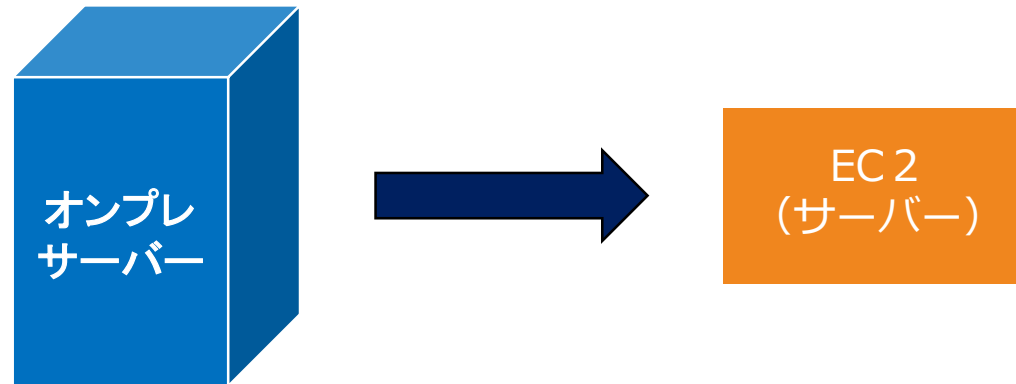


AWSの仕組み



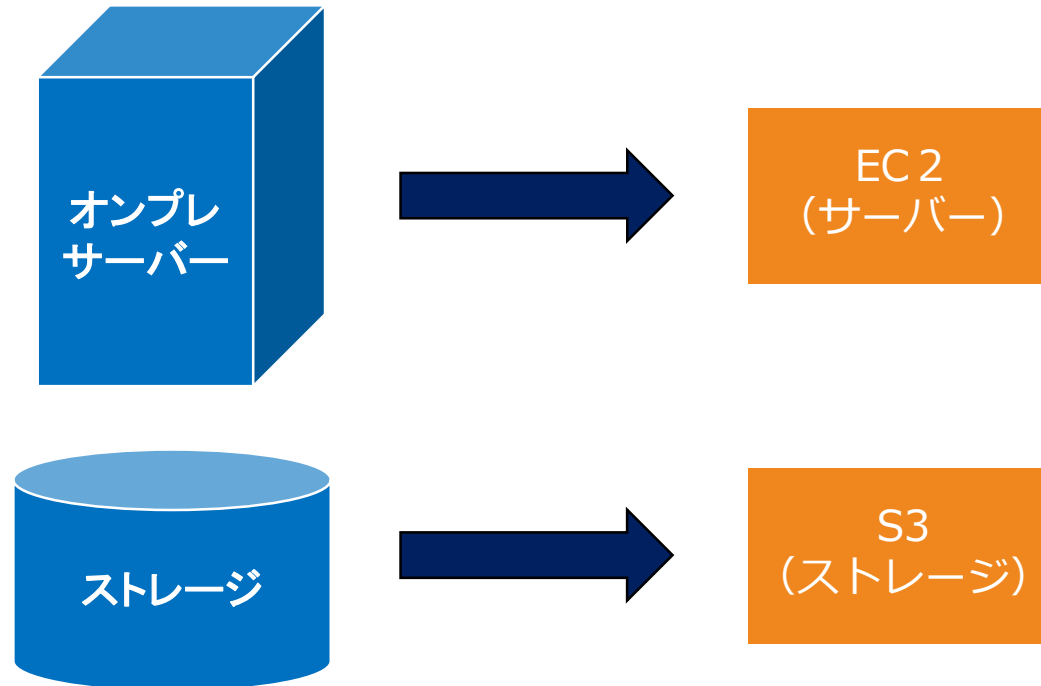
AWSの仕組み

インフラやアプリ開発に必要な機能がオンデマンドのパーツサービスとして提供されている



AWSの仕組み

インフラやアプリ開発に必要な機能がオンデマンドのパーツサービスとして提供されている

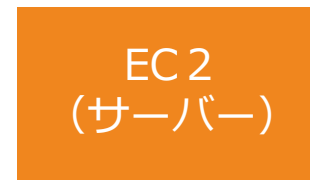


AWSの仕組み

サーバーを立ち上げるのに数十分で無料で今すぐにでも利用できることが大きな特徴



- ✓ 時間がかかる
- ✓ コストがかかる



- ✓ 数十分で立ち上がる
- ✓ 無料から利用可能



AWSの仕組み

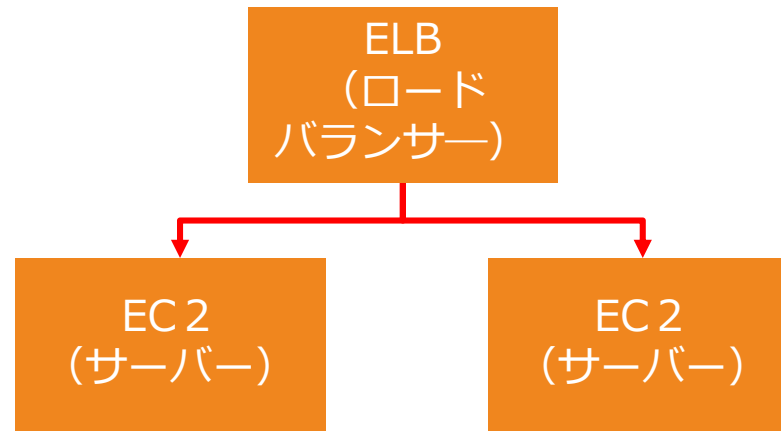
インフラ／システム機能をブロックパーツのようにオンライン上に組合わせて自分の好きな構成を実現する仕組み

EC2
(サーバー)



