

コンテナと サーバーレスの使い分け

荒木 靖宏

アマゾン ウェブ サービス ジャパン合同会社
プリンシパル ソリューションアーキテクト
コンテナスペシャリスト

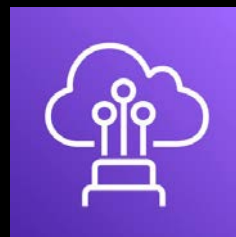
荒木靖宏 です。どうぞよろしく



AWSコンテナスペシャリスト
プリンシパルソリューションアーキテクト

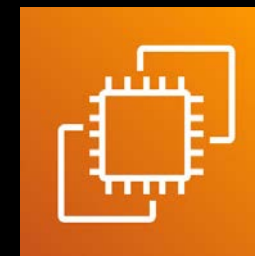
2011年からAWSのソリューションアーキテクトです

10年間好きなサービスは AWS Direct Connect と Spot Instance



このセッションの対象者

- 仮想サーバ、サーバーレス、コンテナについての知識はあるものの、いざサービスを作る段階になって、全体設計への適用に悩んでいる方
- サーバーレスやコンテナということ意識せずに作っていたシステムを刷新したいが何から手をつけていいのかわからない方



このセッションの進め方

コンテナとサーバーレスを知る

Webアプリケーションで
コンテナとサーバーレスをそれぞれ使えるようになる

次の一手：
コンテナとサーバーレスのハイブリッドシステム

コンテナとサーバーレスを知る

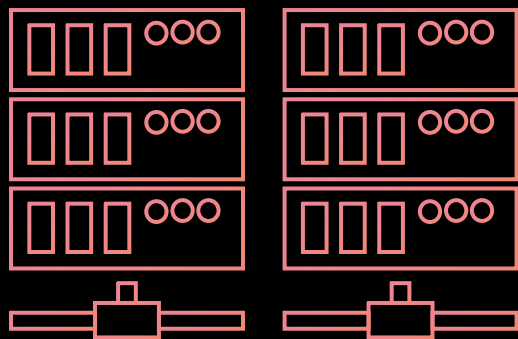


Amazon Simple Storage Service (Amazon S3)

