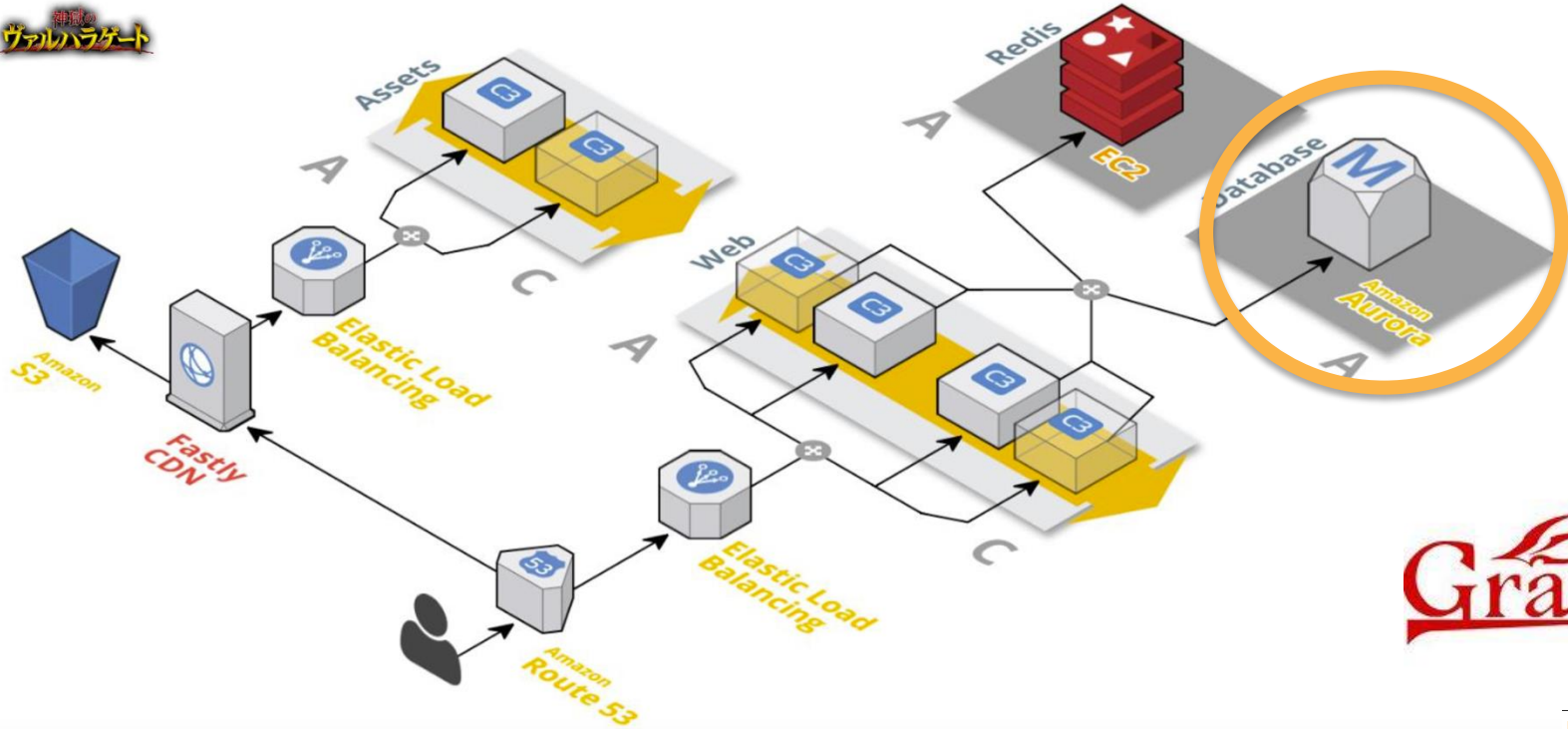


# AWS 導入事例:株式会社グラニ

## Case Study

### Architecture of Valhalla Gate

神界  
ヴァルハラゲート

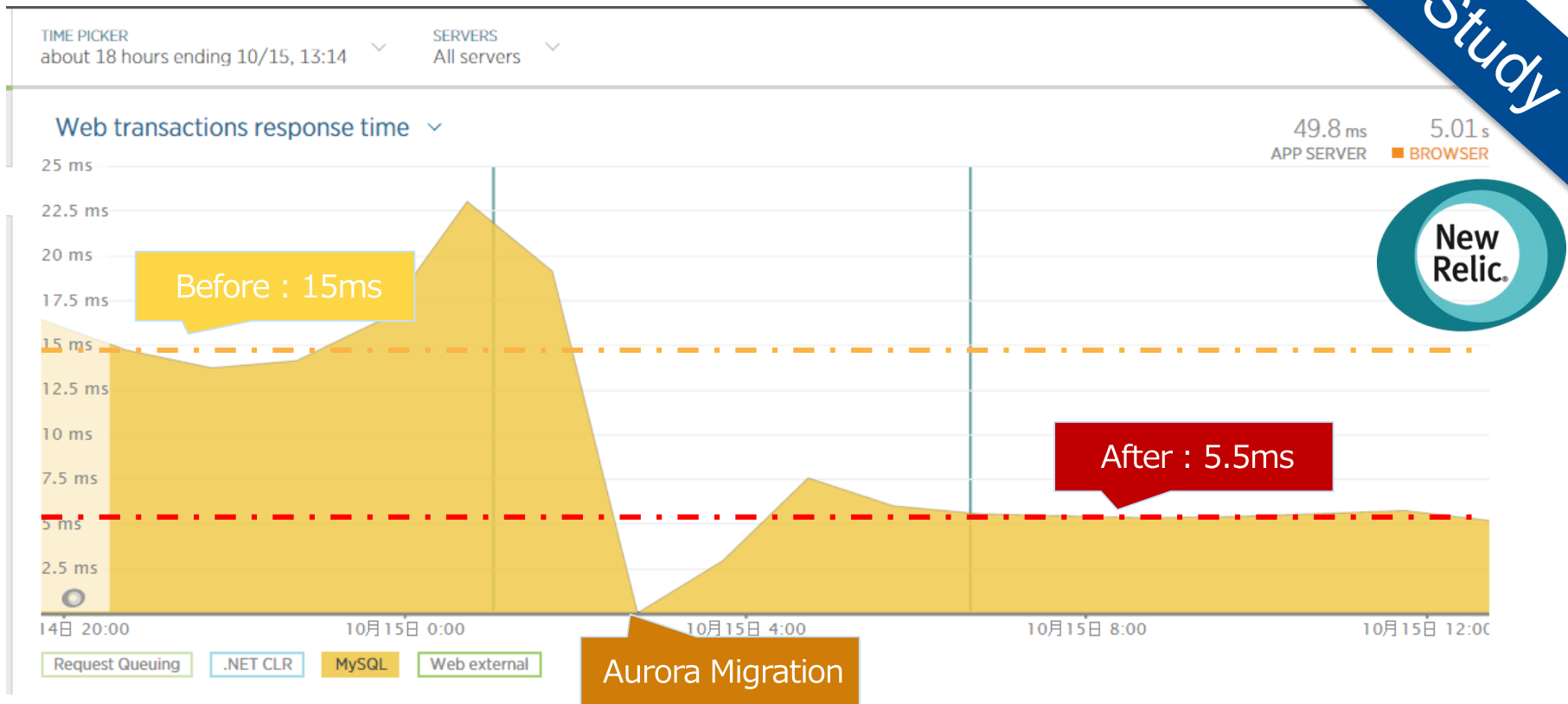


Grani



# Aurora 3x faster than MySQL (Total)

Case Study



# Amazon Auroraによるコスト削減効果

Case Study

性能向上によるDB 統合でノード削減も期待できる

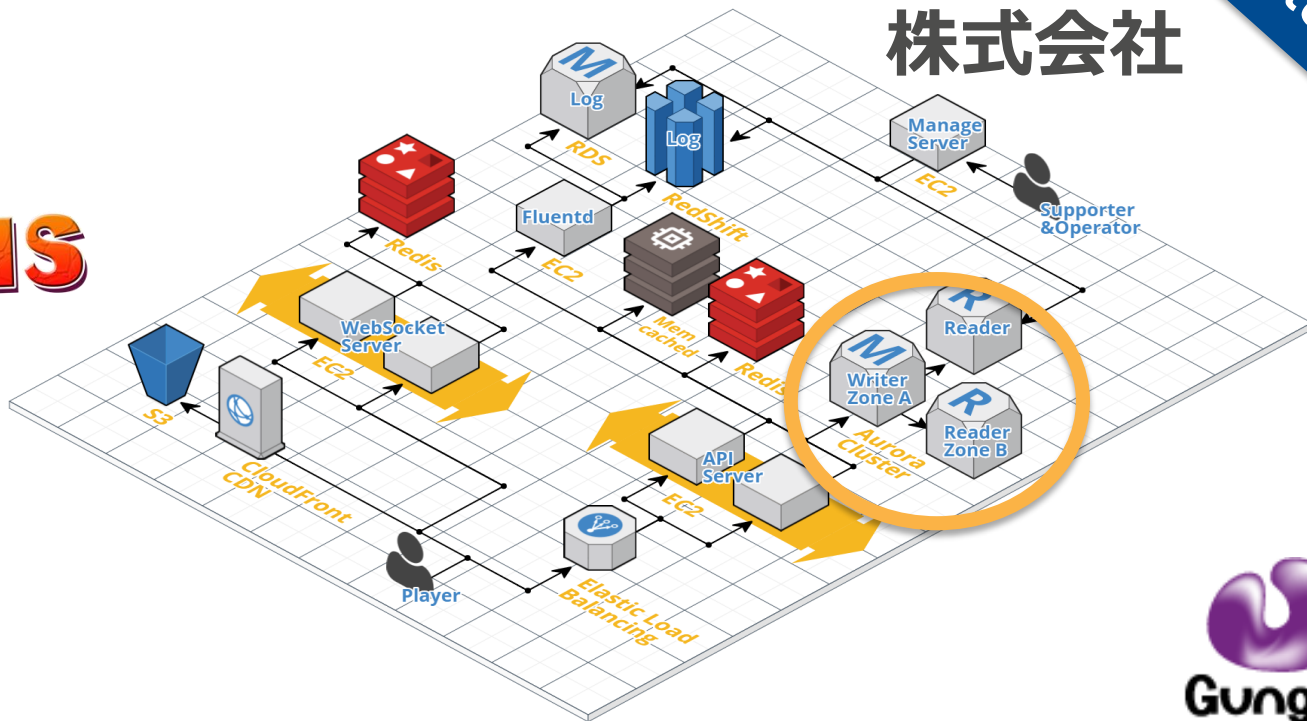
Grani様の場合、DB統合も行い **年間 2,200万円超** のコスト削減効果

RDS (db.r3.4xlarge / gp2 / OnDemand)	Hourly	Daily	Yearly
RDS for MySQL(MultiAZ + 1 ReadReplica)	$\$4.54 + \$2.27 = \$6.81$	\$163.44	\$59,655.60
Aurora (+ 1 Replica)	$\$2.8 * 2 = \$5.6$	\$134.40	\$49,056.00
削減効果	▲\$1.21	▲\$29.04	▲\$10599.6

# AWS 導入事例: ガンホー・オンライン・エンターテイメント

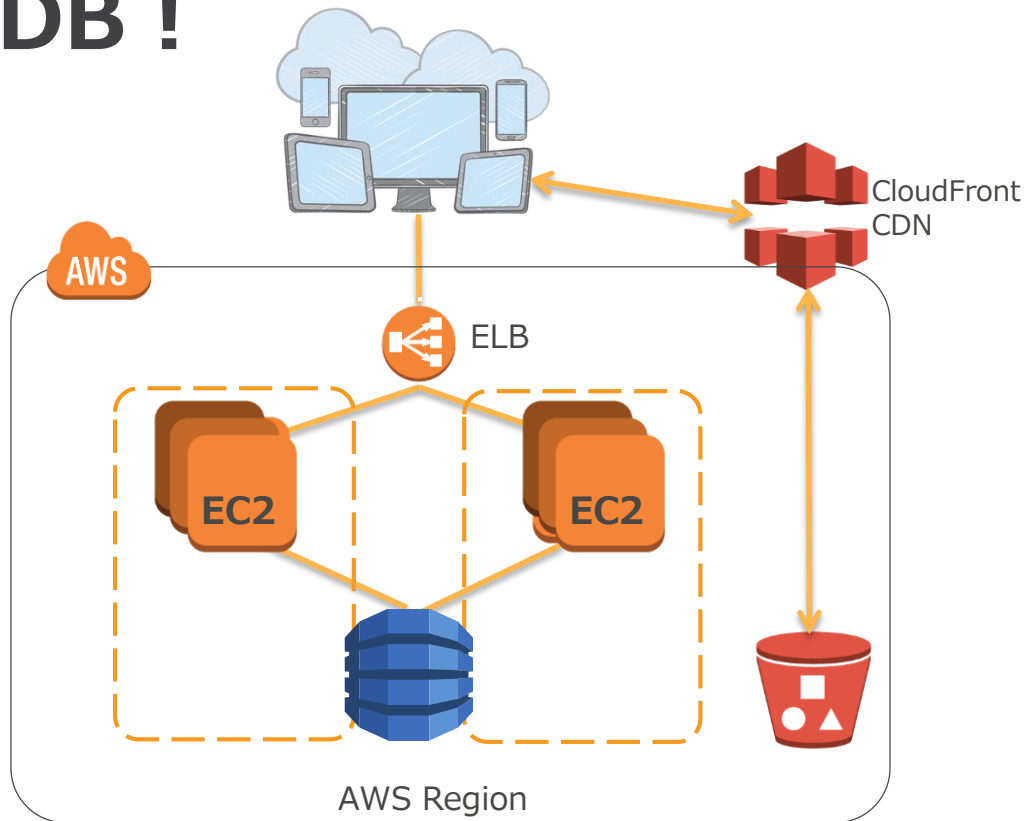
株式会社

PUZZLE  
&  
DRAGONS



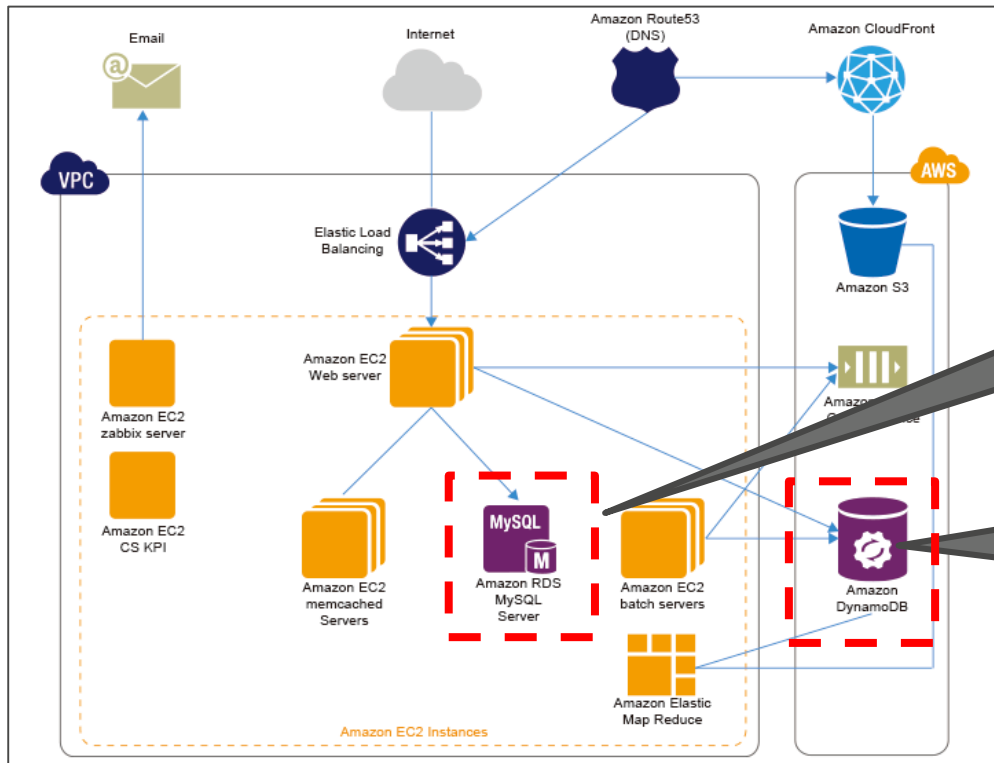
# Amazon DynamoDB !