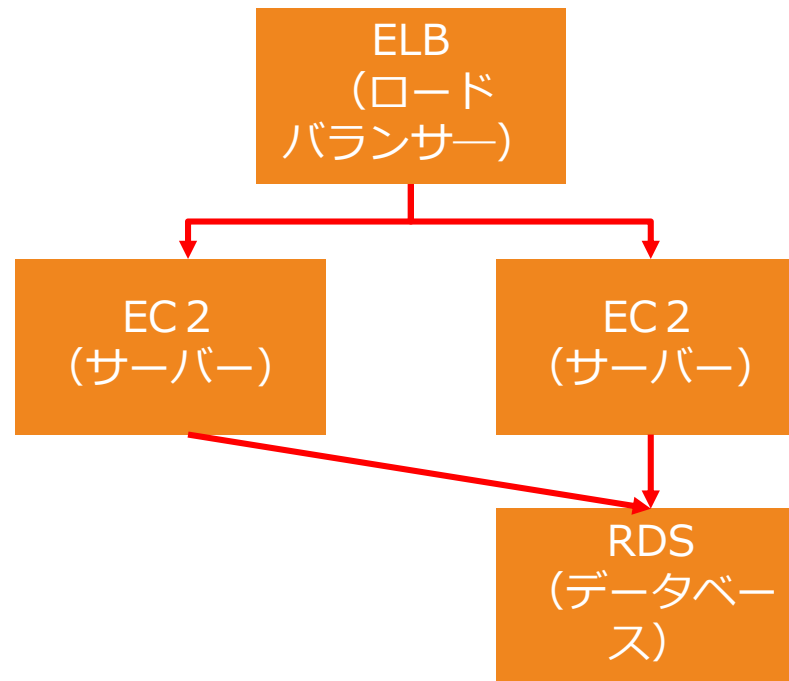
AWSの仕組み

インフラ／システム機能をブロックパーツのようにオンライン上に組合わせて自分の好きな構成を実現する仕組み



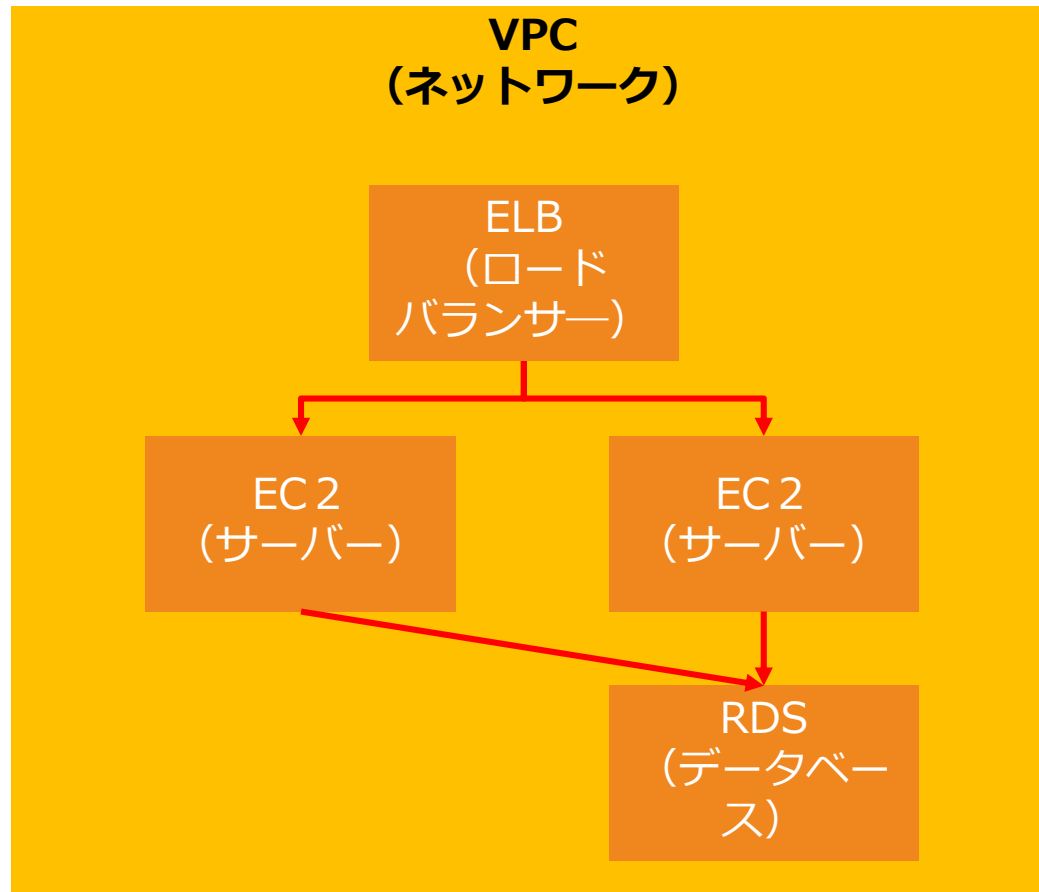
AWSの仕組み

インフラ／システム機能をブロックパーツのようにオンライン上に組合わせて自分の好きな構成を実現する仕組み



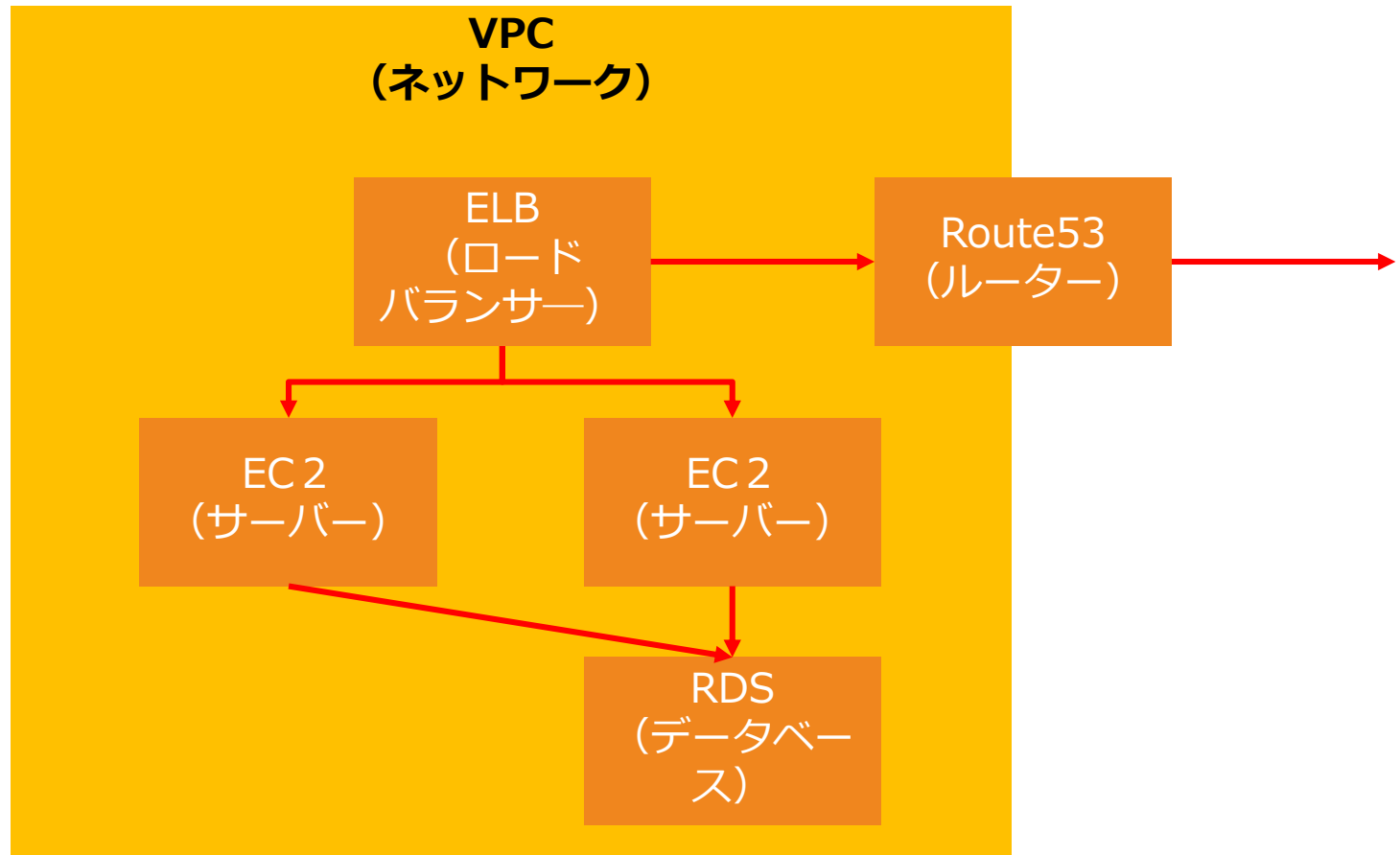
AWSの仕組み

インフラ/システム機能をブロックパーツのようにオンライン上に組合わせて自分の好きな構成を実現する仕組み



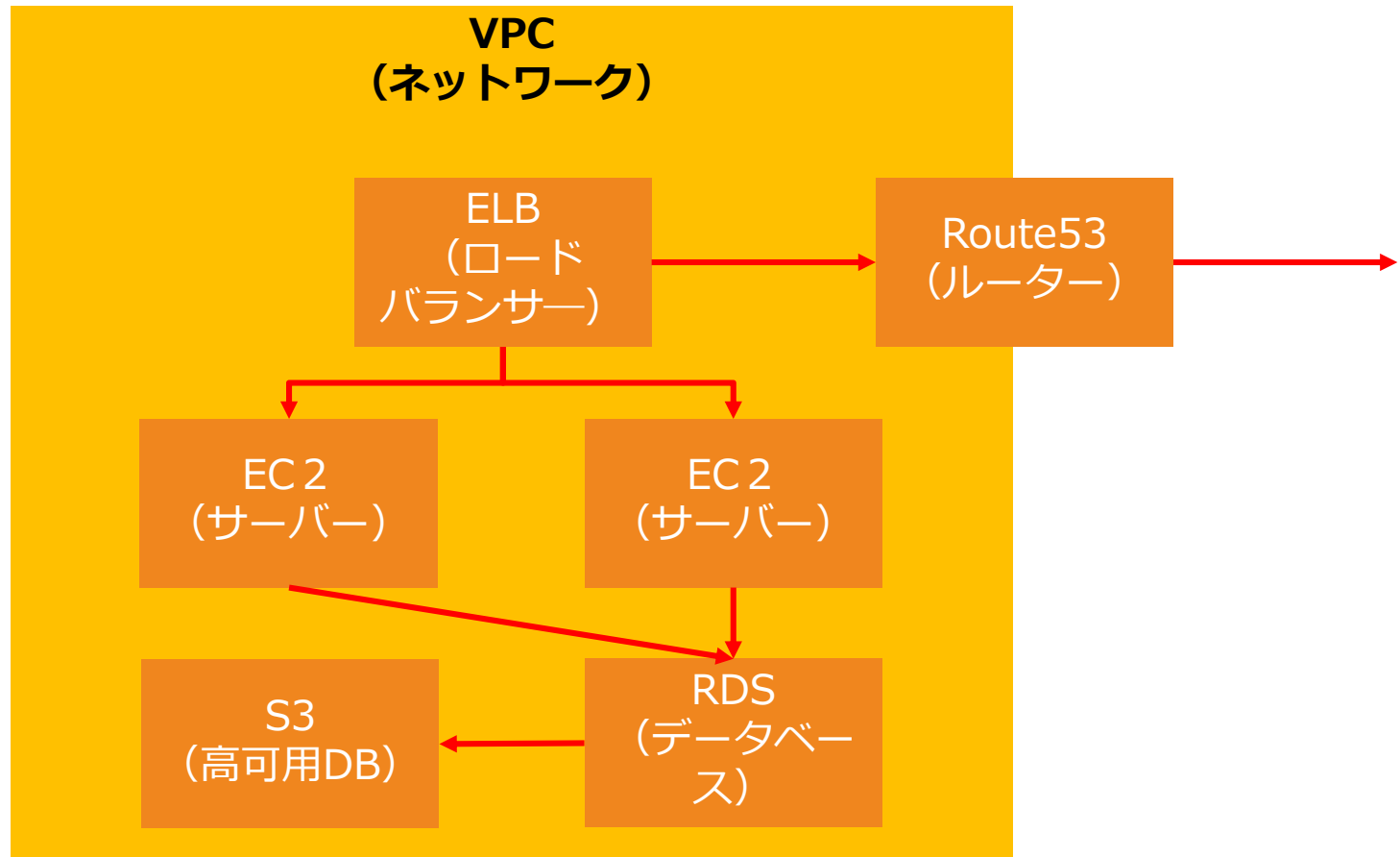
AWSの仕組み

インフラ/システム機能をブロックパーツのようにオンライン上に組合わせて自分の好きな構成を実現する仕組み