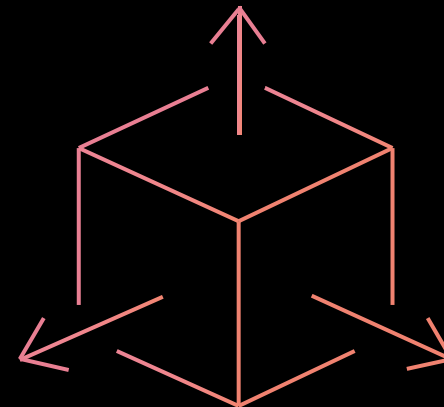


Amazon DynamoDB

マネージドサービスに共通の特徴



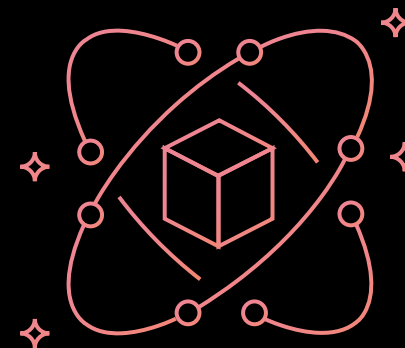
サーバ管理が不要



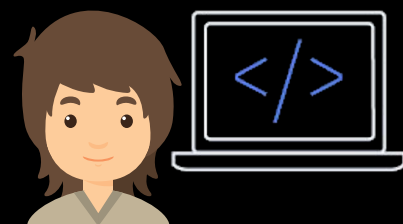
柔軟なスケーリング



アイドル時のリソース確保が不要



組み込まれた高可用性



アプリケーションコード

Code

Middleware

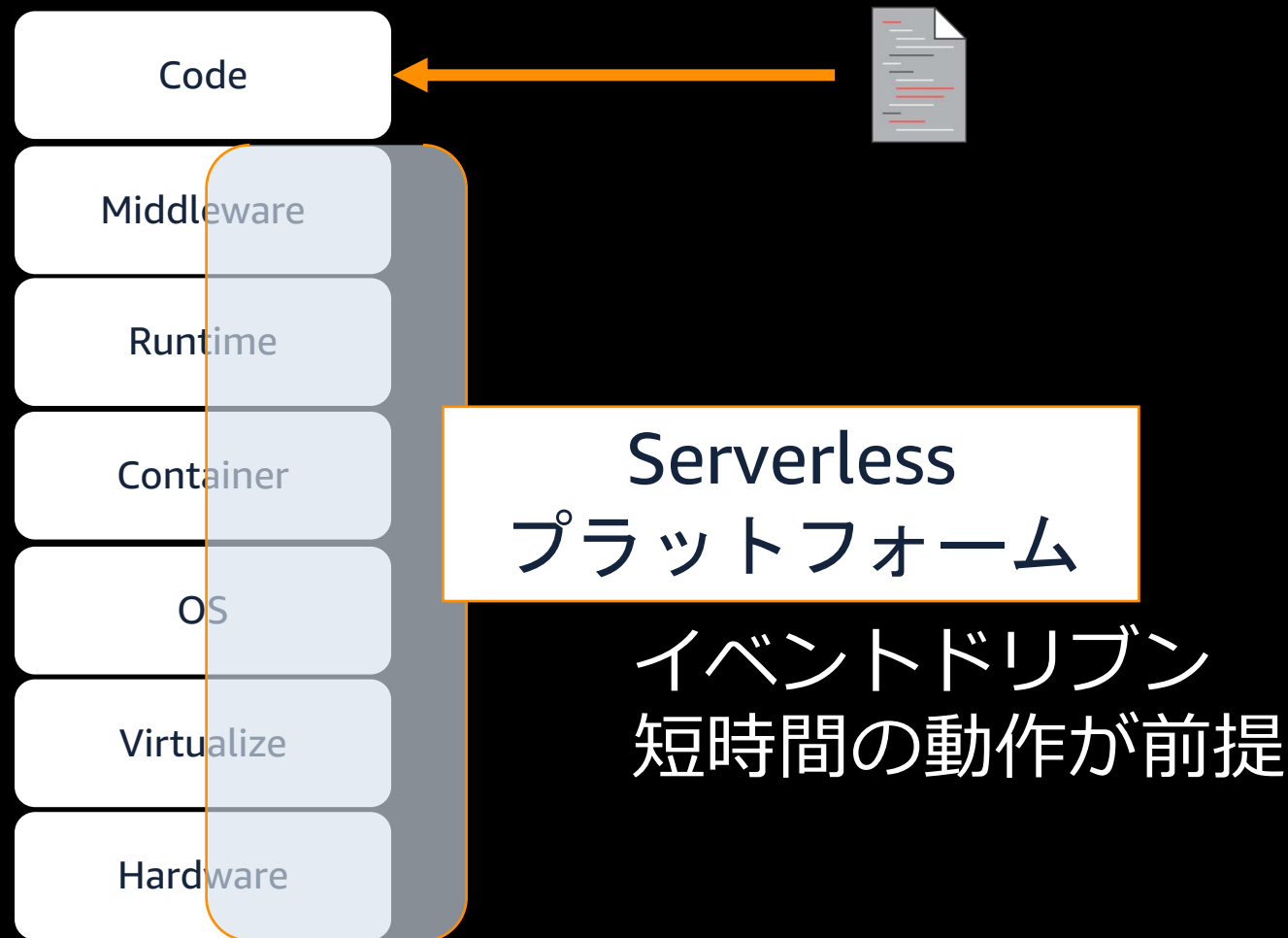
Runtime

Container

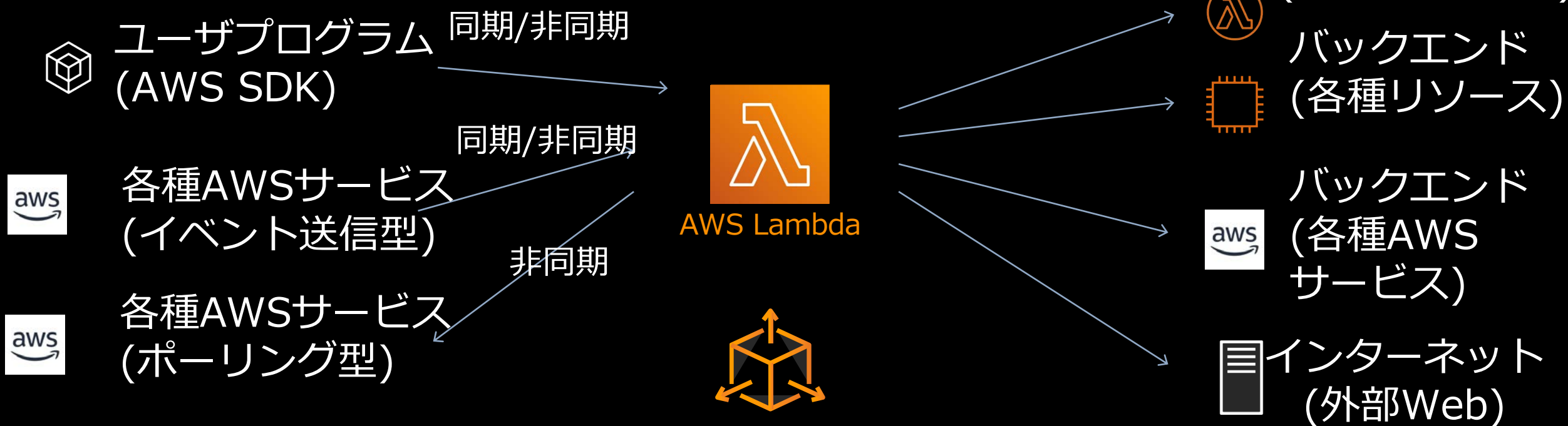
OS

Virtualize

Hardware



AWS Lambda

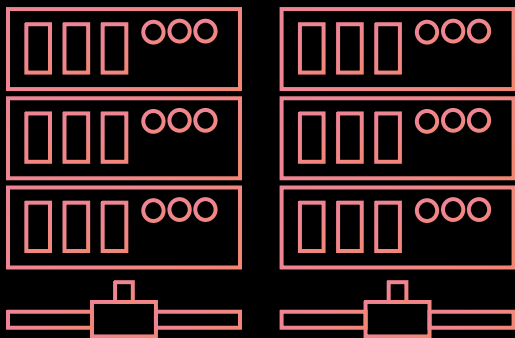


Java/Go/Node.js/Python/.Net /
C#/Ruby and more

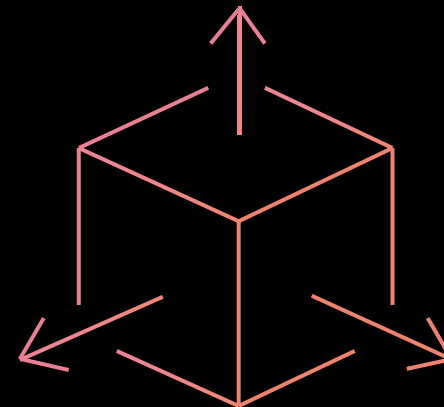
自動スケーリングと組み込みの耐障害

性
「関数」を様々なイベントをトリガーに実行

AWS Lambda によって得るもの



サーバ管理が不要

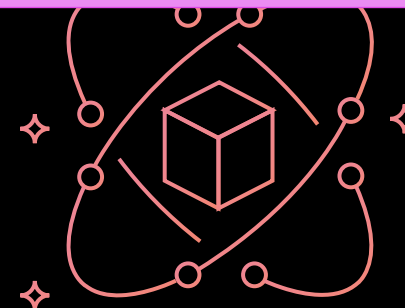


柔軟なスケーリング

利用は簡単。ただし制約も厳しい



アイドル時のリソース確保が不要



組み込まれた高可用性

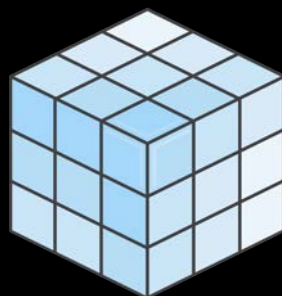
コード



ランタイム



ライブラリ



コンテナ
プラットフォーム

長時間動作する
プロセスとして動作
制約が少ない

Code

Middleware

Runtime

Container

OS

Virtualize

Hardware

Serverless
プラットフォーム

イベントドリブン
短時間の動作が前提
制約が厳しい

コンテナはあらゆる場所で行可能

AWS Region

Serverless



AWS Fargate

EC2 options



Amazon EC2

Edge and 5G



AWS
Local
Zones



AWS
Wavelength

On-premises



AWS
Outposts



EKS
Anywhere
ECS
Anywhere

コンテナオーケストレータが提供する機能



<https://kubernetes.io/>



宣言的な設定が可能

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: hello-k8s
  labels:
    app: hello-k8s
  namespace: hamaan
spec:
  replicas: 3
  selector:
    matchLabels:
      app: hello-k8s
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
    type: RollingUpdate
  template:
    metadata:
      labels:
        app: hello-k8s
```

```
spec:
  containers:
    - image: paulbouwer/hello-kubernetes:1.7
      imagePullPolicy: Always
      name: hello-kubernetes
      ports:
        - containerPort: 8080
```