- NoSQL データベース
- フルマネージド
- スキーマレス
- 高可用性
- 容量無制限
- プロビジョンスループット
- セカンダリーインデックス
- PUT/GET Keys
- Document型 (JSON) も



# AWS 導入事例:株式会社マイネット

## Case Study



DynamoDBで管理が難しいデータはRDB(RDS)で管理するも、全体のボリュームは小さくなるため運用が容易に。

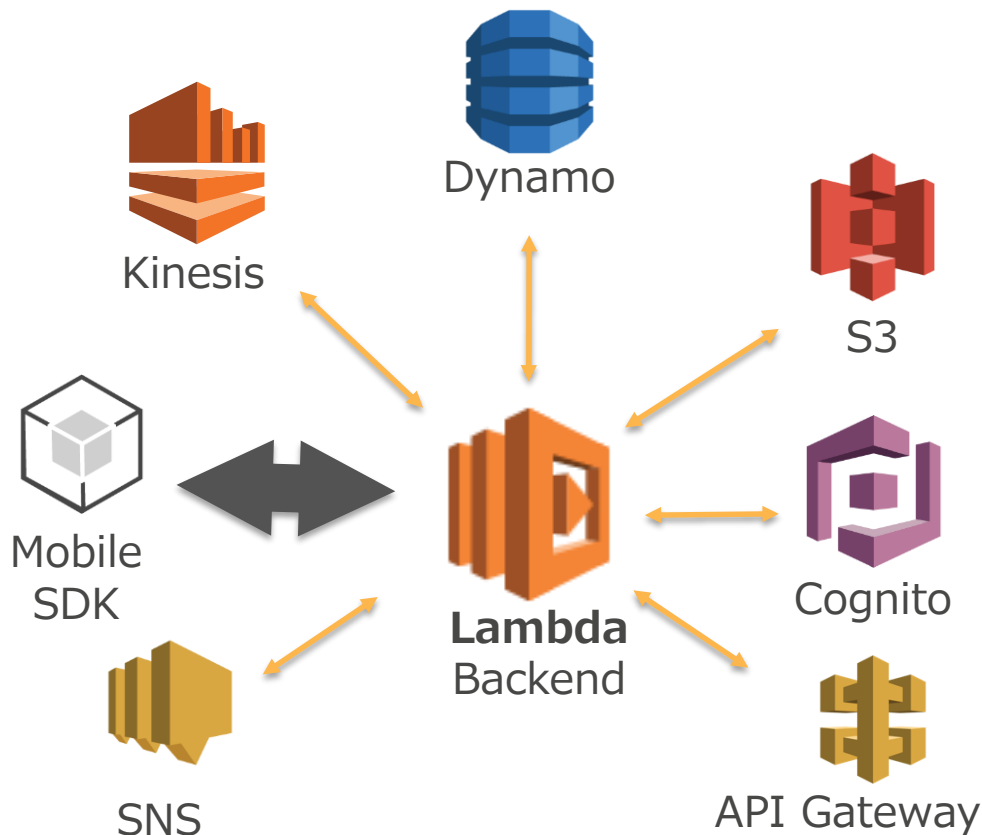
DynamoDBをメインDBにすることで、可用性向上のための運用はAWSが実施。トラフィック増減もアーキテクチャを変えずにパラメータ変更だけで対応。

参考 AWS事例のページより。詳細は以下をご参照ください。  
<http://aws.amazon.com/jp/solutions/case-studies/mynet/>

# Serverless Architecture

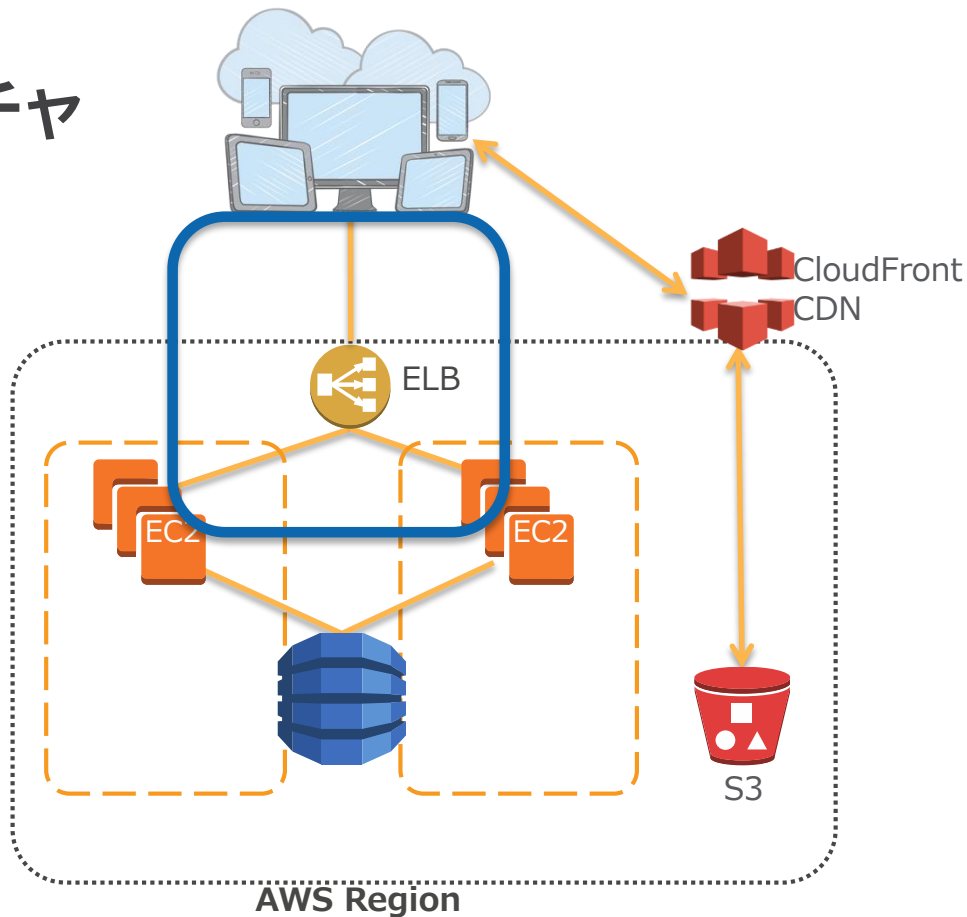
# Serverless Architecture

仮想サーバーである**Amazon EC2**  
**を使わず**に、クラウドネイティブに  
アプリケーションやサービスを  
開発、実行するための**サービス群**  
とアーキテクチャー



# 従来のゲームサーバ アーキテクチャ

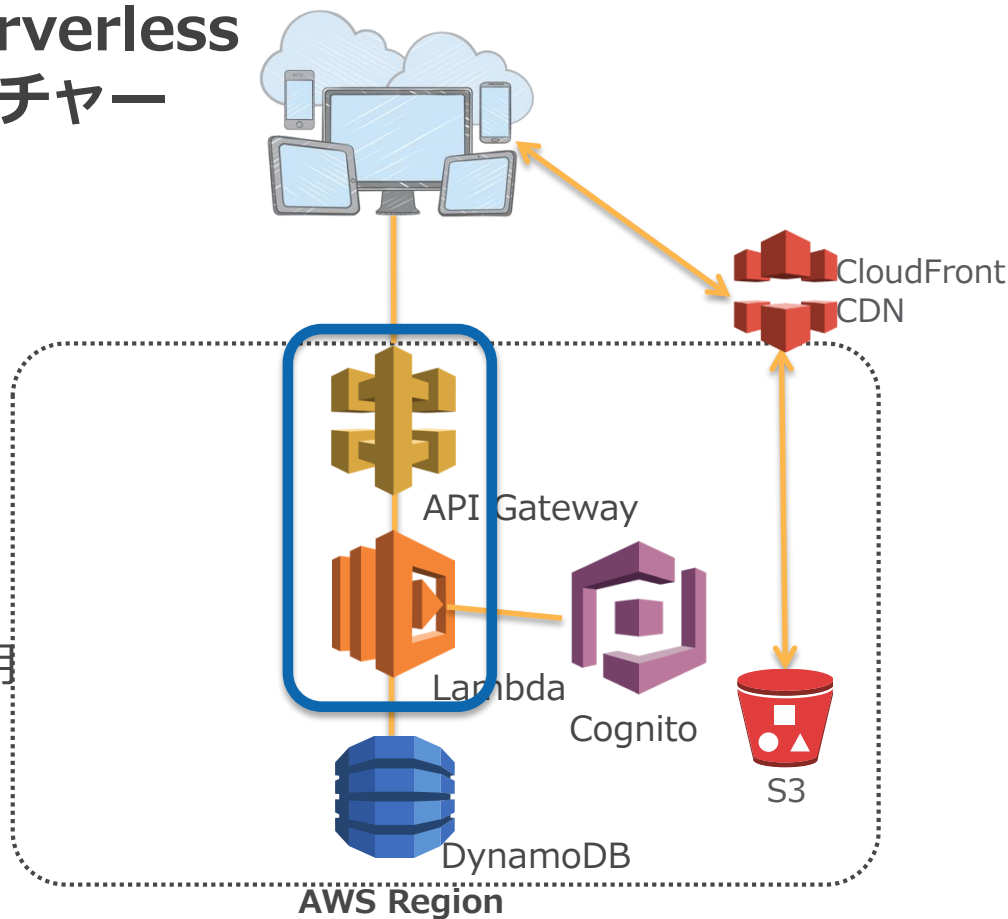
- ゲームAPIサーバ
  - ユーザログイン
  - ロビーサーバ
  - スコアボード
  - プレゼントリスト
- HTTP(S) API
  - HTTP(S)でアプリケーションサーバと通信





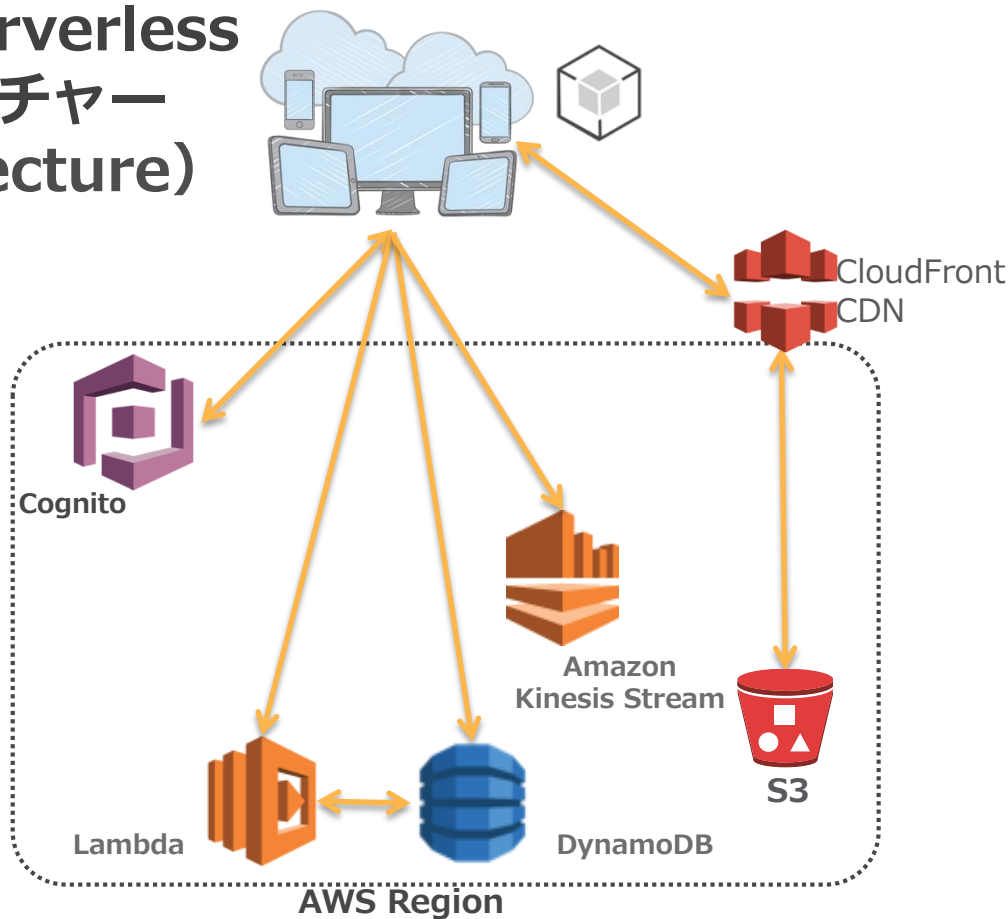
# クラウドネイティブなServerless ゲームアーキテクチャ

- API Gateway
  - REST APIサーバ
  - OS、キャパシティ等インフラの管理不要
  - スロットリング/キャッシュ
- Lambda
  - 各AWSサービスも利用可能
  - 認証にはCognito



# クラウドネイティブなServerless ゲームアーキテクチャー (2-Tier Architecture)

- Cognitoから  
Temporary  
Credentialsを取得
- AWS SDKを通じて各  
AWSリソースのAPIを  
直接アクセス



# 解決される課題

## ゲームアーキテクチャ

### インフラ構築

### インフラの運用管理

- キャパシティ
- スケール
- デプロイボード
- 耐障害性
- モニタリング
- ログ
- セキュリティパッチの適用

### ビジネスの差別化には直接繋がらない機能のアプリ実装

- 認証
- スロットリング
- スケーラビリティの確保



# 解決される課題

## ゲームアーキテクチャ

### インフラ構築

### インフラの運用管理

- キャパシテナー
- スケール List
- デプロイボード
- 耐障害性
- モニタリングクシヨ
- ログイングーピス
- セキュリティパッチの適用

### ビジネスの差別化には直接繋がらない機能のアプリ実装。

- 認証利用した別システ
- スロットリング可能
- スケーラビリティの確保

# 不要



解決される課題

ゲームアーキテクチャ

インフラ構築

インフラの運用管理

# 面白いゲーム 開発に集中！

—スケーラビリティの確保

# 各アーキテクチャでのメリット/デメリット

アーキテクト	メリット	デメリット
従来のゲーム サーバアーキテクチャ	クライアント側は従来のノウハウをフルに活かせる 実績が多く枯れた構成である カスタマイズ性が高い	サーバのスペック、台数などインフラを意識して設計する必要がある サーバの運用は利用者に任されている
Serverless ゲームアーキテクチャ (API Gateway)	クライアント側の実装は従来とあまり変わらずノウハウを活かせる サーバの運用、スケールはAWSに一任 CognitoによるセキュアなAPIアクセス制御が可能 コスト効率が高い	新規性が高く、まだ枯れていない
Serverless ゲームアーキテクチャ (2-Tier Architecture)	インフラ運用、スケール管理は不要 CognitoによるセキュアなAPIアクセス制御が可能 Web APIの設計が不要 最小限のパーツを組合せ コストメリットが高い	新規性が高く、まだ枯れていない クライアントサイドが各AWSリソースに依存する