AWSの仕組み

インフラ/システム機能をブロックパーツのようにオンライン上に組合わせて自分の好きな構成を実現する仕組み



AWSの仕組み

AWSのサービスにはアンマネージド型とマネージド型が存在

アンマネージド型

- スケーリング／耐障害性／可用性
を利用者側で設定し管理する必要がある

【メリット】
設定が柔軟に可能

【デメリット】
管理が面倒

マネージド型

- スケーリング／耐障害性／可用性
がサービスに組み込まれており
AWS側で管理されている

【メリット】
管理が楽

【デメリット】
設定が限定的



AWSの仕組み

AWSのサービスにはアンマネージド型とマネージド型が存在

アンマネージド型

EC2
(サーバー)

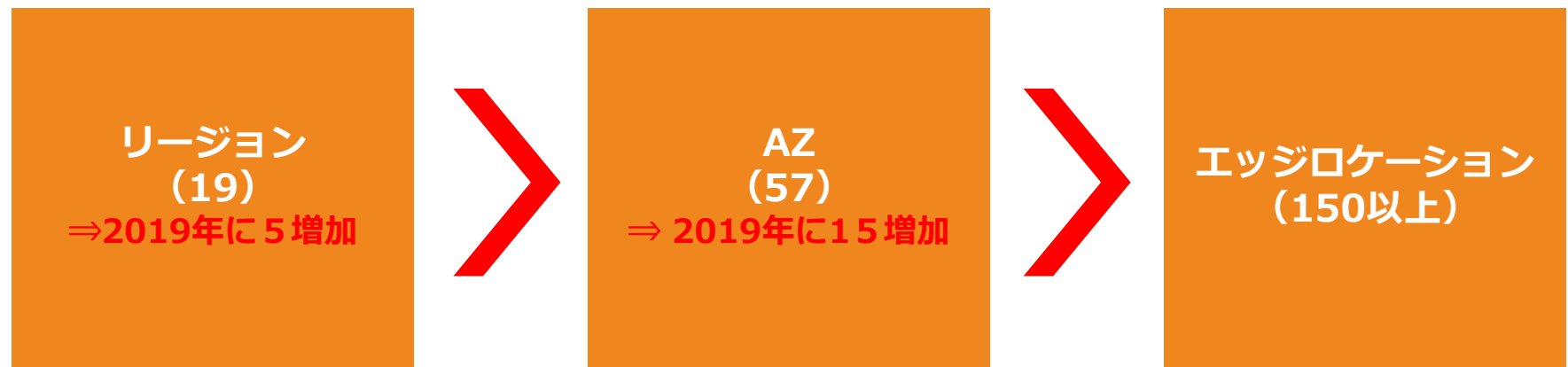
マネージド型

Route53
(DNS)