```
---
apiVersion: v1
kind: Service
apiVersion: v1
metadata:
  name: hello-k8s
spec:
  ports:
    - port: 80
      targetPort: 8080
  selector:
    app: hello-k8s
  type: LoadBalancer
```

サーバーレス



AWS Lambda

アプリケーションコード

アプリケーション
実行環境

開発者が担当

AWSが担当

マネージドコンテナ

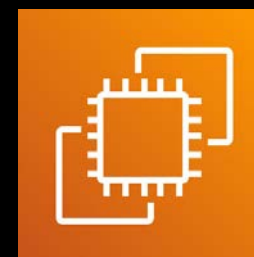


AWS Fargate

アプリケーションコード
コンテナイメージ
コンテナレジストリ

コンテナ
実行環境

仮想サーバー

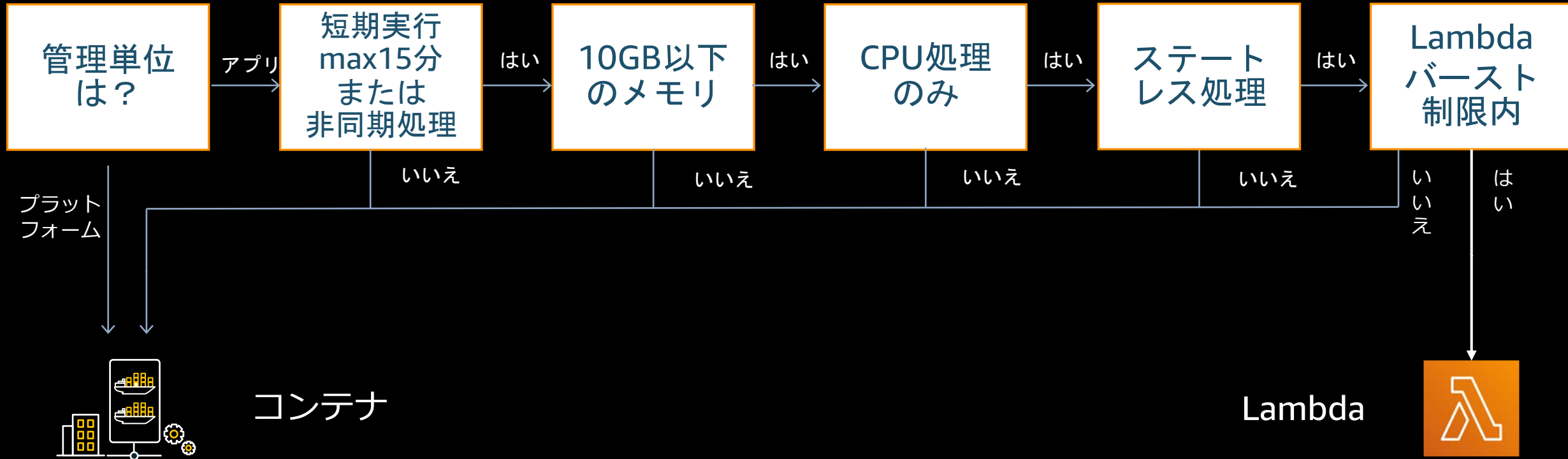


Amazon EC2

アプリケーションコード
AMI (+ User data)
• サーバ/OSの設定
• ミドルウェア
• ランタイム
etc...

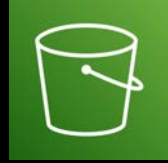
仮想マシン基盤

コンテナと Lambda におけるデシジョンツリー



Webアプリケーションで
コンテナとサーバーレスを
それぞれ使えるようになる

ステートはマネージドサービスにもたせる



Amazon Simple Storage
Service (Amazon S3)