どのアーキテクチャを採用すべきか？

三者択一でなく  
メリット/デメリットを  
見て組み合わせを

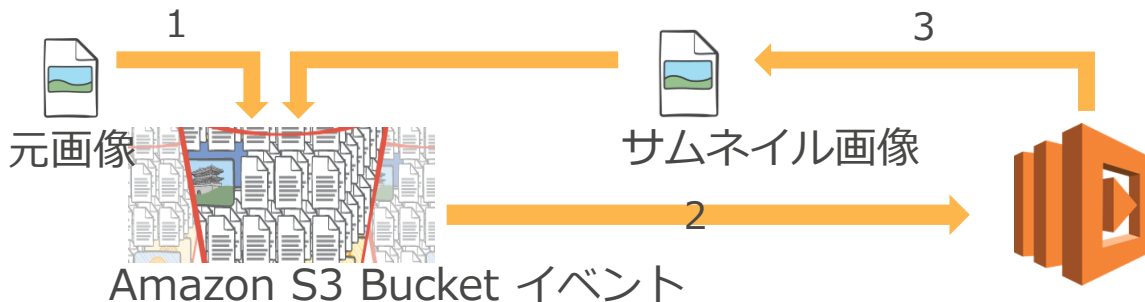


# AWS Lambda

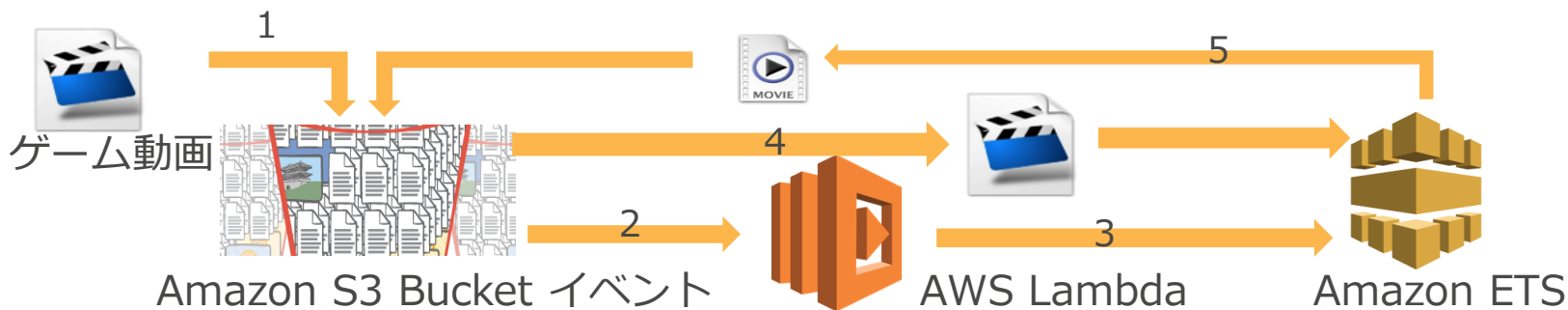


# Lambdaを使ったゲームでの活用

S3にゲーム画面のキャプチャー画像がアップロードされたときにサムネイルの生成やリサイズ、コピーライトの挿入を実行



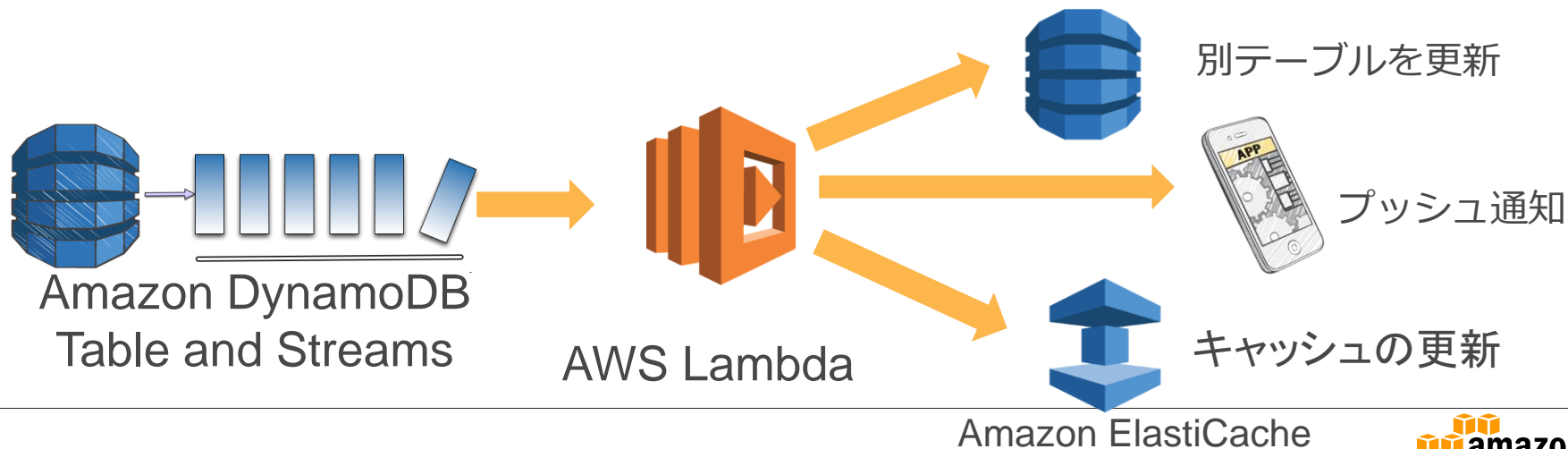
ゲーム動画ファイルのエンコーディング、サムネイル画像を生成



# DynamoDB Streams + Lambdaを使った 値チェックや別テーブルへのコピー

DynamoDB テーブル内の項目レベルの変更の時系列にキャプチャー  
DynamoDBへの書き込みに応じて値チェックをし、別テーブルの更新や  
プッシュ通知を実行

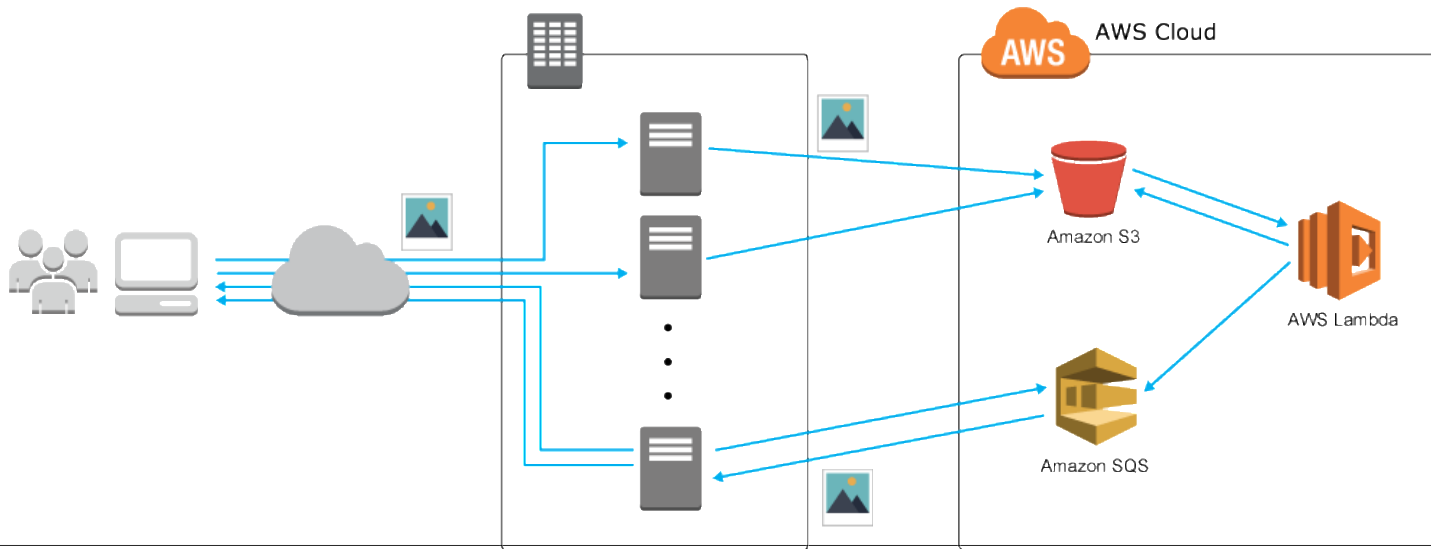
データチェックによってチート判定などにも利用可能



# AWS 導入事例:株式会社スクウェア・エニックス ドラゴンクエストX(思い出アルバム)

ゲーム内で写真撮影し、Lambdaでサムネイル画像作成やコピーライトの追加加工

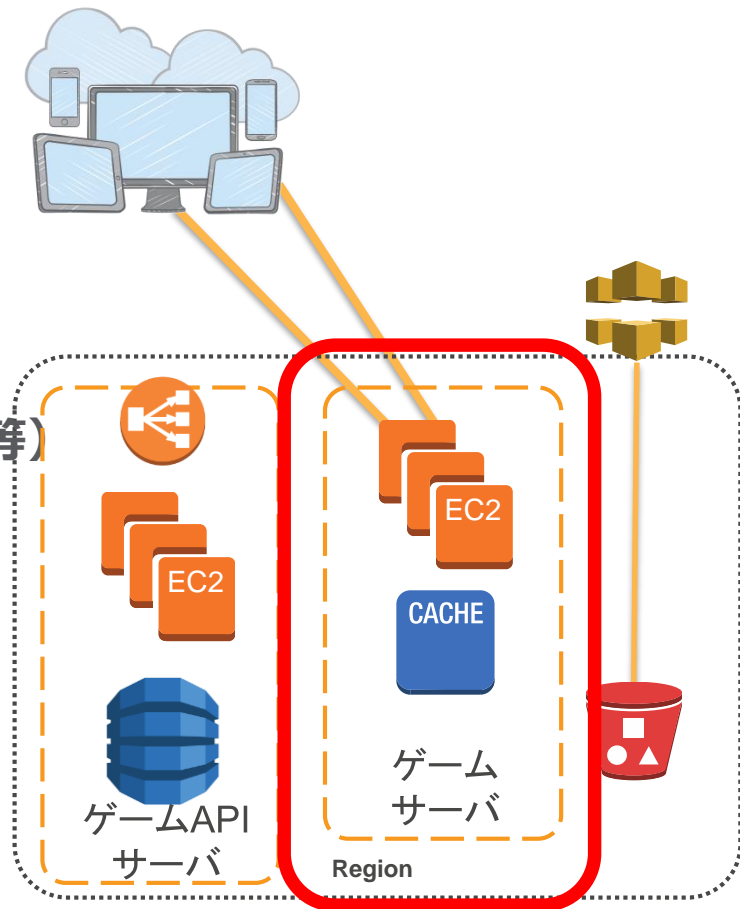
- 処理時間が**数時間→10 数秒**で完了
- 同じ処理をオンプレミスと比べ **20 分の 1 程度まで削減**



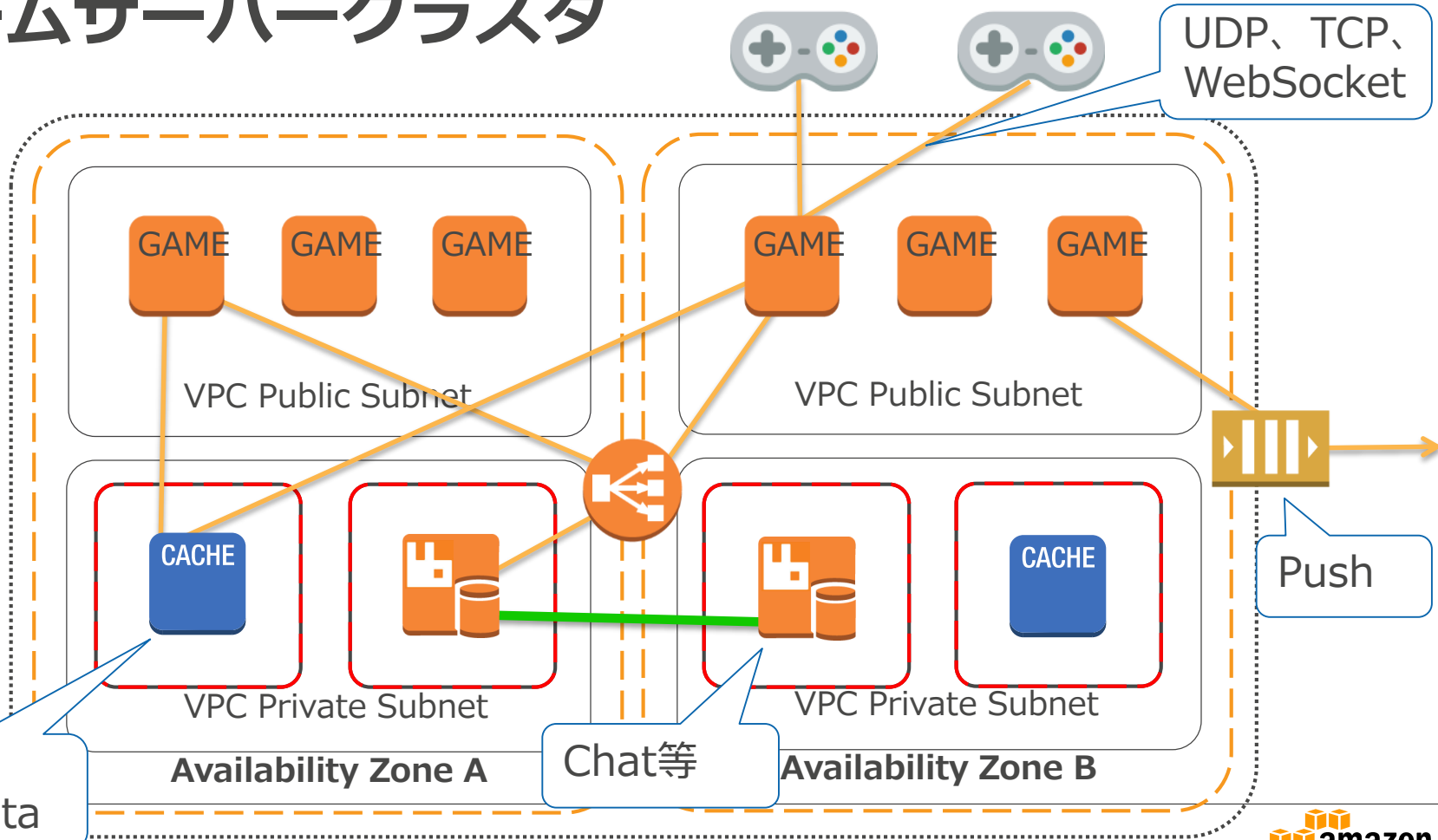
# ゲームサーバ

# オンラインゲームのフロー

- ① HTTP APIを用いてログイン
- ② Game Assetsをダウンロード
- ③ Game Serverのマッチメイク
- ④ **Serverへ接続（TCP, UDP, WebSocket等）**
- ⑤ **離れた仲間と対戦**
- ⑥ **Game Over**
- ⑦ HTTP APIを使って結果の書き込み