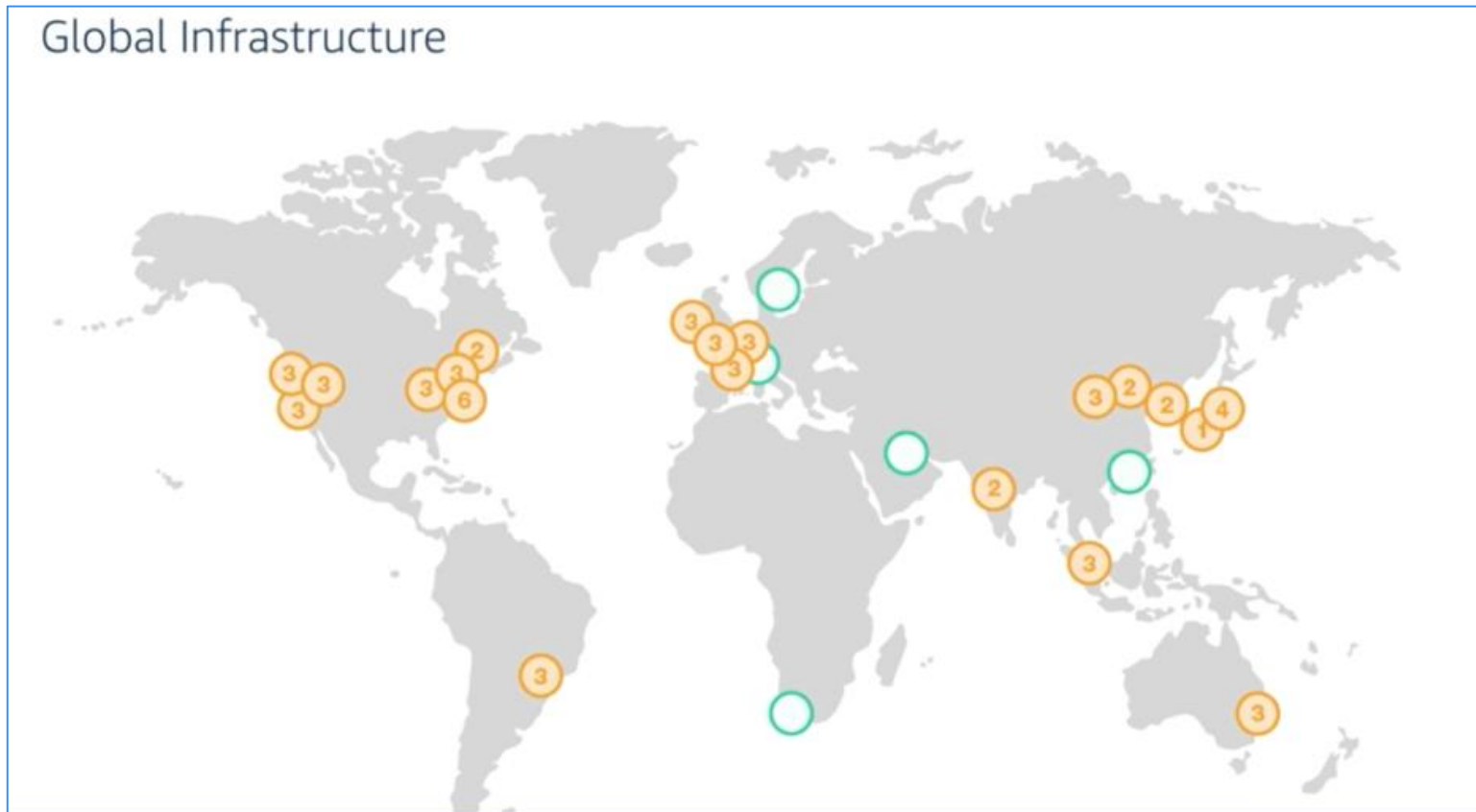
AWSのグローバルレインフラ構成

リージョンとアベイラビリティゾーン (AZ) とエッジロケーションの3つの区分 (数は2019年1月現在)