Amazon Elastic
File System
(Amazon EFS)



Amazon Relational
Database Service
(Amazon RDS)



Amazon
ElastiCache

一般的な3-tier WEB アプリケーション

プレゼンテーション層

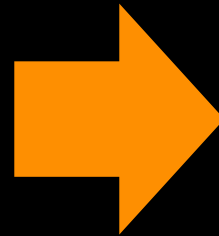
アプリケーション層

永続層



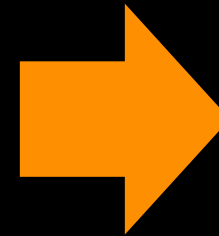
ウェブ
サーバー

Apache、
NGINX等



アプリケーション
サーバー

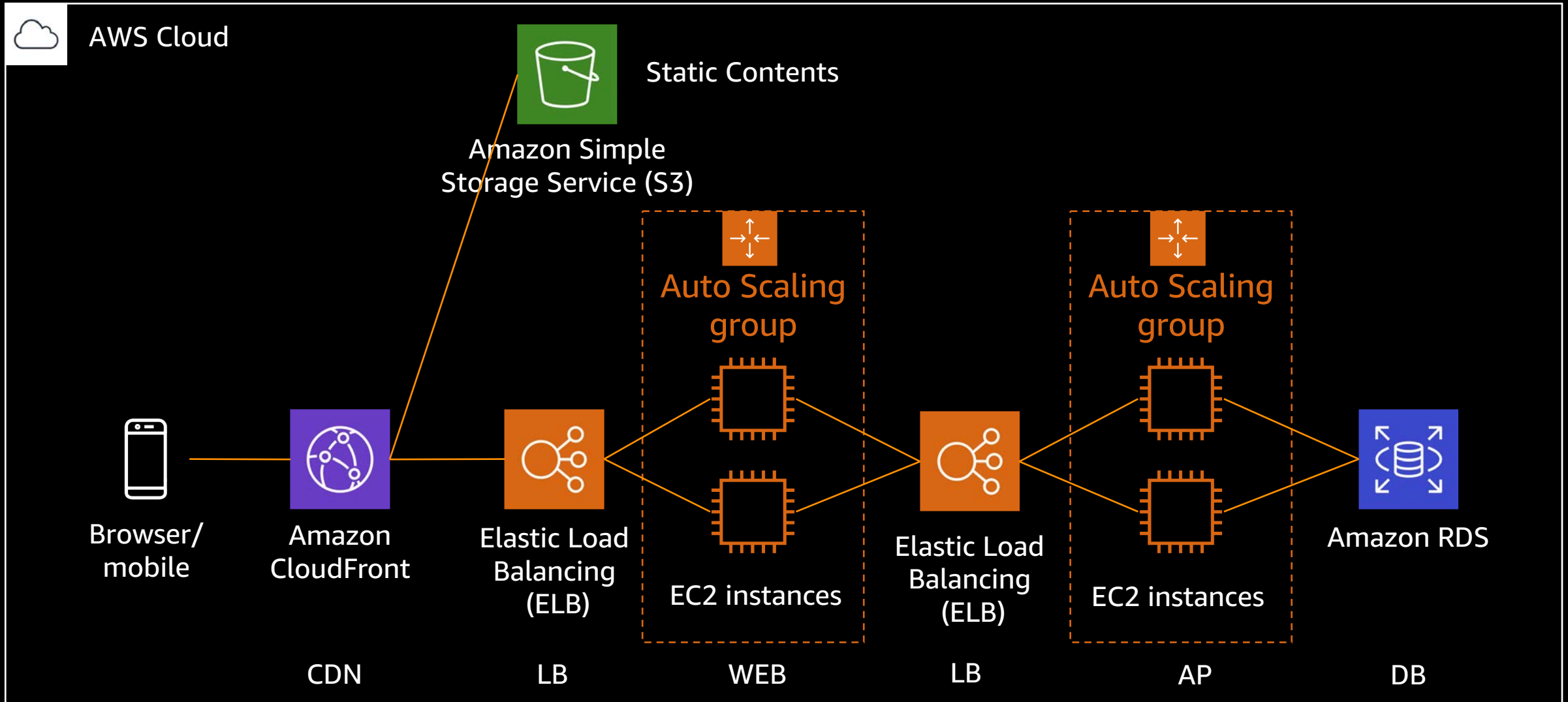
Tomcat、
Weblogic等



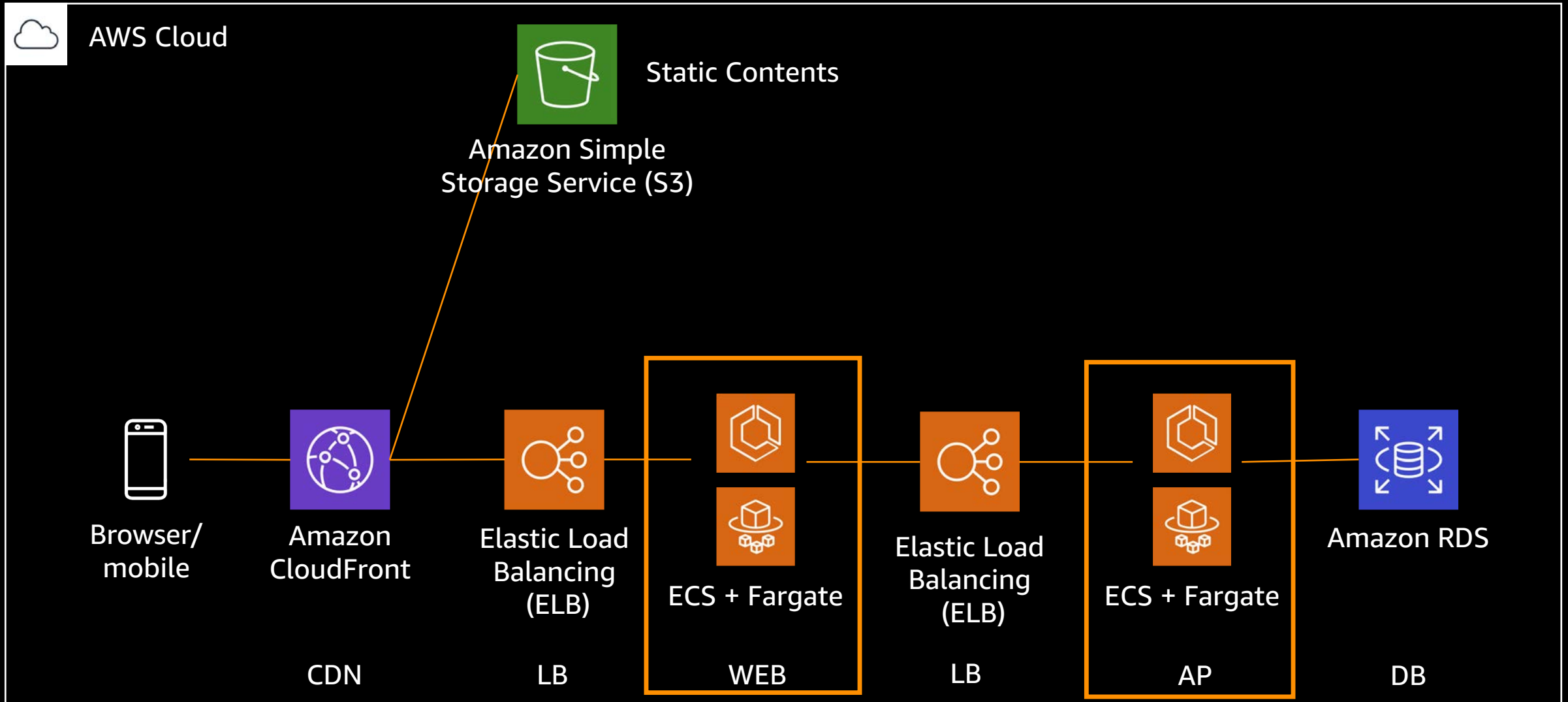
データベース
サーバー

Oracle、
MySQL等

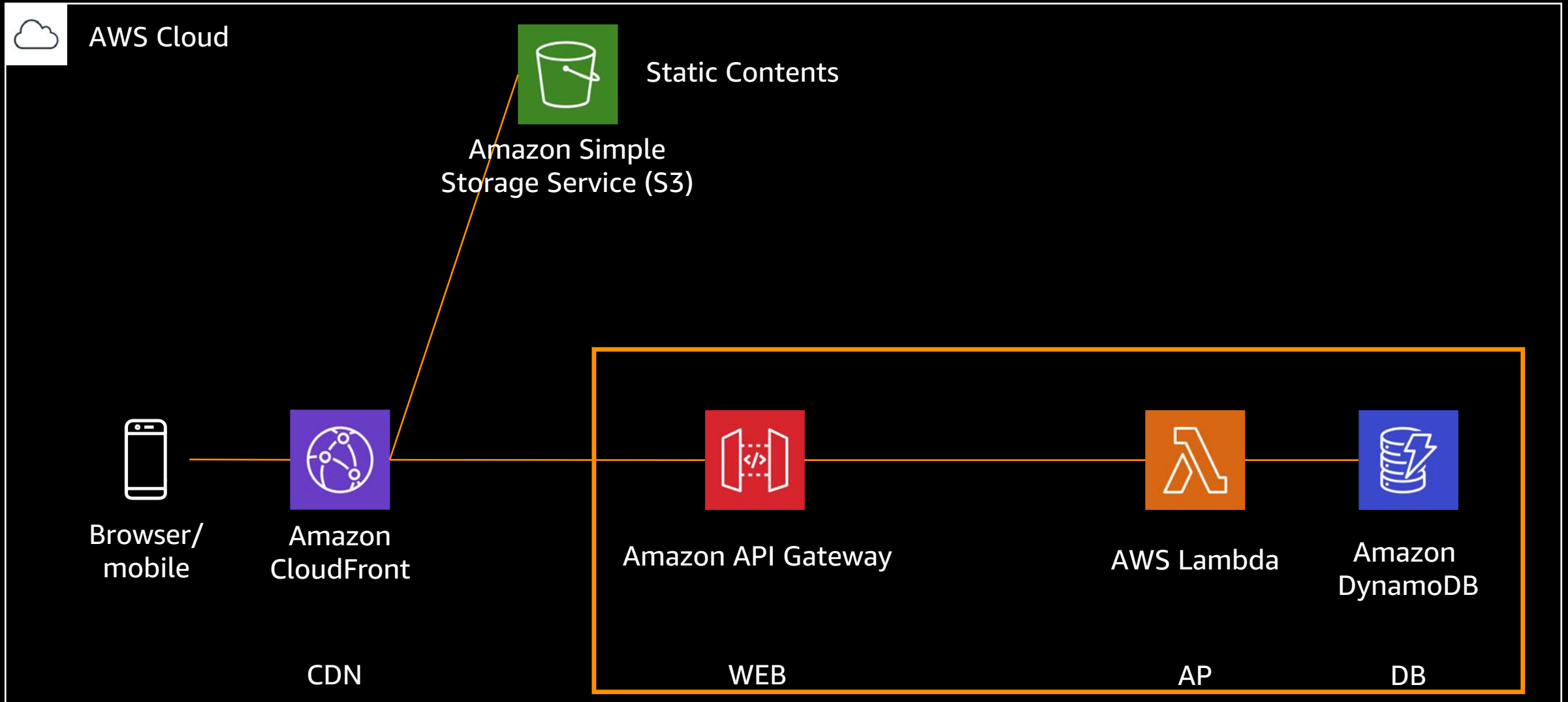
EC2(仮想サーバー)での構成例



コンテナでの構成例



サーバーレスでの構成例

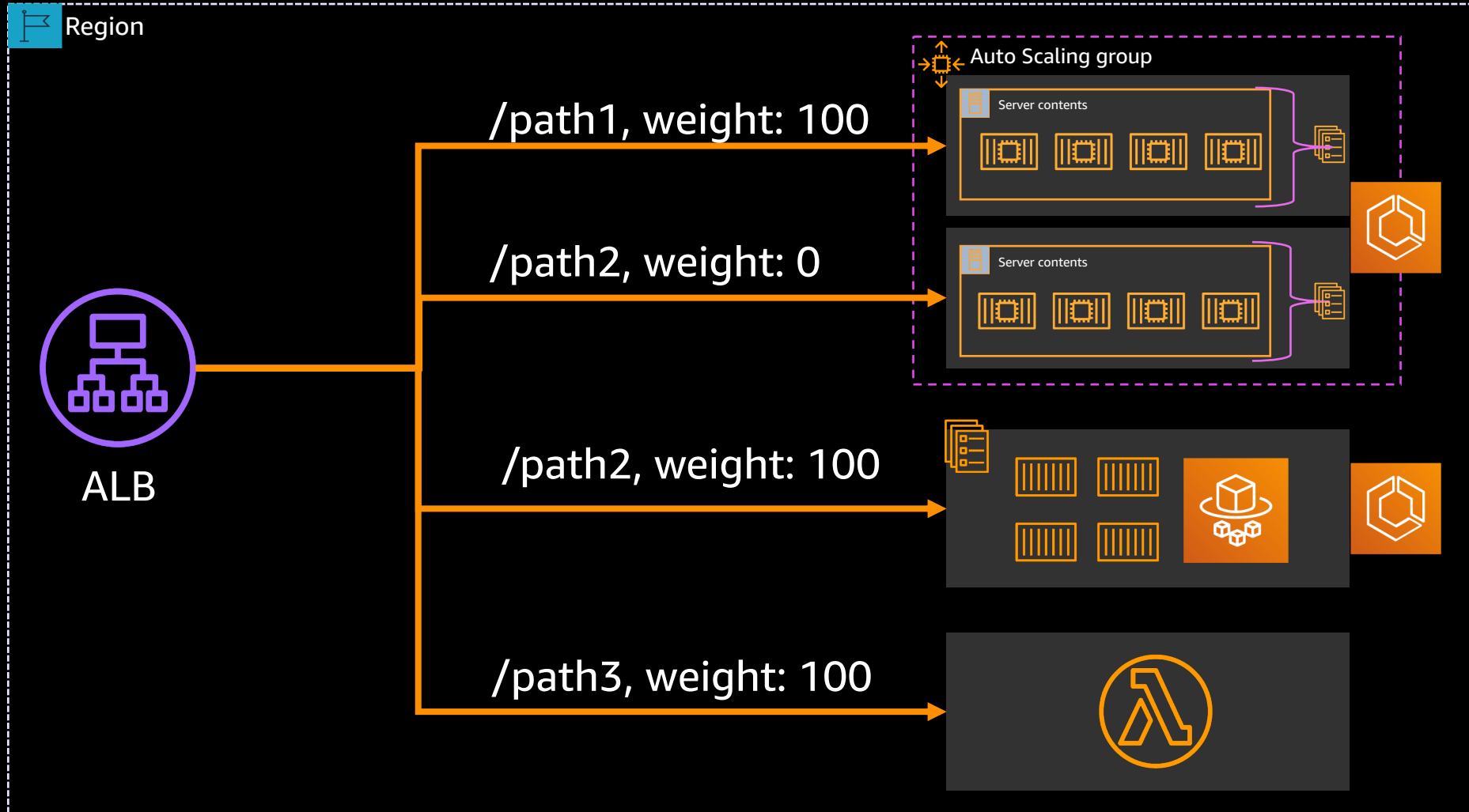




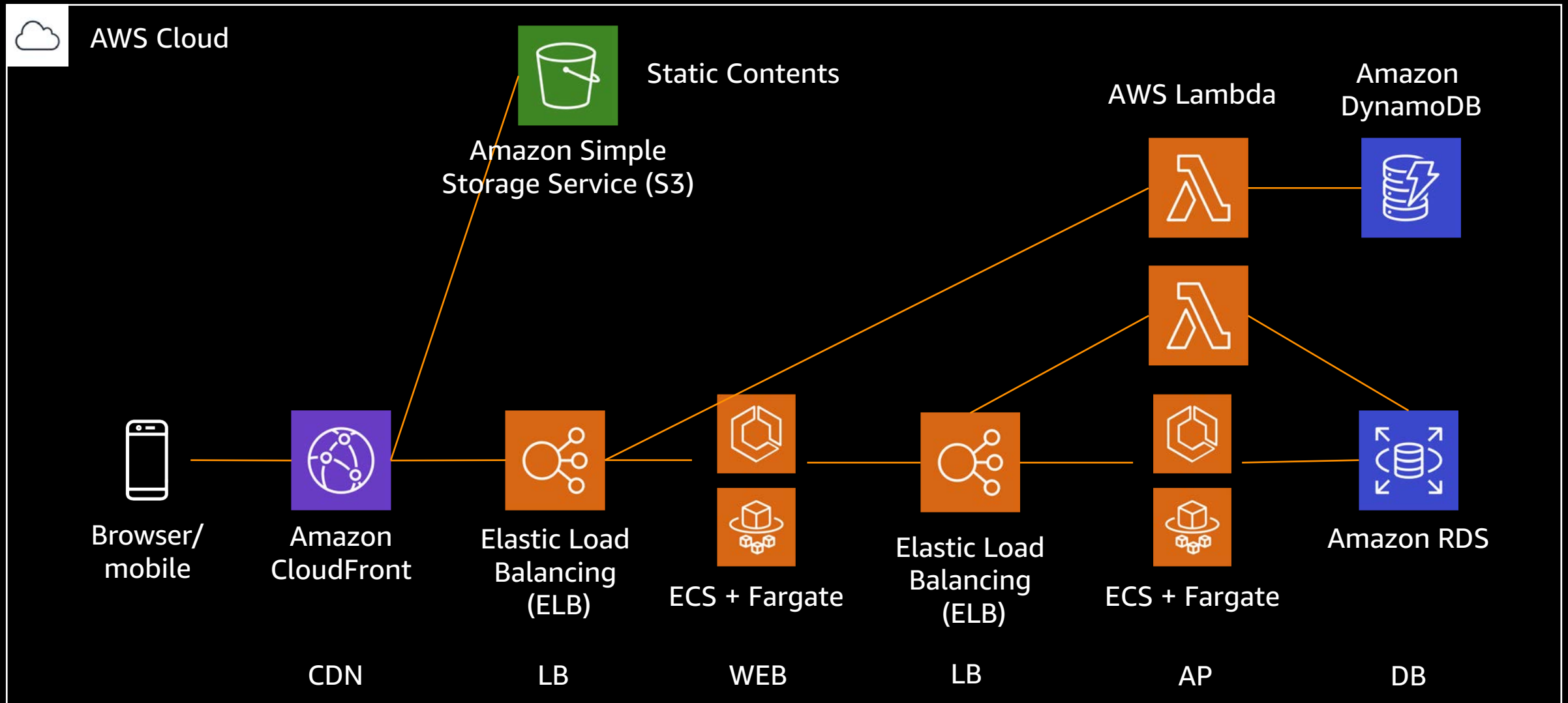
次の一手： コンテナとサーバーレスのハイブリッド

ALBをつかったサービス分割

ALBは重みをつけてターゲットグループを扱える