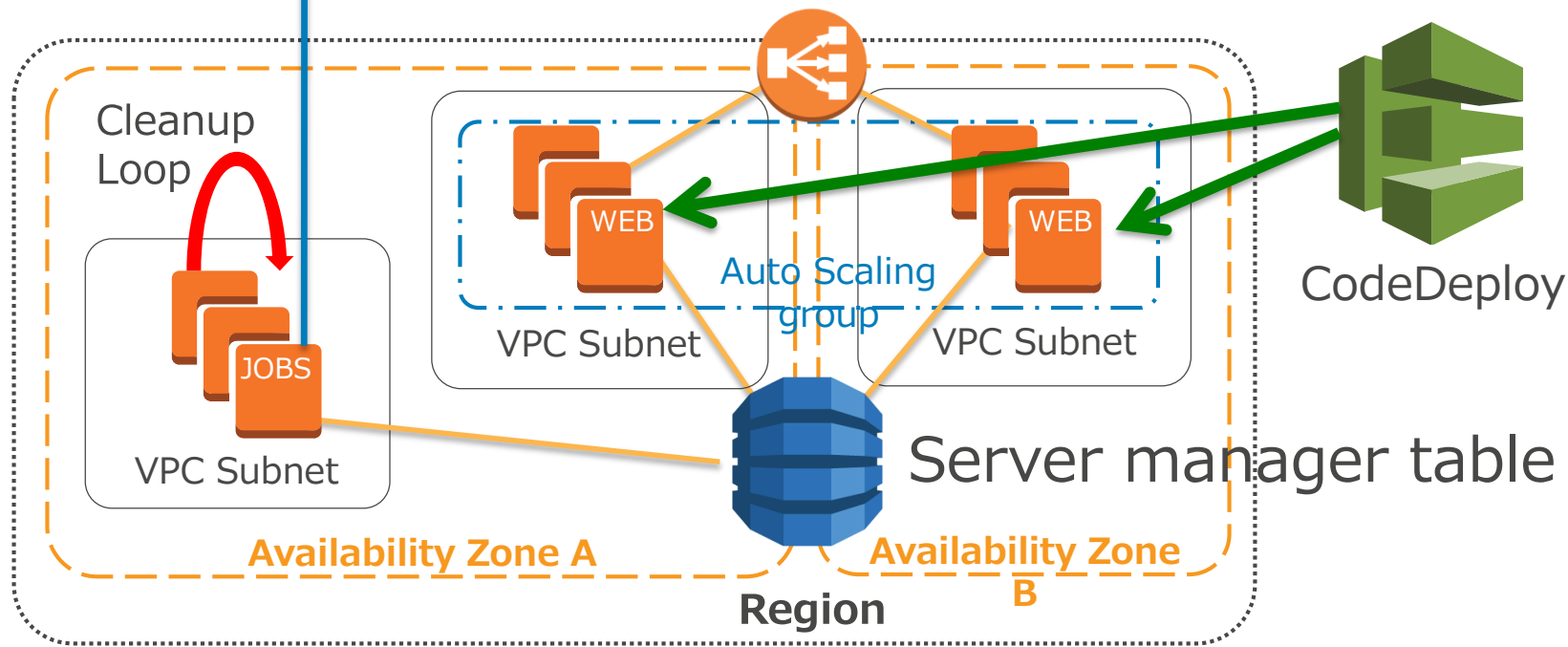


# ゲームサーバークラスタ



# ゲームサーバの起動、登録管理

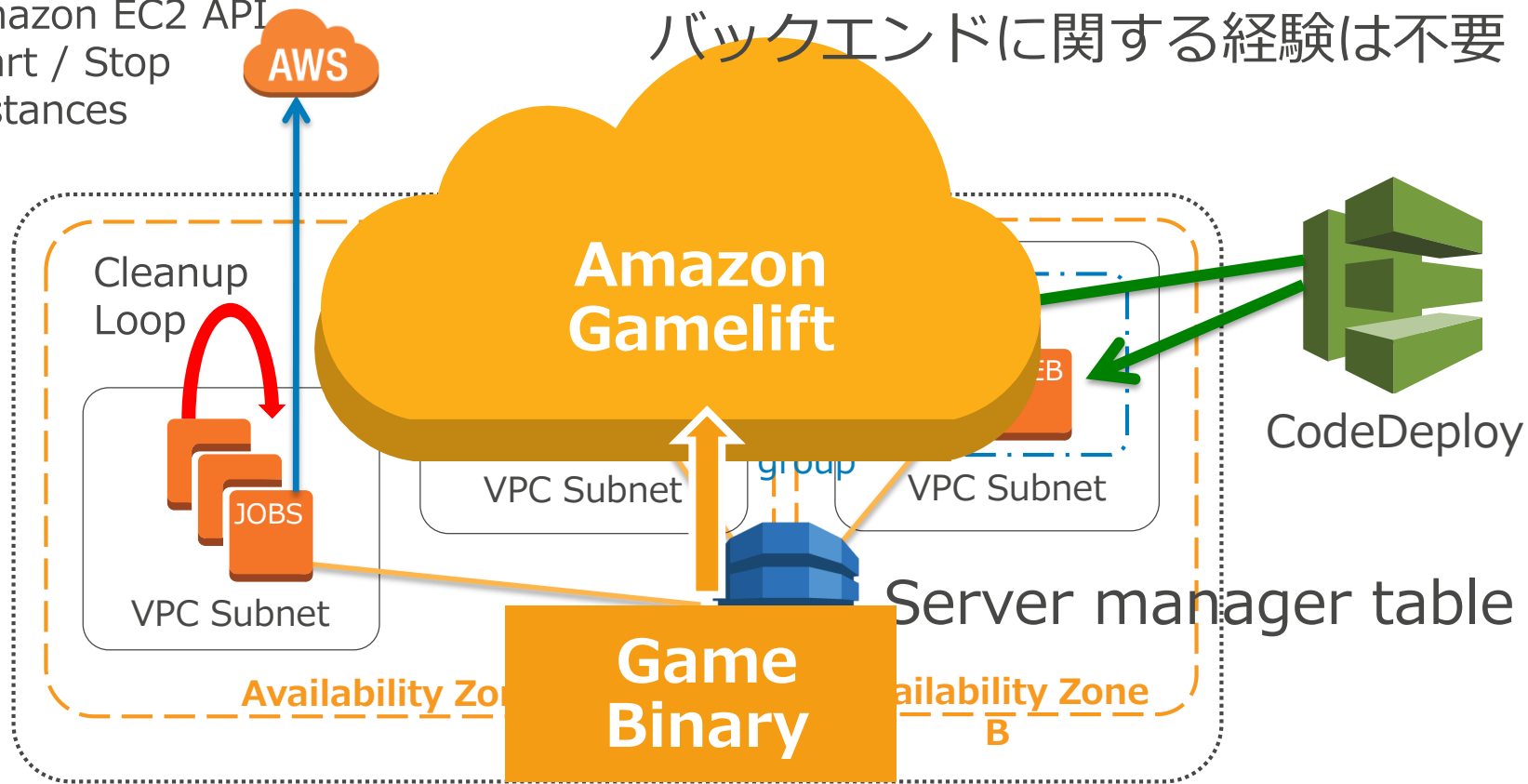
Amazon EC2 API  
Start / Stop  
インスタンス



# Amazon GameLift

Amazon EC2 API  
Start / Stop  
Instances

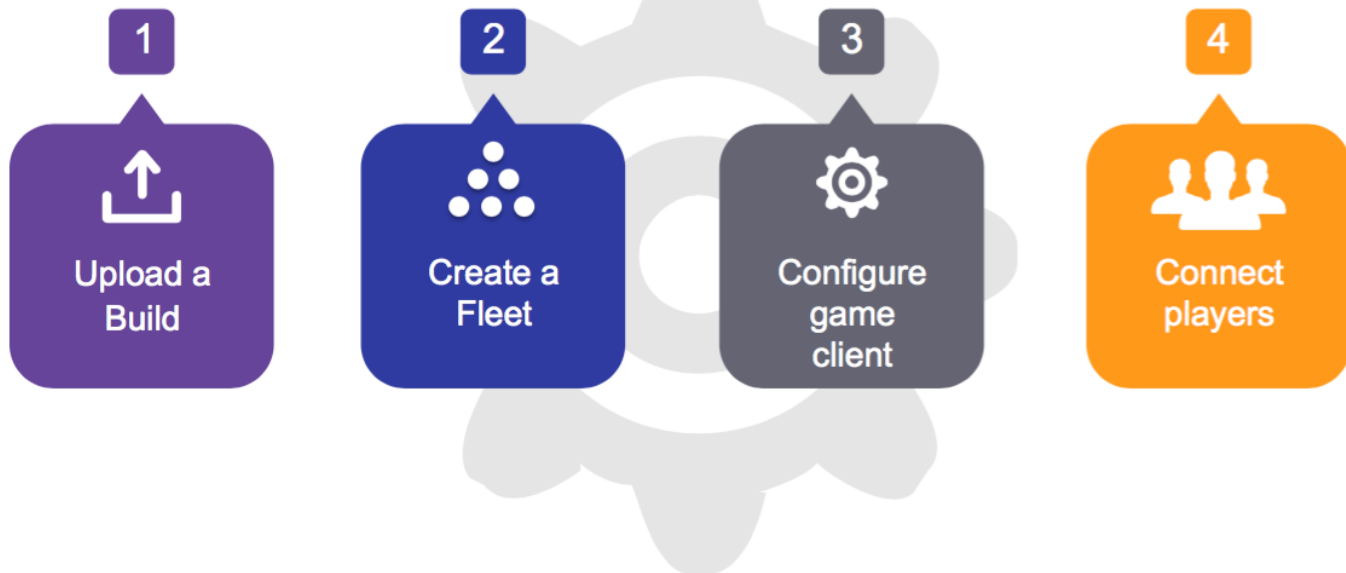
バックエンドに関する経験は不要 !

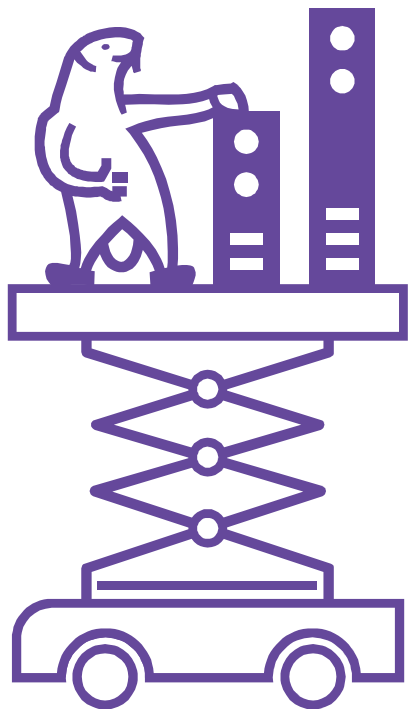




# GameLiftを始めるまでの手順

Deploy and scale, in minutes

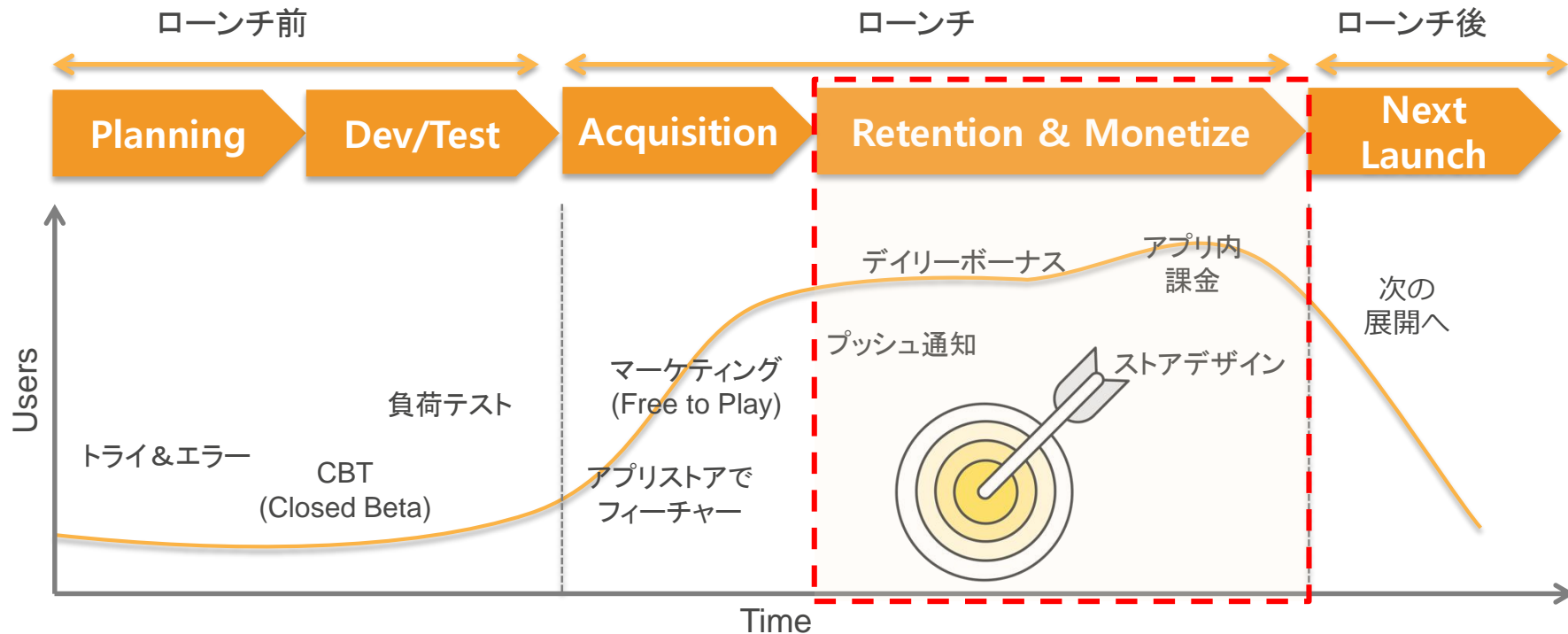




# Amazon GameLift

- セッションベースマルチプレイヤーゲームサーバのためのマネージド・サービス
- 設定したプレイヤーセッションによってインスタンスを自動でスケールアップ、ダウン
- 世界中のユーザへLow-latencyプレイヤ体験を提供
- ダウンタイム無しでライブアップデート
- ゲームのDAU単位による料金体系

# Game Life Cycle



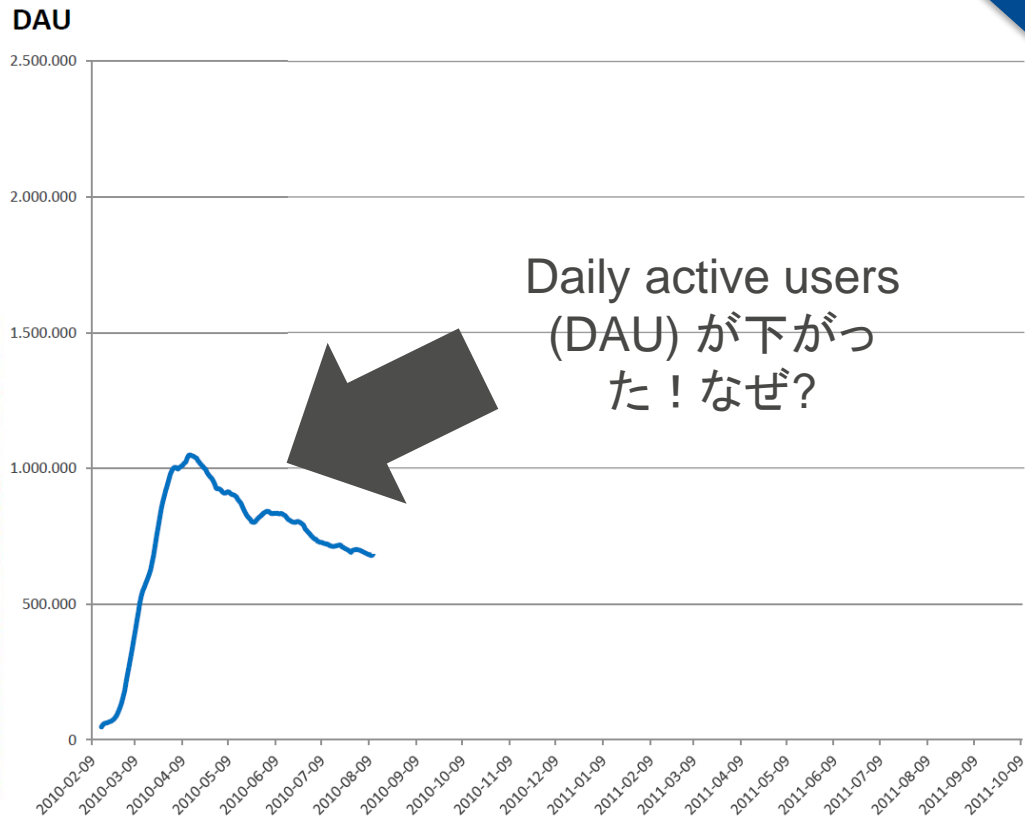
# Big Data!