リージョン

リージョンは国や地域における地理的に隔離されたAWS拠点



リージョン

日本には東京と大阪の2つのリージョンが設置されている

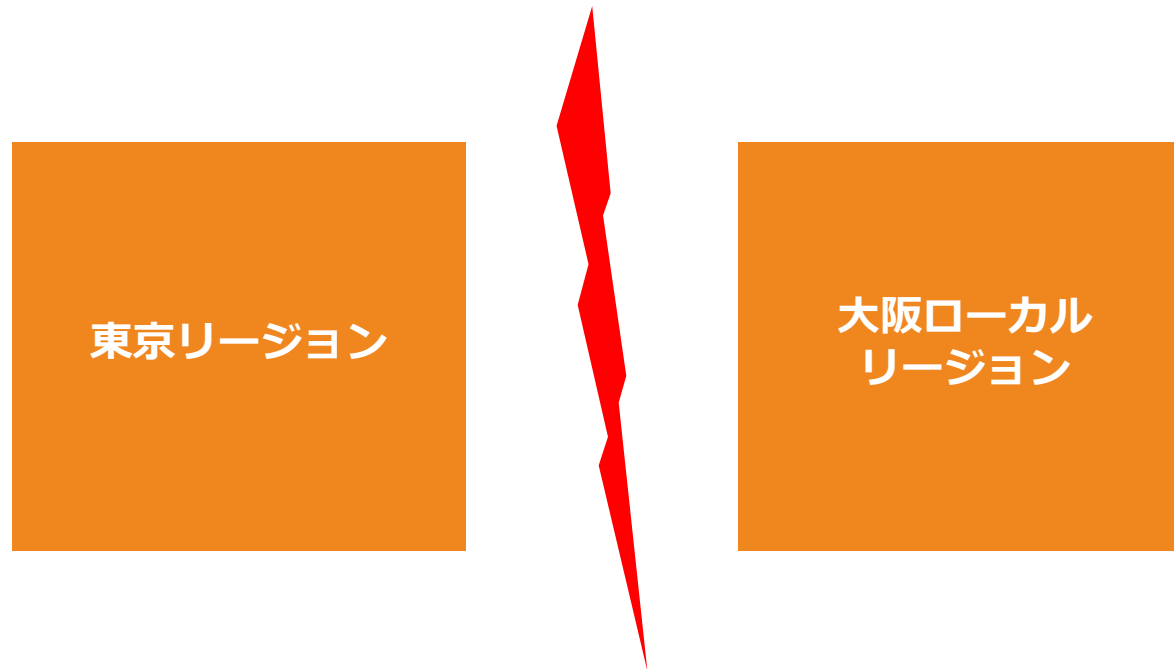
東京リージョン

大阪ローカル
リージョン



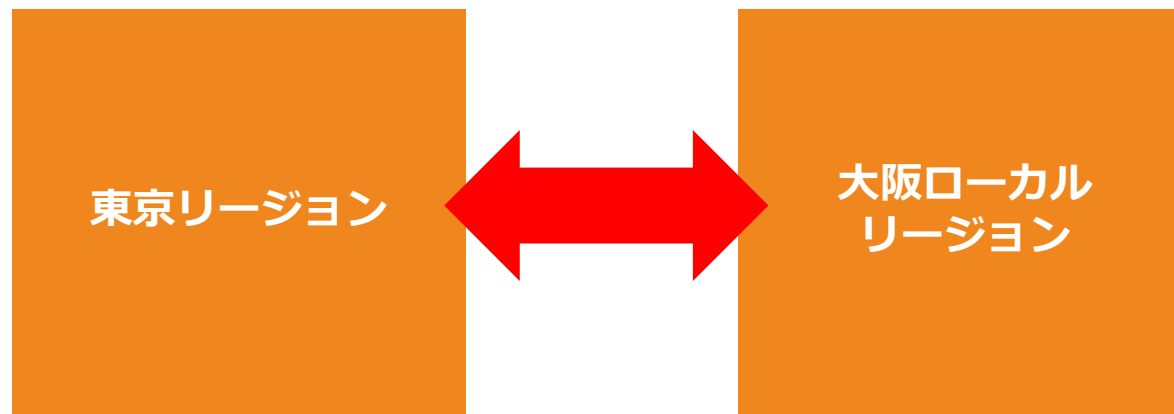
リージョン

リージョンとリージョンは物理的に独立したインフラ拠点



リージョン

ただし、隣接リージョン間は広帯域の専用ネットワークで接続されている



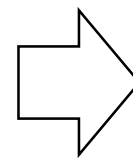
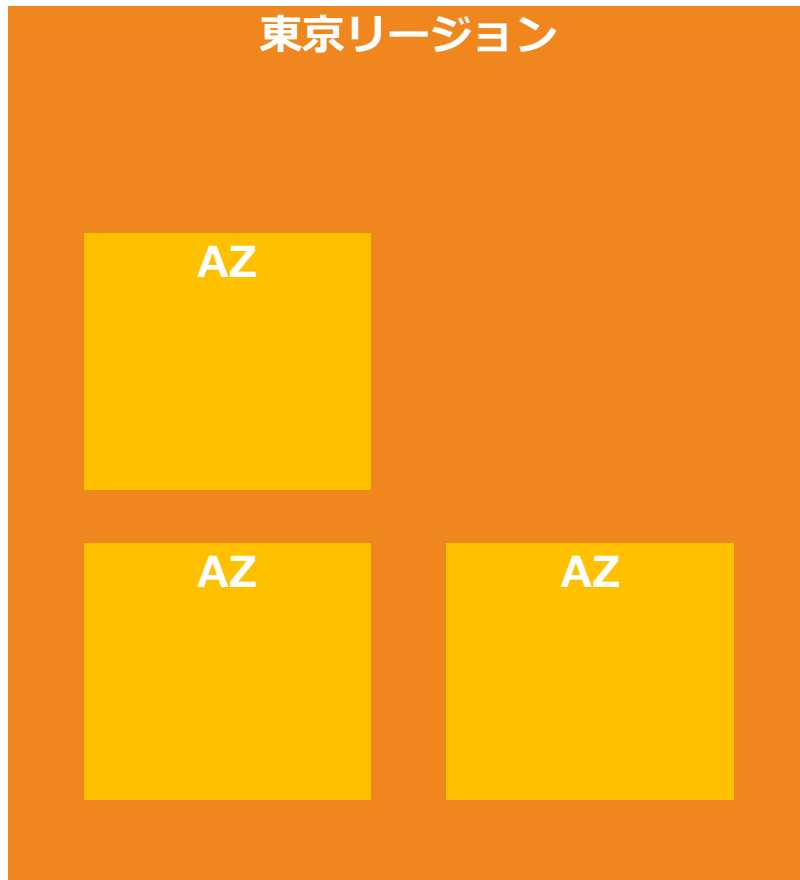
リージョン

リージョンに応じてAWSサービスの利用可否と値段が異なる



アベイラビリティゾーン (AZ)