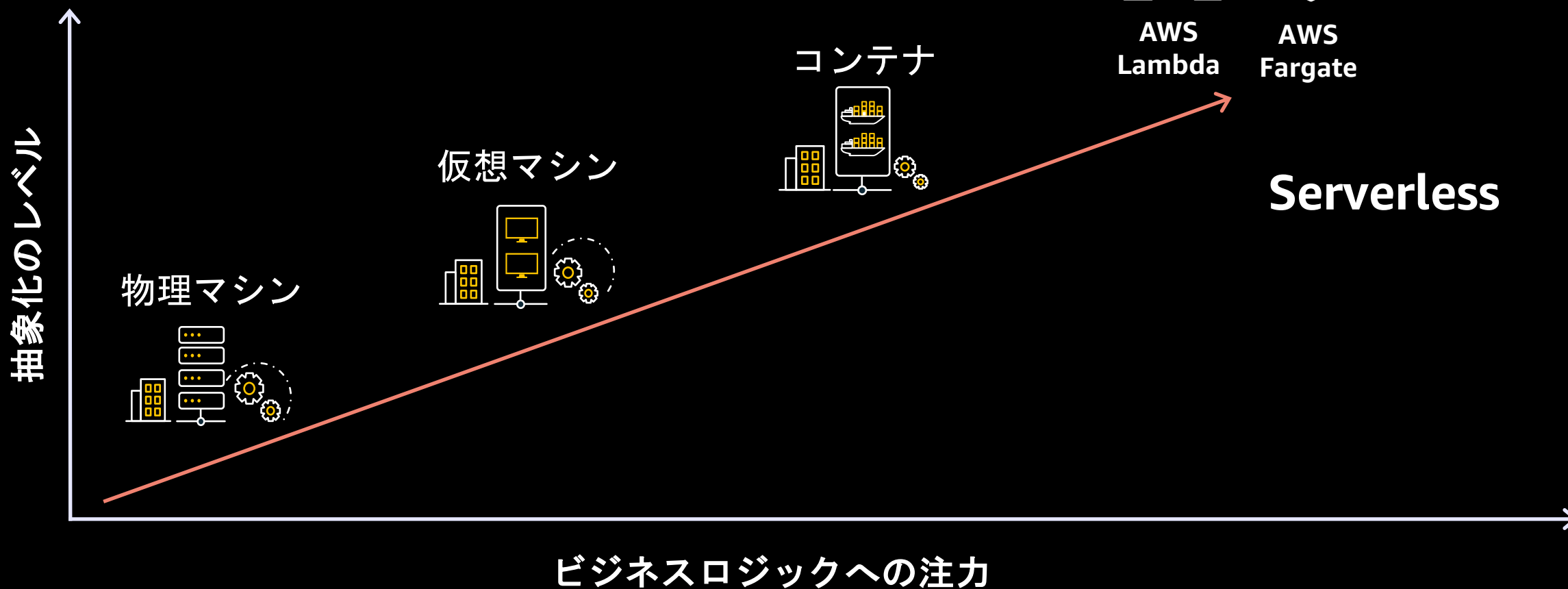


ハイブリッド構成例



まとめ

78%の新規アプリがコンテナまたはサーバーレス
58%がコンテナ、21%がサーバーレス



Source: [Cloud Native Application Security Report | Snyk](#)



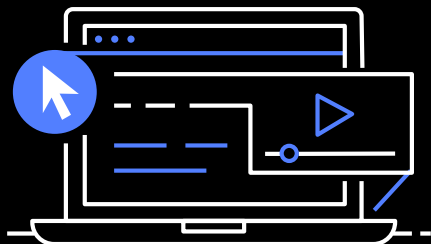
サービスを構成するビルディングブロック (一部)

Network  Elastic Load Balancing	Storage  Amazon S3  Amazon EFS	Database  Amazon DynamoDB  Amazon RDS  Amazon Aurora
API Proxy  Amazon API Gateway  AWS AppSync	Streaming  Amazon Kinesis	
Authentication, WAF  Amazon Cognito  AWS WAF	Monitoring and Debugging  Amazon CloudWatch  AWS X-Ray	Messaging and Queues  Amazon SQS  Amazon SNS

まとめにかえて

コンテナとサーバーレス、適材適所で。
AWSのマネージドサービスを活用しましょう。

AWS デジタルトレーニング



学習方法を柔軟に選ぶことができます。