- アナリティクスを活用し収益を確保しマネタイズを行う
- 何百万ものデータログをコスト効率よく保存
- トレンドを見極める
- 予測モデルを構築する
- リアルタイムに対応する



# User Retention: Bubble Island

## Case Study



# User Retention: Bubble Island

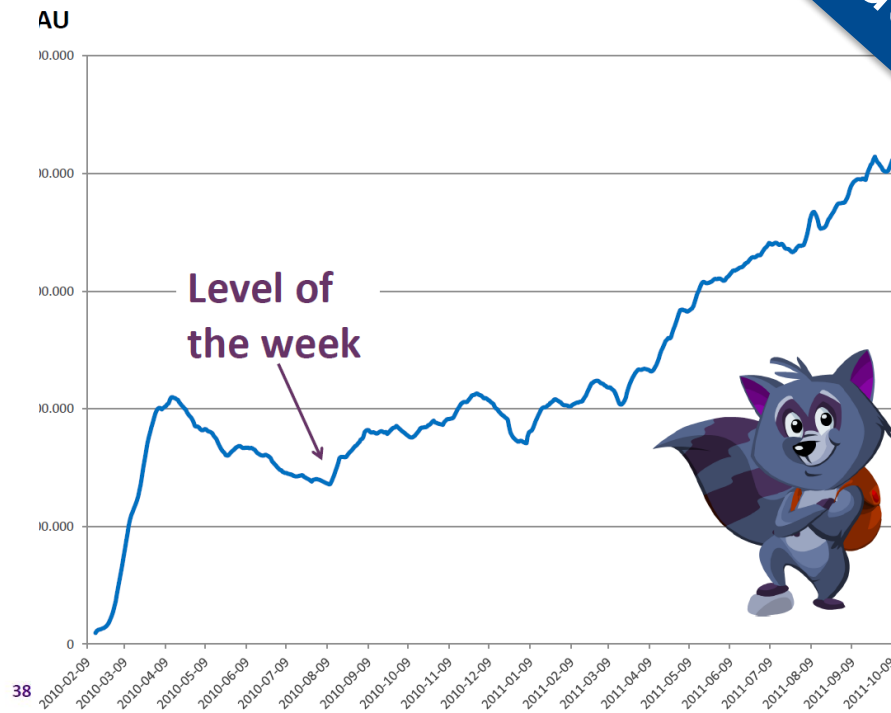
## Case Study

リリース数週間後:

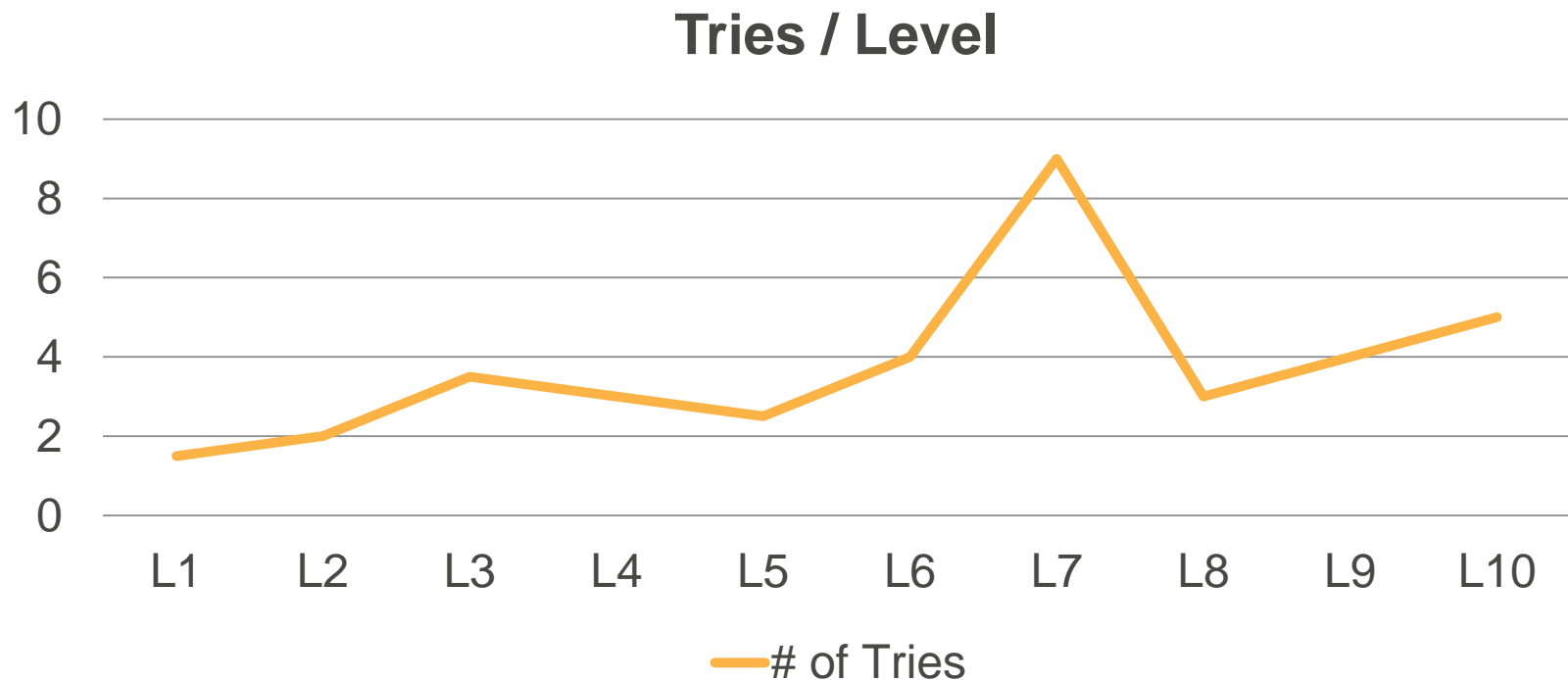
すべてのレベルをクリア?

OR

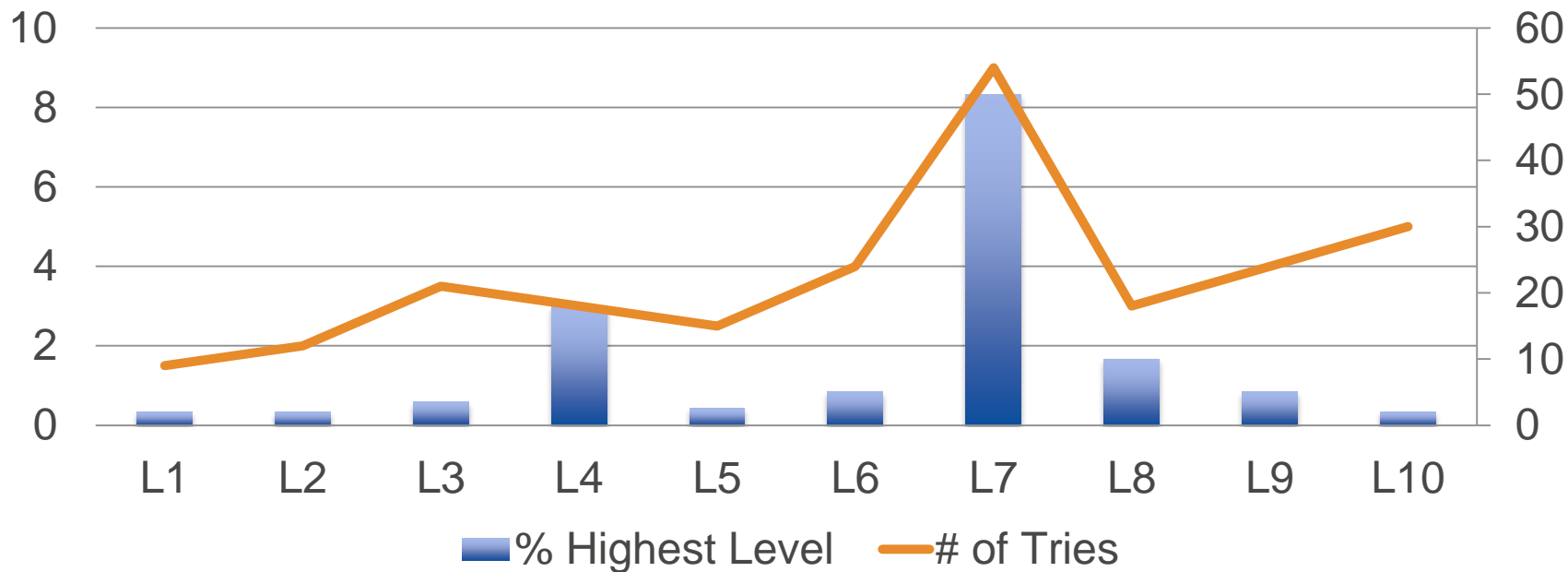
レベルデザインが難しい?



# Example: レベル進行(1つのメトリクス)



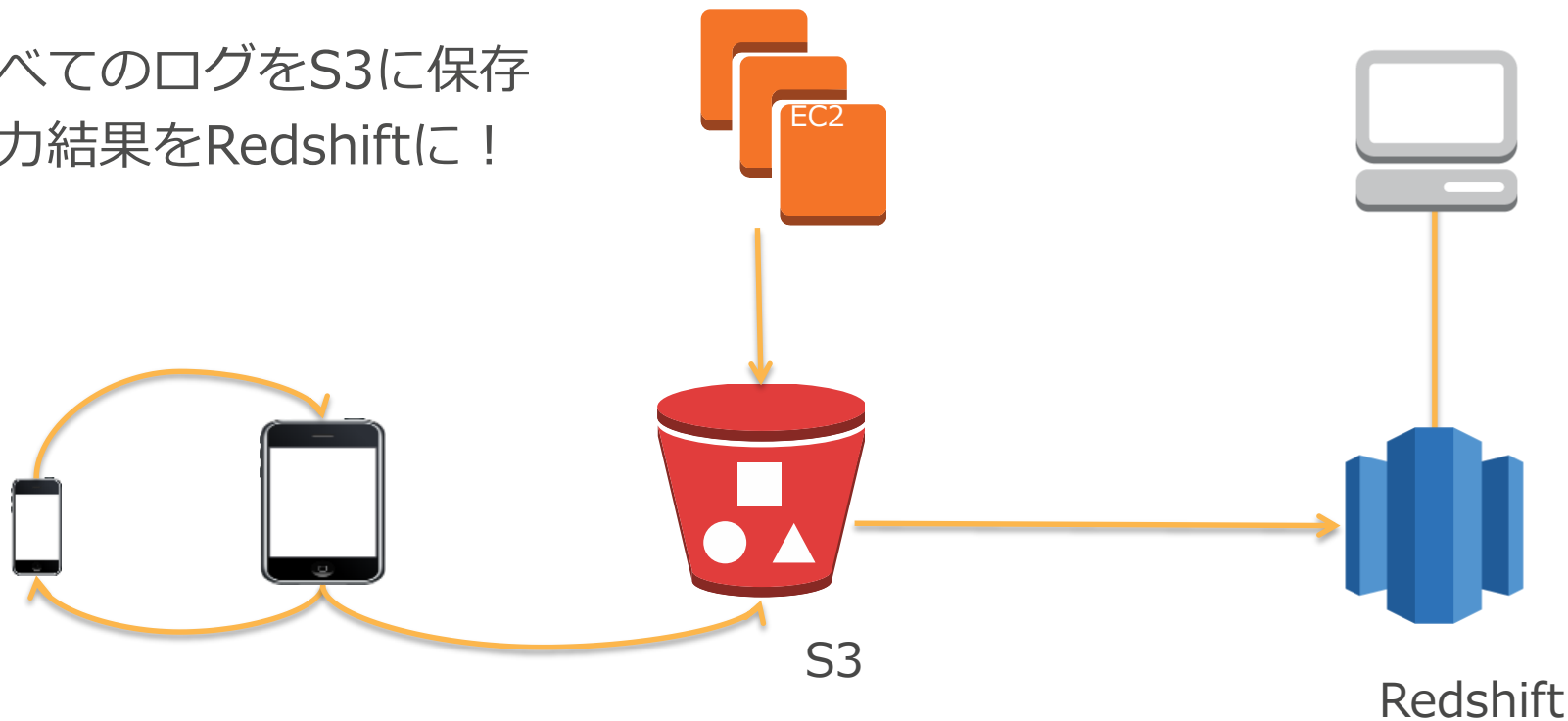
## Example: レベル進行(2つのメトリクス)