リージョンの中に複数の独立したインフラ拠点が存在し、それをアベイラビリティゾーンと呼ぶ

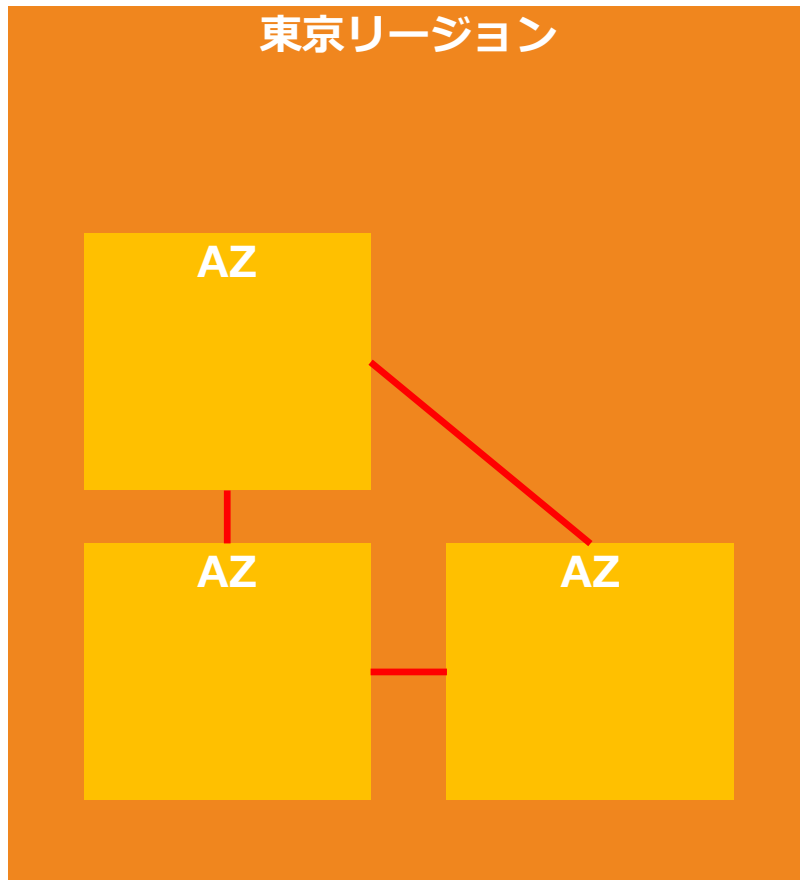


1つのリージョンには
2つ以上のAZが存在



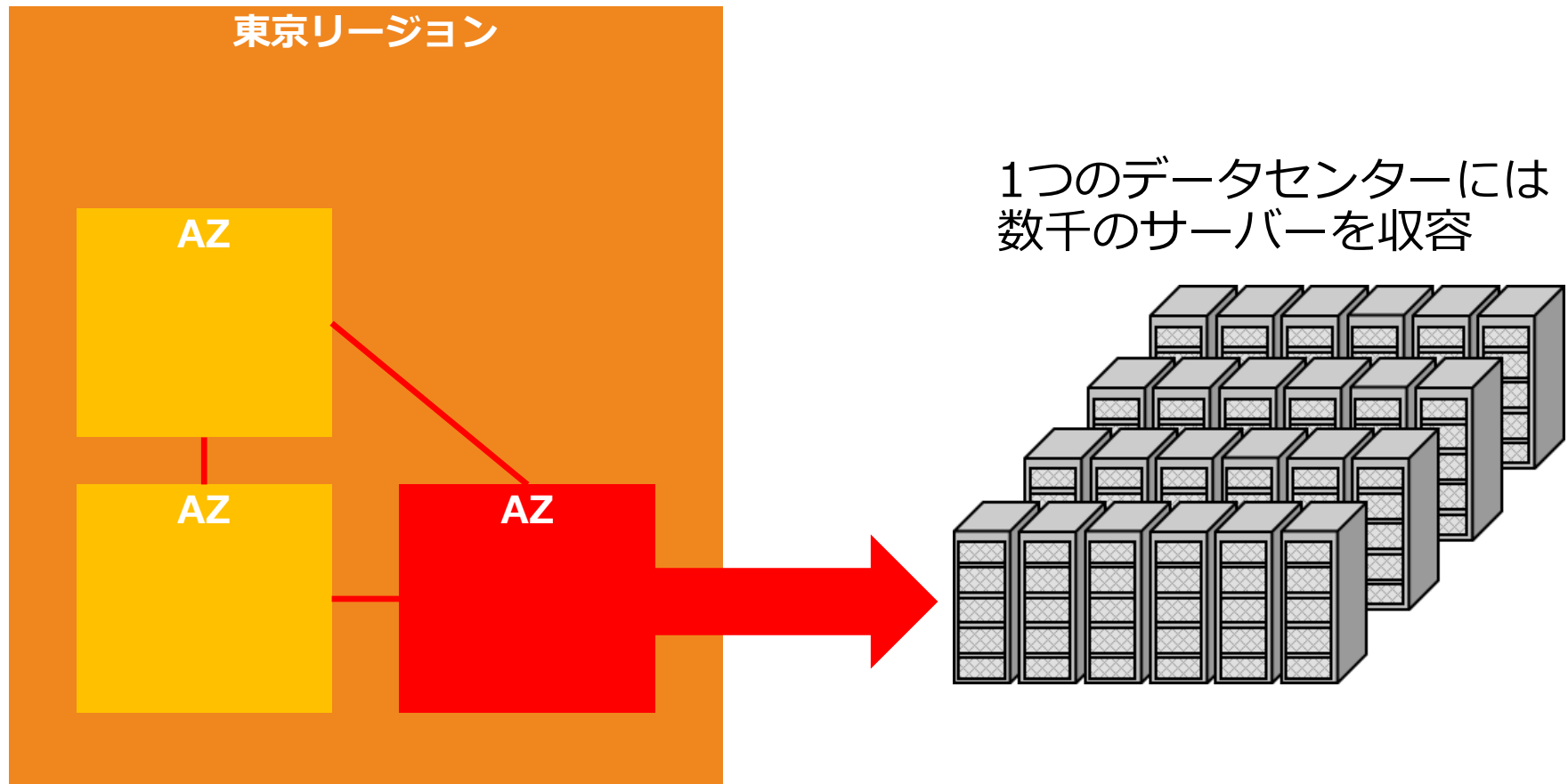
アベイラビリティゾーン (AZ)

同リージョン内のAZ同士は低レイテンシーのリンクで接続されている



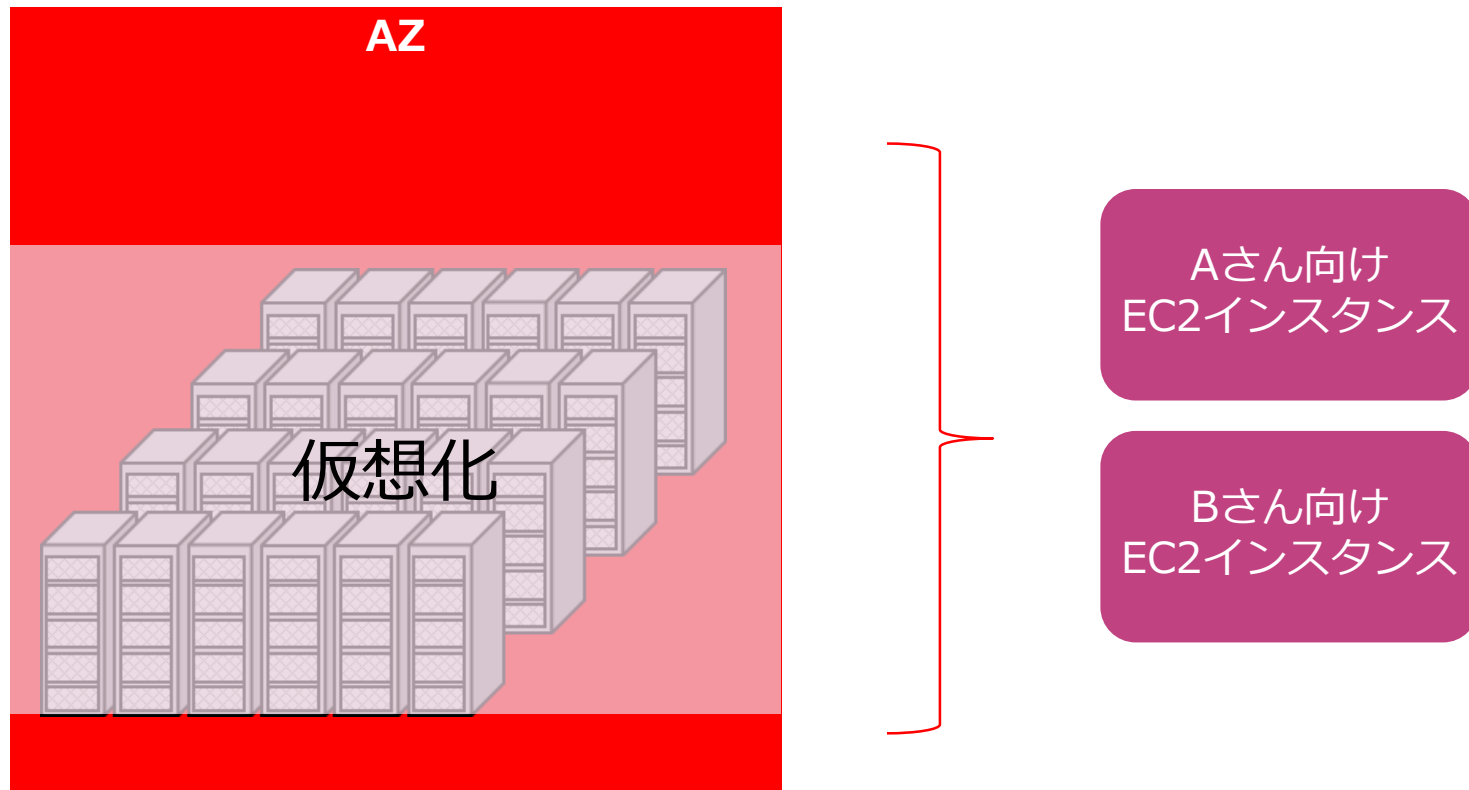
アベイラビリティゾーン (AZ)