550以上の無料デジタルトレーニングでクラウドのスキルを身につけることができ、クラスルームトレーニングでさらに深く学ぶこともできます。

おすすめのコース

- **[AWS Cloud Practitioner Essentials \(Japanese\)](#)**

このコースは、特定の技術領域ではなく、アマゾン ウェブ サービス (AWS) クラウドを全体的に理解したい方を対象としています。受講者は、AWS クラウドの概念、AWS のサービス、セキュリティ、アーキテクチャ、料金、サポートについて学習し、AWS クラウドについての知識を深めます。このコースは、AWS Certified Cloud Practitioner 試験の準備にも役立ちます。

- **[Getting Started with AWS Security, Identity, and Compliance \(Japanese\)](#)**

このコースでは、AWS のセキュリティテクノロジー、ユースケース、メリット、サービスの概要について説明します。AWS のセキュリティ、アイデンティティ、コンプライアンスのカテゴリにおけるさまざまなサービスを紹介します。このコースを修了すると、クラウド内のセキュリティの重要性を理解し、データの保護に使用する AWS のサービスを選択できるようになります。

- **[Moving to Managed Databases on AWS \(Japanese\)](#)**

リフトアンドシフト戦略は、クラウドを利用しようとする組織が一般的に採用する方法です。このコースでは、AWS ヒーローである Alex DeBrie が、アマゾン ウェブ サービス (AWS) においてセルフマネージドデータベースからフルマネージドデータベースソリューションに移行することのメリットについて説明します。また、AWS クラウドに移行する際に考慮すべき重要な要素についても学習します。最後に、この Moving to Managed Databases on AWS の 8 つのコースのカリキュラムの一環として、一般的なデータベースの移行に関するユースケースのコースをご紹介します。

AWS Builders Online Series に ご参加いただきありがとうございます

楽しんでいただけましたか? ぜひアンケートにご協力ください。
本日のイベントに関するご意見/ご感想や今後のイベントについてのご希望や改善のご提案などがございましたら、ぜひお聞かせください。



aws-apj-marketing@amazon.com



twitter.com/awscloud_jp



facebook.com/600986860012140



<https://www.youtube.com/user/AmazonWebServicesJP>



<https://www.linkedin.com/showcase/aws-careers/>



twitch.tv/aws

Thank you!

荒木 靖宏

アマゾン ウェブ サービス ジャパン合同会社
プリンシパル ソリューションアーキテクト
コンテナスペシャリスト