# はじめはシンプルに収集

すべてのログをS3に保存  
出力結果をRedshiftに！



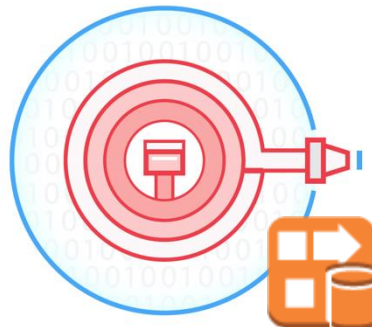
# Amazon Kinesis プラットフォーム

ストリーミングデータを収集・処理するためのフルマネージドサービス群



## Amazon Kinesis Streams

ストリーミングデータを  
処理するための  
アプリケーションを  
独自に構築



## Amazon Kinesis Firehose

ストリーミングデータを  
Amazon S3, Amazon  
Redshift, Amazon ES へ  
簡単に配信



## Amazon Kinesis Analytics

ストリーミングデータを  
標準的な SQL クエリーでリアル  
タイムに分析

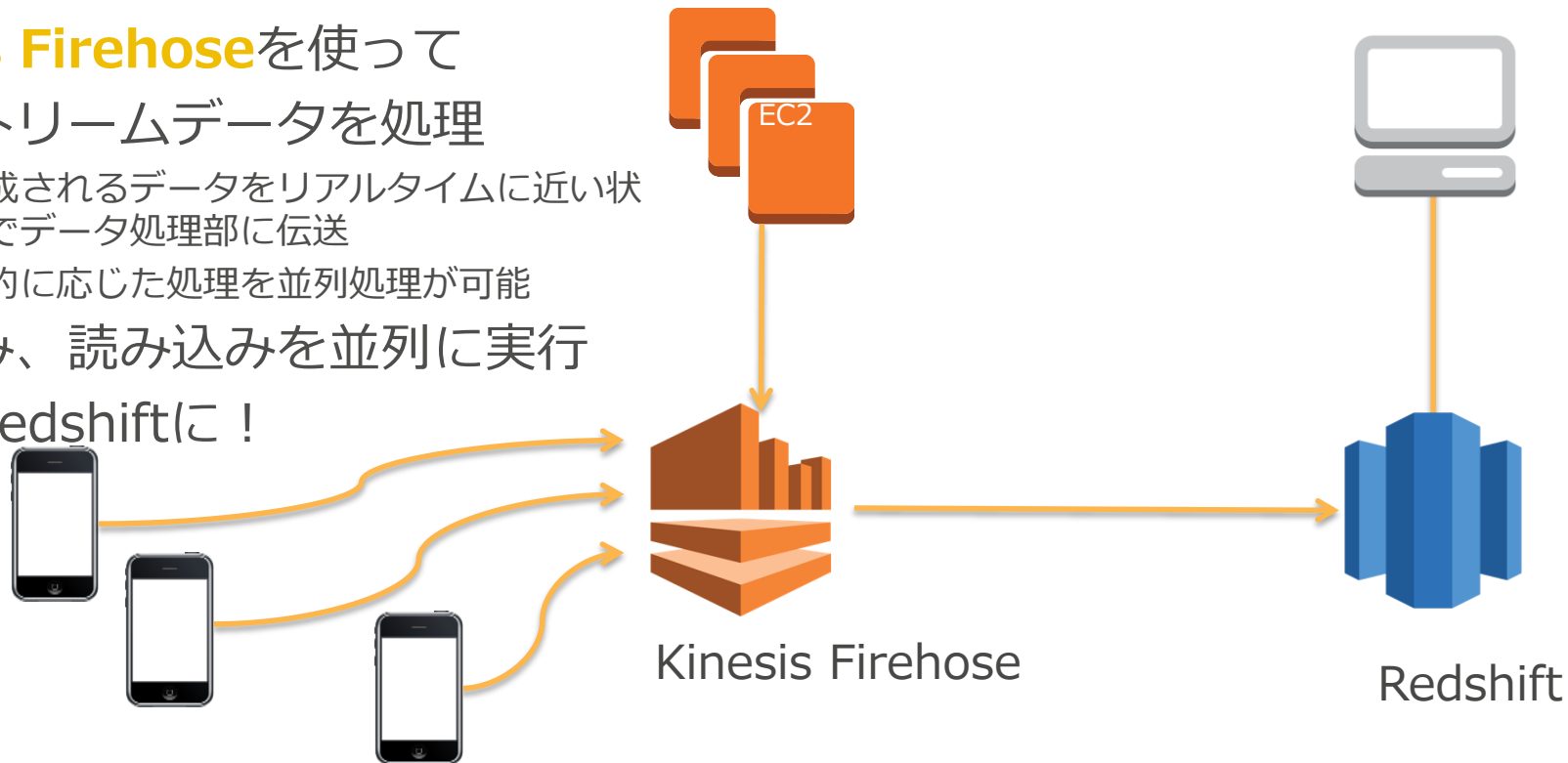
# 必要なデータはいち早く処理！

**Kinesis Firehose**を使って

ストリームデータ进行处理

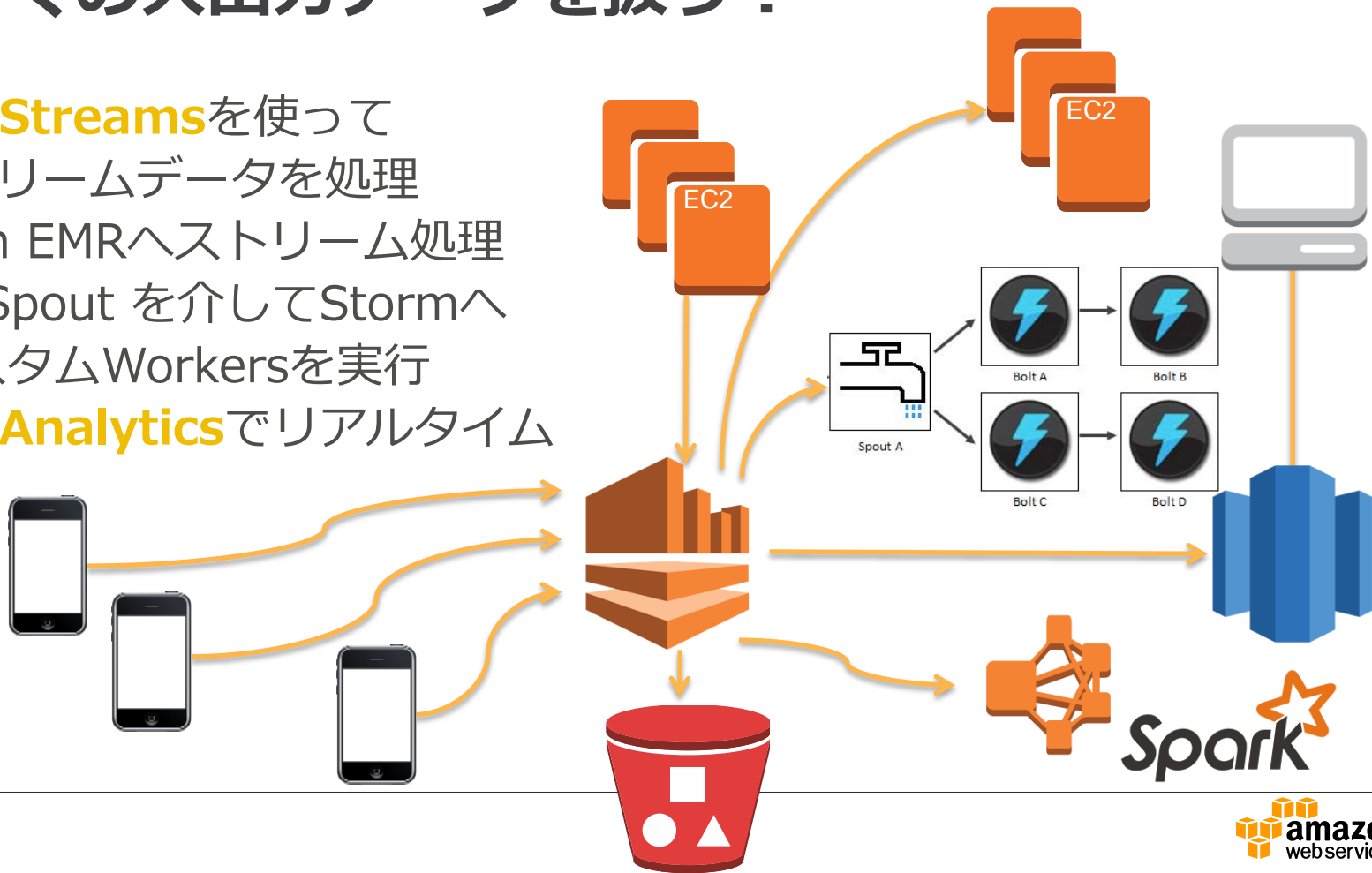
- 生成されるデータをリアルタイムに近い状況でデータ処理部に伝送
- 目的に応じた処理を並列処理が可能

書き込み、読み込みを並列に実行  
出力をRedshiftに！



# 更に多くの入出力データを扱う！

**Kinesis Streams**を使って  
ストリームデータ処理  
Spark on EMRへストリーム処理  
Kinesis Spout を介してStormへ  
EC2 カスタムWorkersを実行  
**Kinesis Analytics**でリアルタイム  
解析





**NEW!**

# Amazon Athena

Interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL

Generally Available Today

# Amazon Athenaを発表

- Amazon S3に置いたデータをインタラクティブにSQLクエリ可能。
- AthenaはPrestoで提供されるSQL Engineが利用でき、JSON, CSV, ログファイル, 区切り文字のあるテキストファイル, Apache Parquet, Apache ORCに対して、クエリが可能。
- ペタバイトクラスのデータに対するクエリをサポート。データをS3から取り込む手間はない
- ANSI-SQLをサポート。特殊なスキル不要。
- バージニア、オレゴンで利用可能





4TB/day  
Amazon  
S3

Archival -  
Amazon  
Glacier

24/7  
Hadoop  
-clusters  
EC2

Kinesis  
for Real-  
Time

Case Study

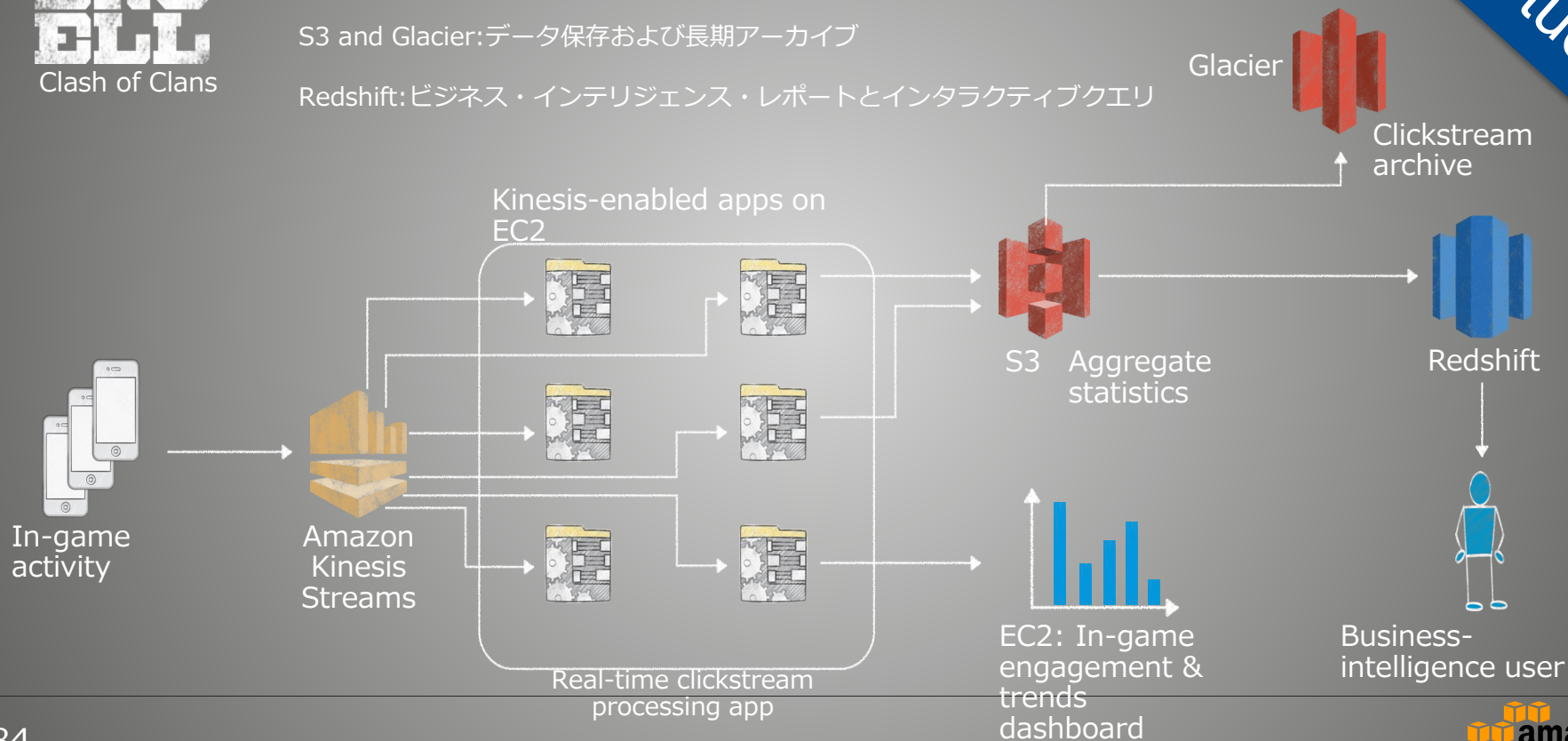


Kinesis Streams: ゲーム内の行動のリアルタイム・データ・ストリーム

Multiple Kinesis applications: ダッシュボード、分析、保管、予測

S3 and Glacier: データ保存および長期アーカイブ

Redshift: ビジネス・インテリジェンス・レポートとインタラクティブクエリ





# AWS re:Invent

GAM301

## Real-Time Game Analytics with Amazon Kinesis, Amazon Redshift, and Amazon DynamoDB

Suhas Kulkarni – VP Engineering, GREE International

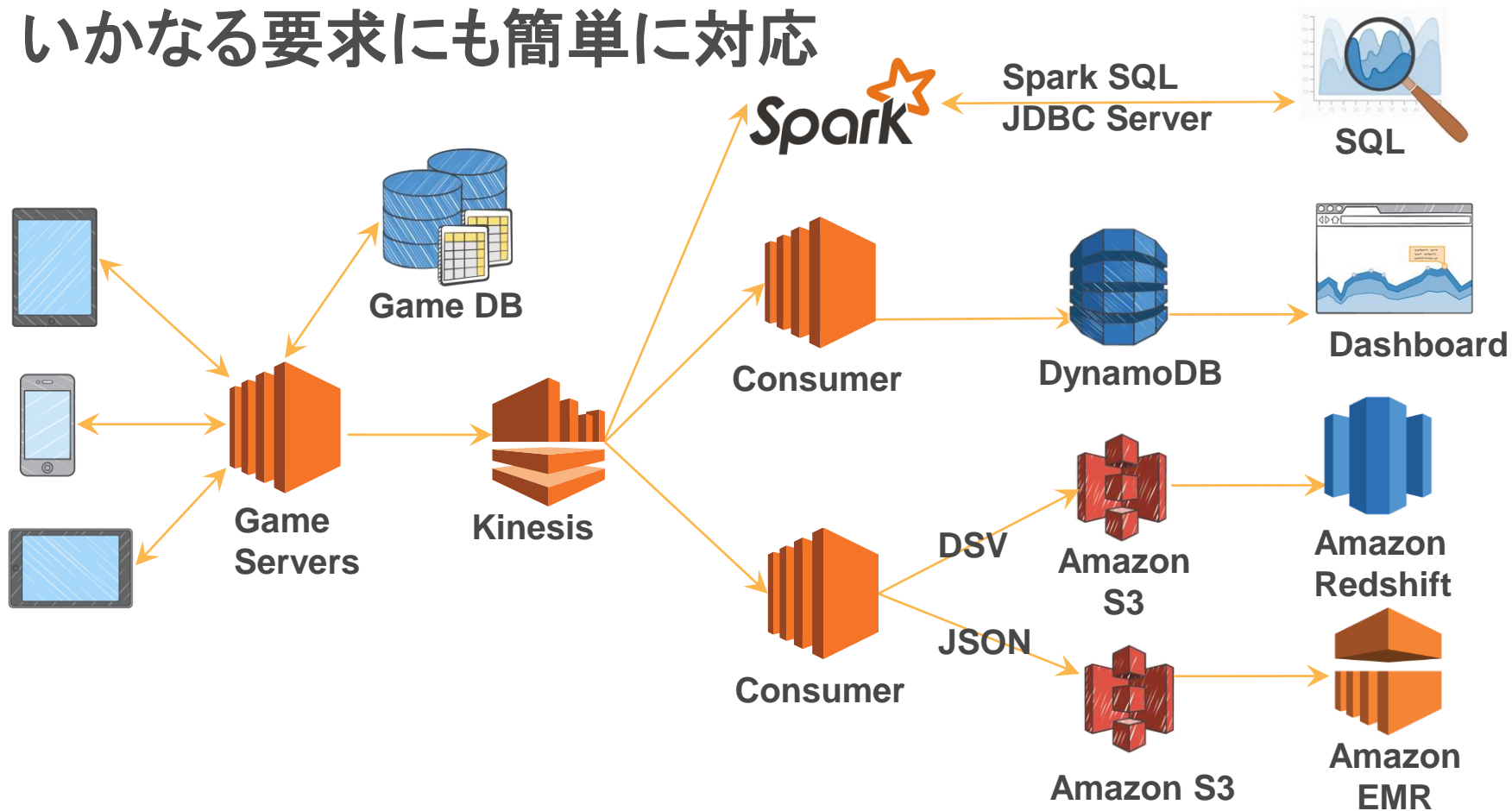
Kandarp Shah – Engineering Manager, GREE International

November 12, 2014 | Las Vegas

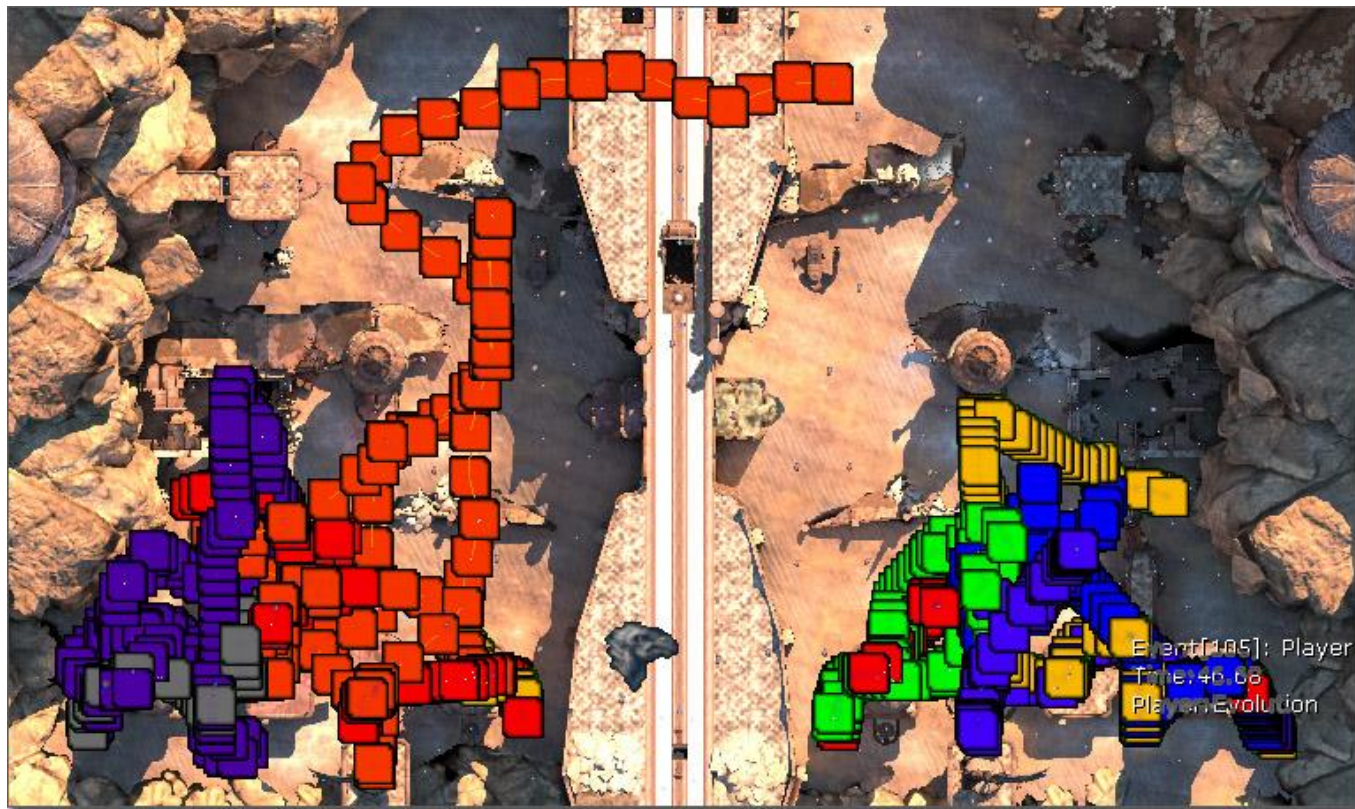


© 2014 Amazon.com, Inc. and its affiliates. All rights reserved. May not be copied, modified, or distributed in whole or in part without the express consent of Amazon.com, Inc.