AZは1つの複数の物理的なデータセンターで構成されている



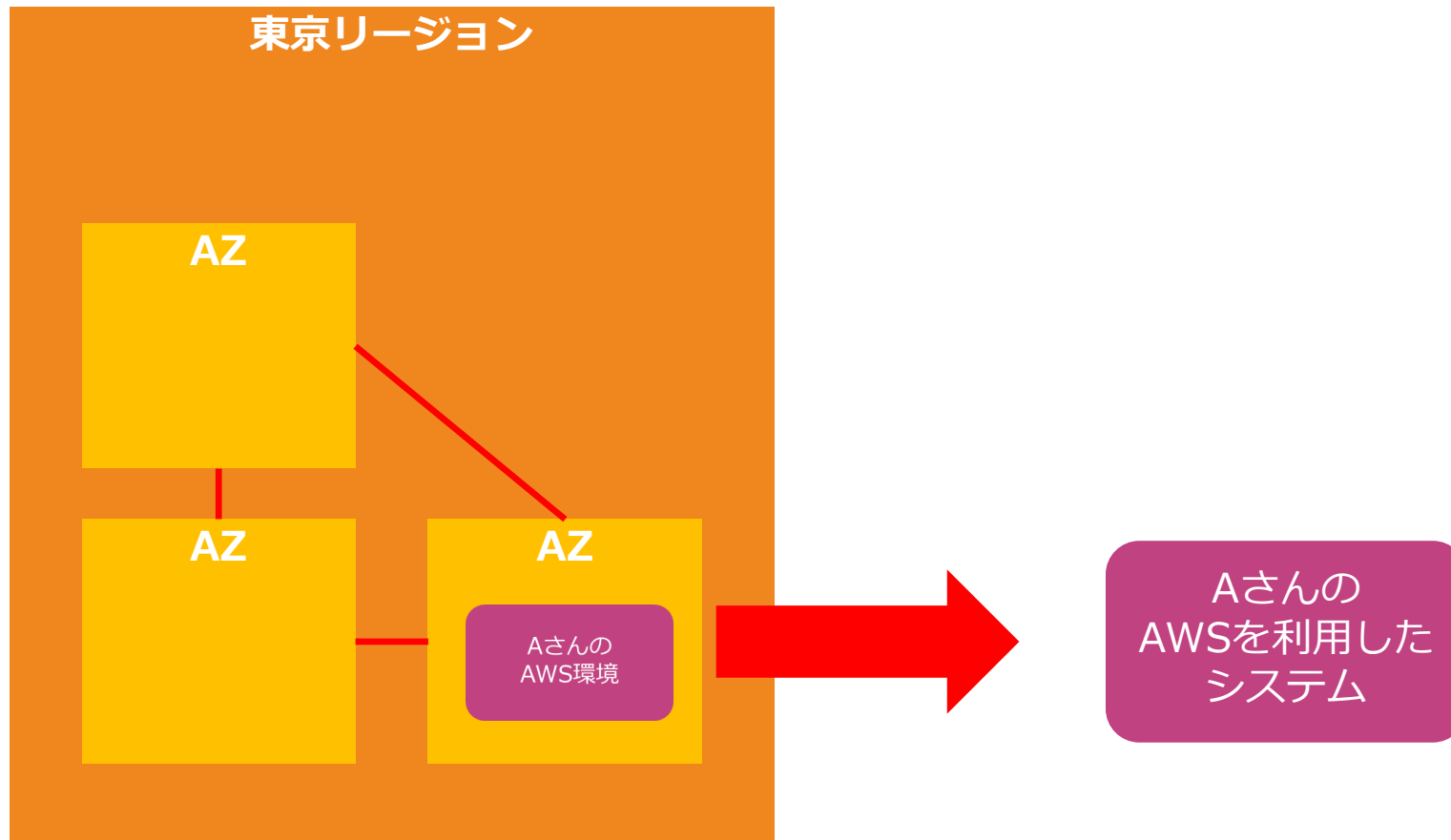
アベイラビリティゾーン (AZ)

AZにある物理インフラを仮想化してユーザーにインフラ機能としてサービスを提供している



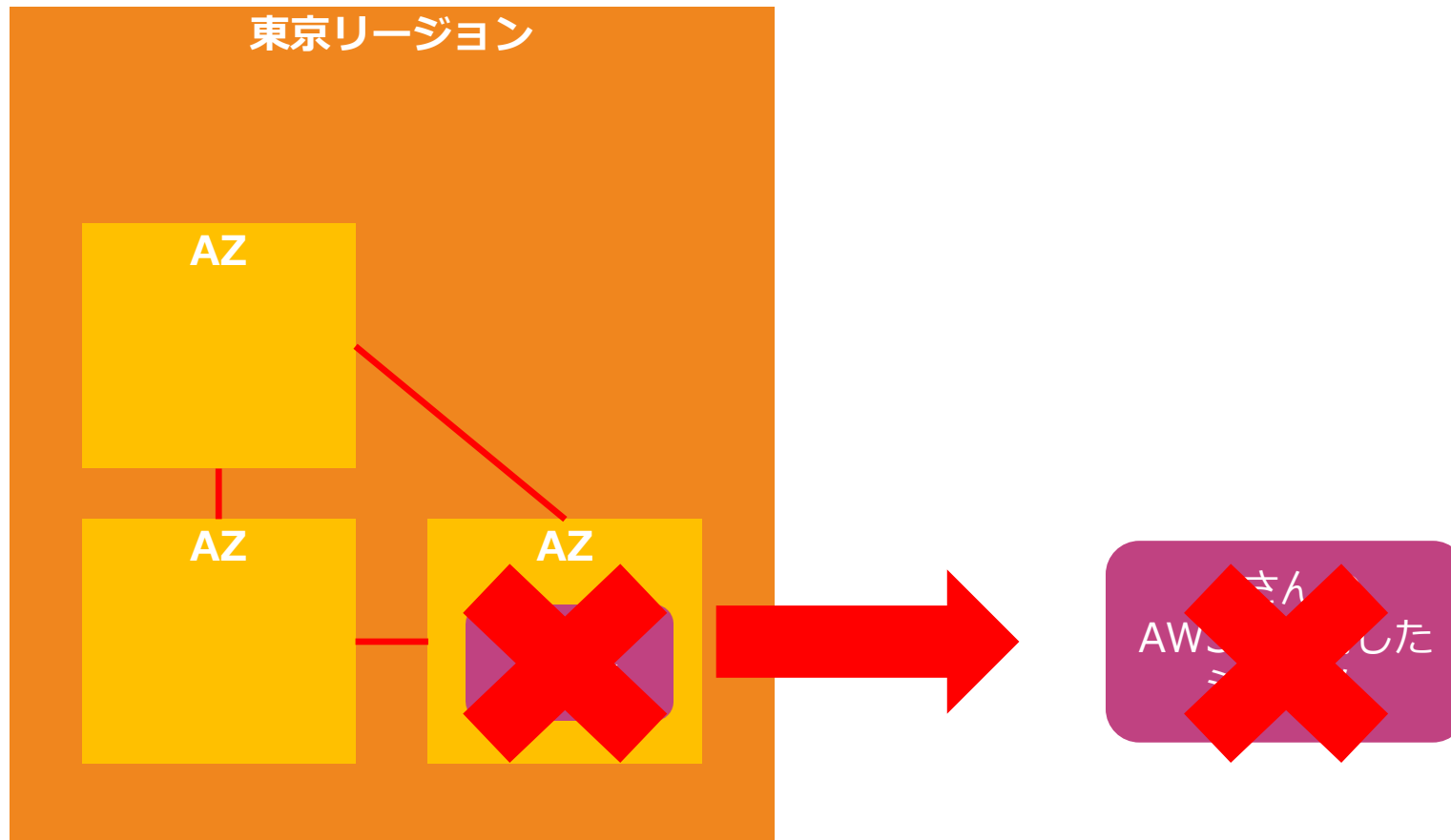
アベイラビリティゾーン (AZ)