# いかなる要求にも簡単に対応



# Real-Time ヒートマップ



# How to iterate?

ゲームの改修はすべての人への影響ありますか？



Old



New

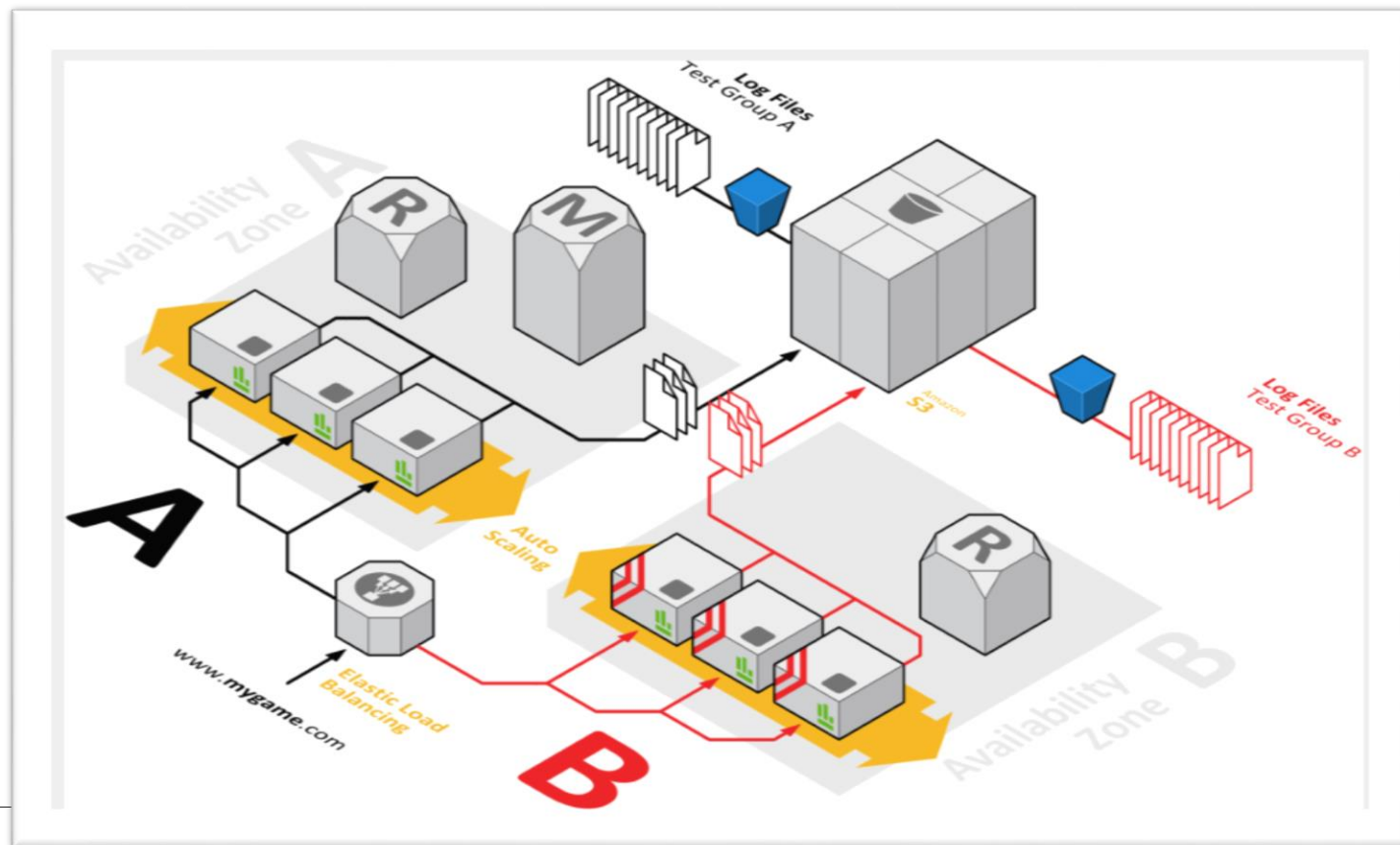


アイテム価格の変更を  
A/Bテストを実施

アイテム売上が24%増！



# AWS上でのA/B テスト

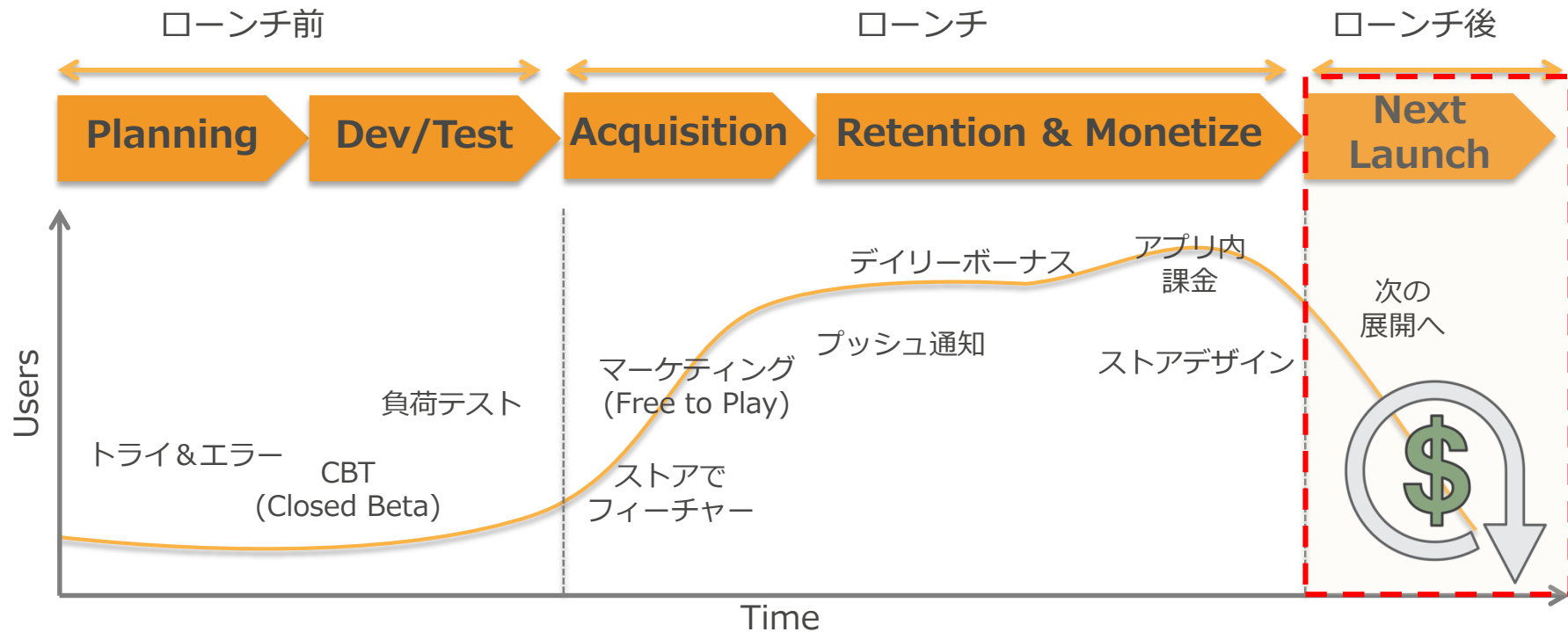


# AWS Pinpoint

- モバイルアプリケーション向けターゲットプッシュ通知サービス
- クライアントデバイスからエンゲージ情報を収集し、ユーザのセグメントおよびファンネル分析を実施
- カスタムイベントやカスタムアトリビュートを利用し、特定セグメントに対するキャンペーン通知と結果の再分析や、A/Bテスト分析に活用
- スケジュール通知やサイレント通知にも対応
- 送信メッセージ数、受信イベント数、ターゲットユーザー数毎に課金
- バージニアリージョンにて提供を開始



# Game Life Cycle



# 次の展開は？

- 数分で全てのリソース、コストを削減！

OR

- 新しい地域、国でゲームをローンチしますか  
(グローバル展開) ？





# Global Gaming on AWS

# ネットワークゲーム（FPS,MMO等）の特徴

- 多数のユーザーがひとつのゲームセッションやワールドに接続して遊ばれる
- あるユーザーのアクションは他のユーザーに対して（できるかぎり）リアルタイムに伝える必要がある

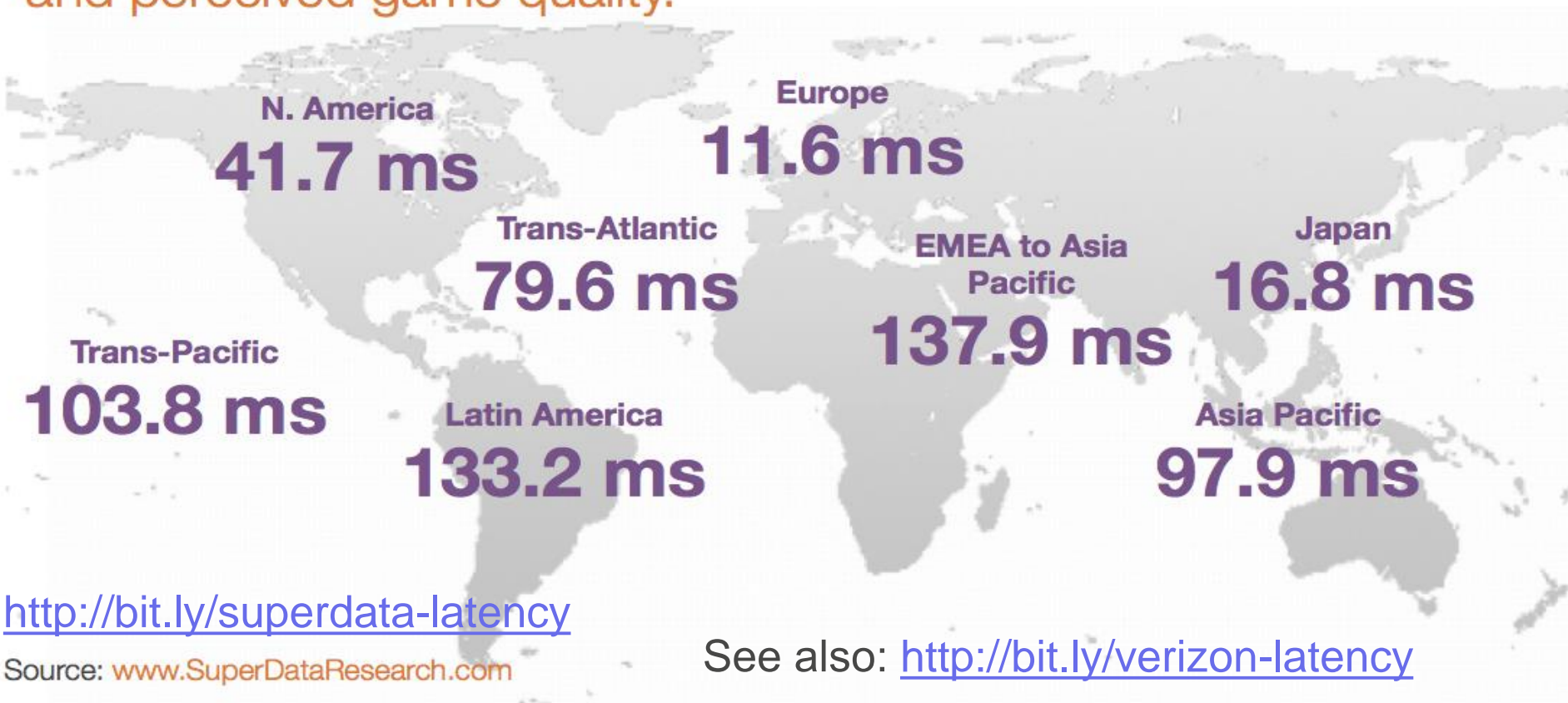
Challenge

世界は広く、光の速さには限界がある。

距離=タイムラグ

# Domestic latency in selected regions:

"higher latency decreases game response time, user performance and perceived game quality."

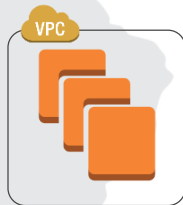
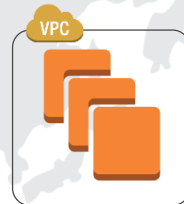
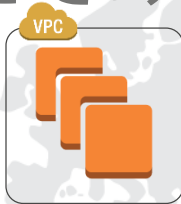
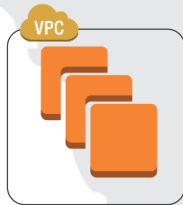


<http://bit.ly/superdata-latency>

Source: [www.SuperDataResearch.com](http://www.SuperDataResearch.com)

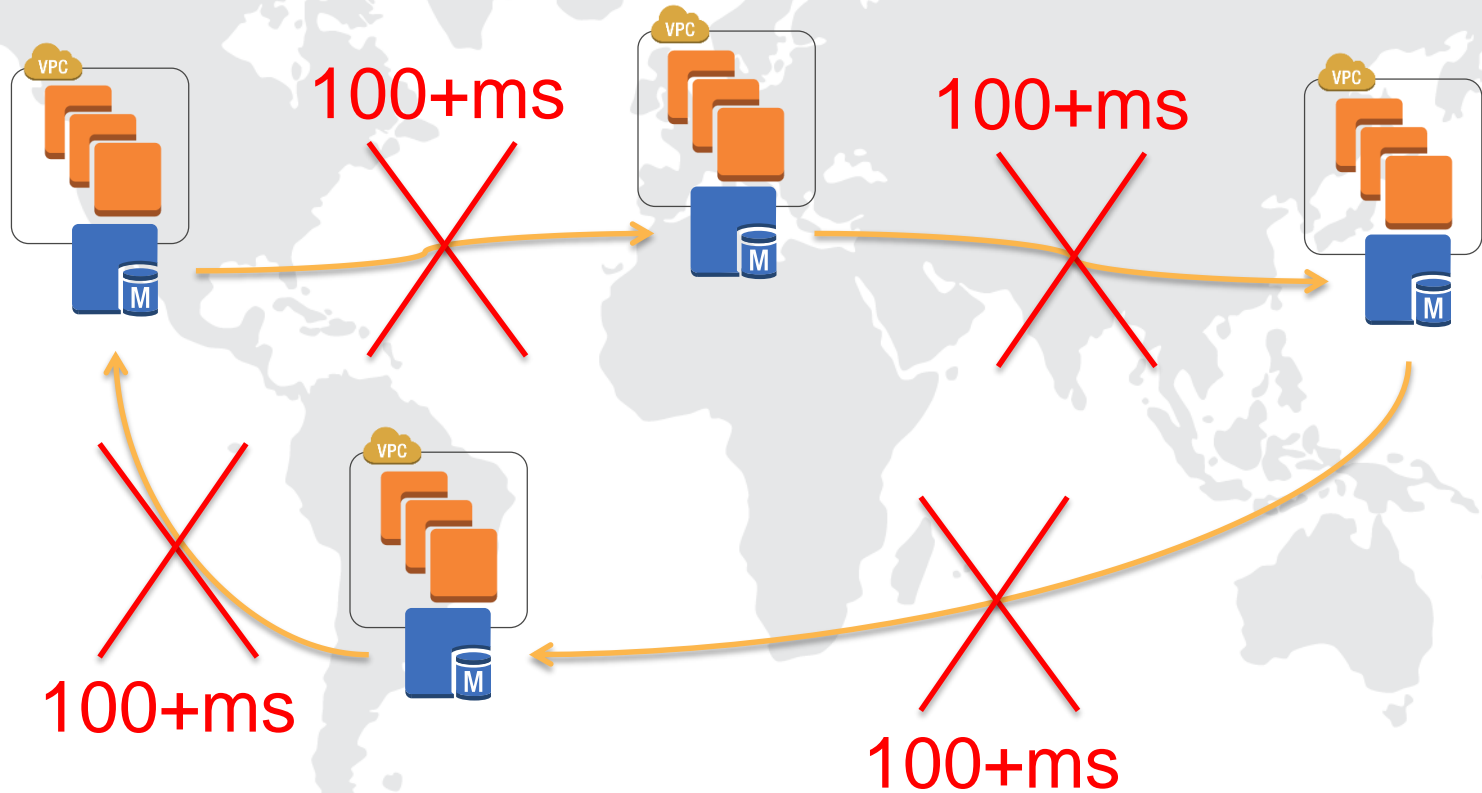
See also: <http://bit.ly/verizon-latency>

# 複数のAWSリージョンにサーバーを配置する アーキテクチャを考えてみる



リージョン間のデータ同期に  
データベースレプリケーション  
を使う？

# 100msを超えるレイテンシはリアルタイム ゲームに大きな影響を与える



一歩引いて考える: 多くのケースにおいて  
ユーザーは地理的に近いユーザーとのゲーム  
を好む

同じタイムゾーン

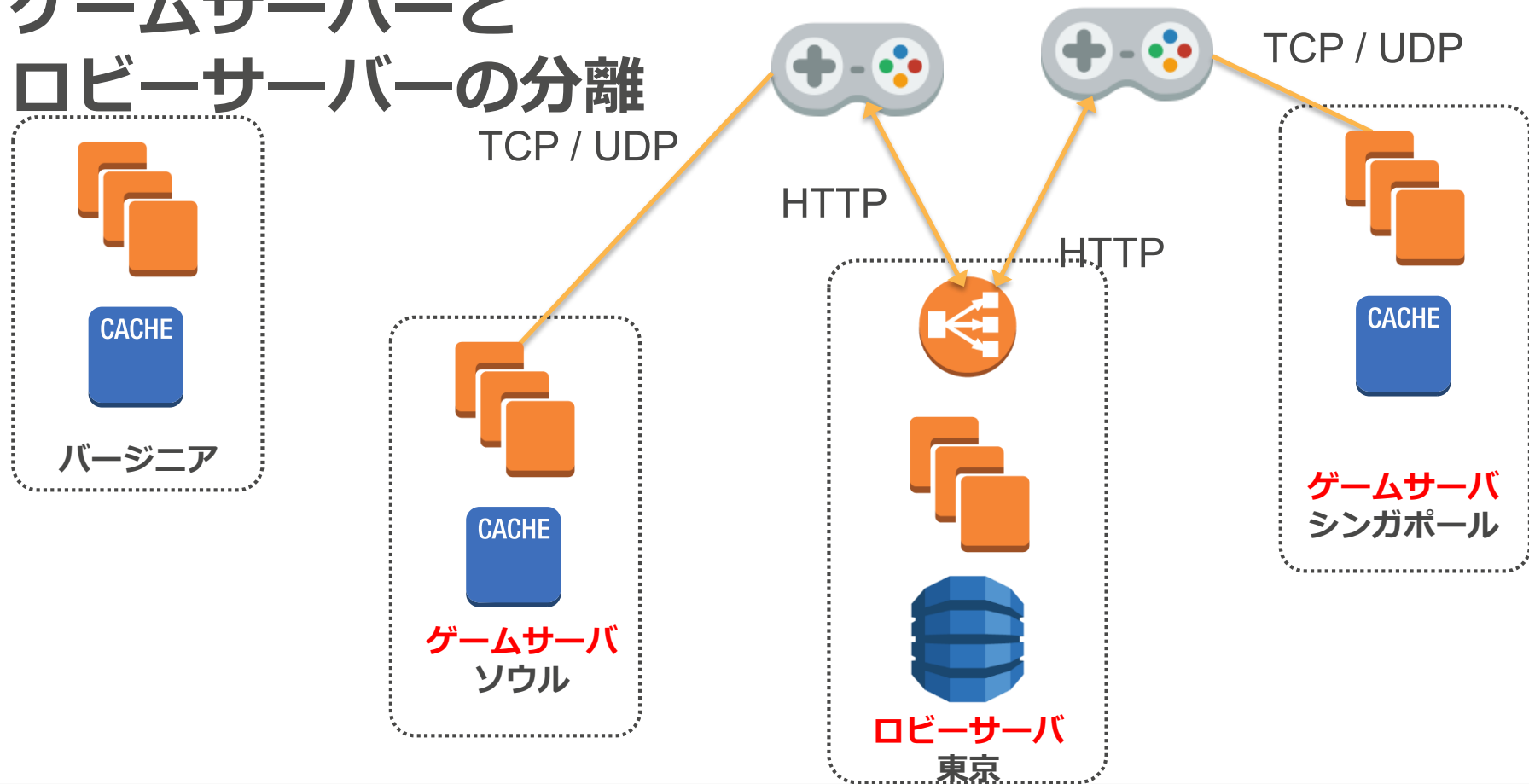
同じ言語

同じくらいのレイテンシ

**Solution:**

ゲームサーバーを  
地理的に分散配置する

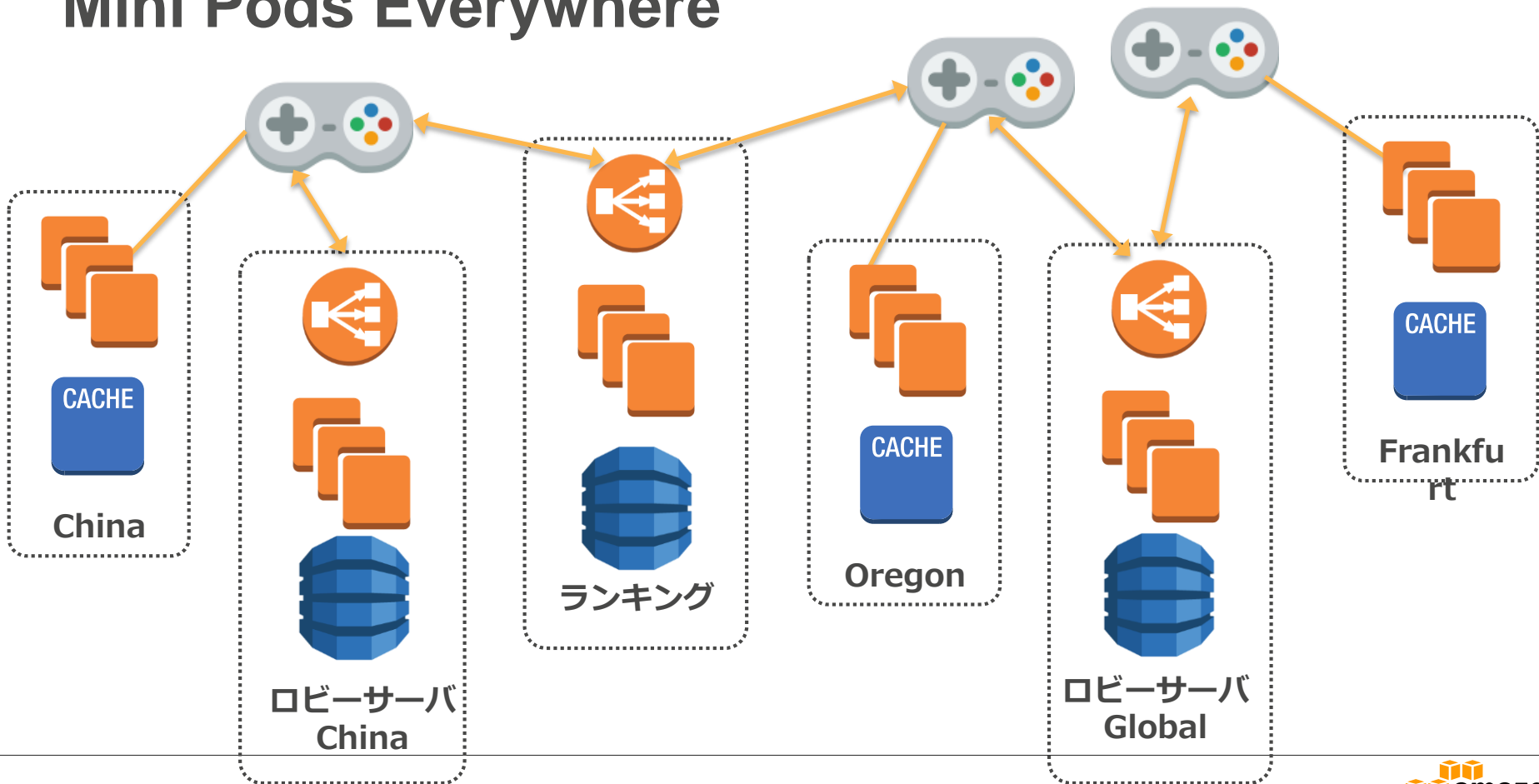
# ゲームサーバーと ロビーサーバーの分離





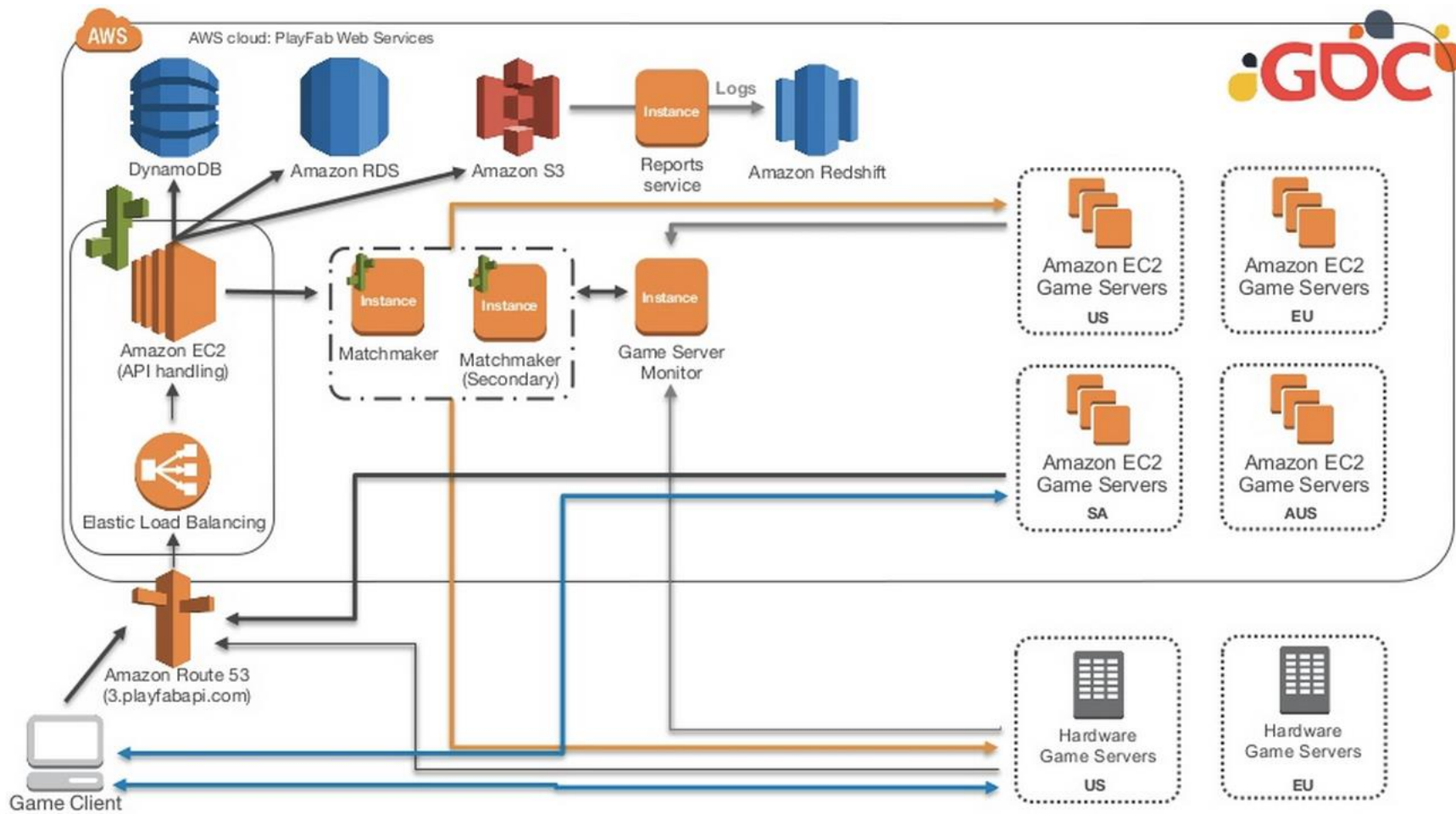
# China?

# Mini Pods Everywhere



# Playfab's architecture

## Architecture Overview



# Q&A



# オンラインセミナー資料の配置場所

## AWS クラウドサービス活用資料集

- 「AWS 資料」で検索
- <http://aws.amazon.com/jp/aws-jp-introduction/>

### 日本語資料のカテゴリー一覧

本資料集では、この利便性を皆様にご利用いただけるよう、トレーニング、ソリューション/事例、プロダクト別、セキュリティ・コンプライアンス、その他という5つのカテゴリーで資料をご用意しております。



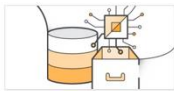
トレーニング資料

はじめてAWSをご利用いただくお客様向けに、AWS の概要、アカウント作成に関するご案内をいたします。



ソリューション・事例紹介資料

実際に他のお客様がどのようにAWS をご利用いただいているかをご覧いただける参考資料をご用意いただけます。



製品・サービス別資料

無料オンラインセミナー「AWS Black Belt Tech Webinar」や各種セミナーで紹介された、ソリューションアーキテクトによる各サービスの解説資料をご用意いただけます。

## AWS Solutions Architect ブログ

- 最新の情報、セミナー中のQ&A等が掲載されています
- <http://aws.typepad.com/sajp/>

# 公式Twitter/Facebook AWSの最新情報をお届けします



@awscloud\_jp



検索



もしくは

<http://on.fb.me/1vR8yWm>

最新技術情報、イベント情報、お役立ち情報、  
お得なキャンペーン情報などを日々更新しています！

# AWSの導入、お問い合わせのご相談

AWSクラウド導入に関するご質問、お見積り、資料請求をご希望のお客様は以下のリンクよりお気軽にご相談ください

<https://aws.amazon.com/jp/contact-us/aws-sales/>

お問い合わせ

日本担当チームへのお問い合わせ >

関連リンク

フォーラム

## 日本担当チームへのお問い合わせ

AWS クラウド導入に関するご質問、お見積り、資料請求をご希望のお客様は、以下のフォームよりお気軽にご相談ください。平日営業時間内に日本オフィス担当者よりご連絡させていただきます。

※ご請求金額またはアカウントに関する質問は[こちらからお問い合わせください](#)。  
※Amazon.com または Kindle のサポートに問い合わせは[こちらからお問い合わせください](#)。

アスタリスク（\*）は必須情報となります。

姓\*

名\*

※「AWS お問い合わせ」で検索してください

# AWS Black Belt Online Seminar



## 今後の配信予定

- 12月21日(水) **18:00-19:00** Amazon Workspaces

## 申し込みサイト

<https://aws.amazon.com/jp/about-aws/events/webinars/>

(もしくは「AWS イベント」で検索)





ご参加ありがとうございました