よって、1つのAZ内のみでAWSサービスを利用しているとデータセンターの停止によるサービス停止の可能性がある



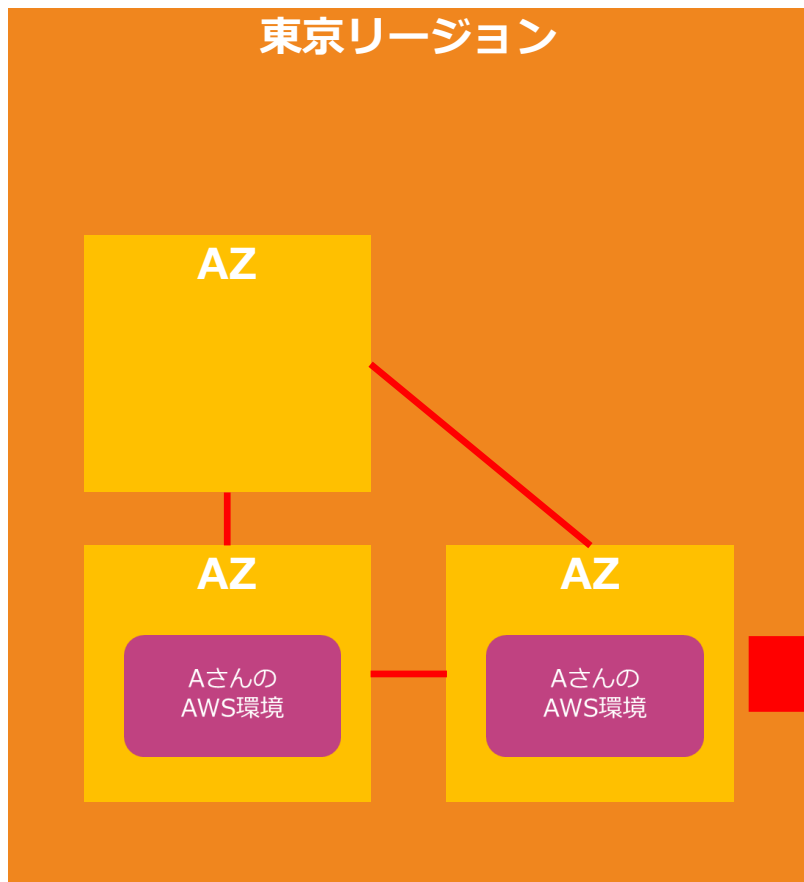
アベイラビリティゾーン (AZ)

よって、1つのAZ内のみでAWSサービスを利用しているとデータセンターの停止によるサービス停止の可能性がある



アベイラビリティゾーン (AZ)

複数AZで分けて信頼性の高いシステム構成にするのが基本的なAWSアーキテクチャとなる



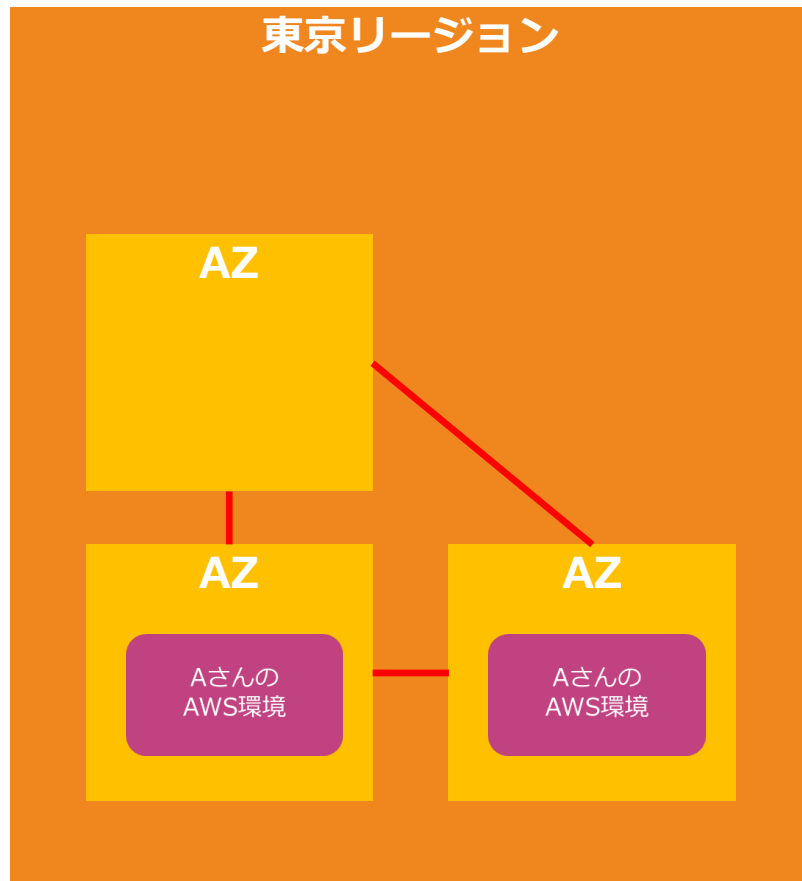
【推奨】
1つのリージョンに2つのAZから始める

Aさんの
AWSを利用した
システム



アベイラビリティゾーン (AZ)

複数AZを跨ぐと物理的な耐久性などが向上するが、システム間の連携や共有が制限される

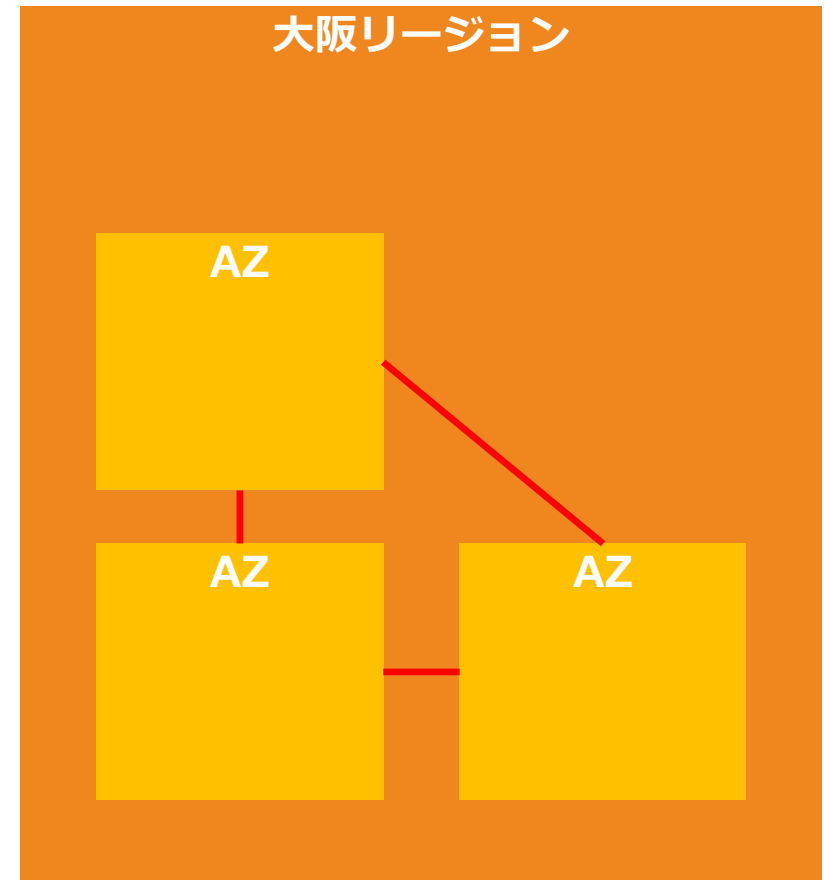
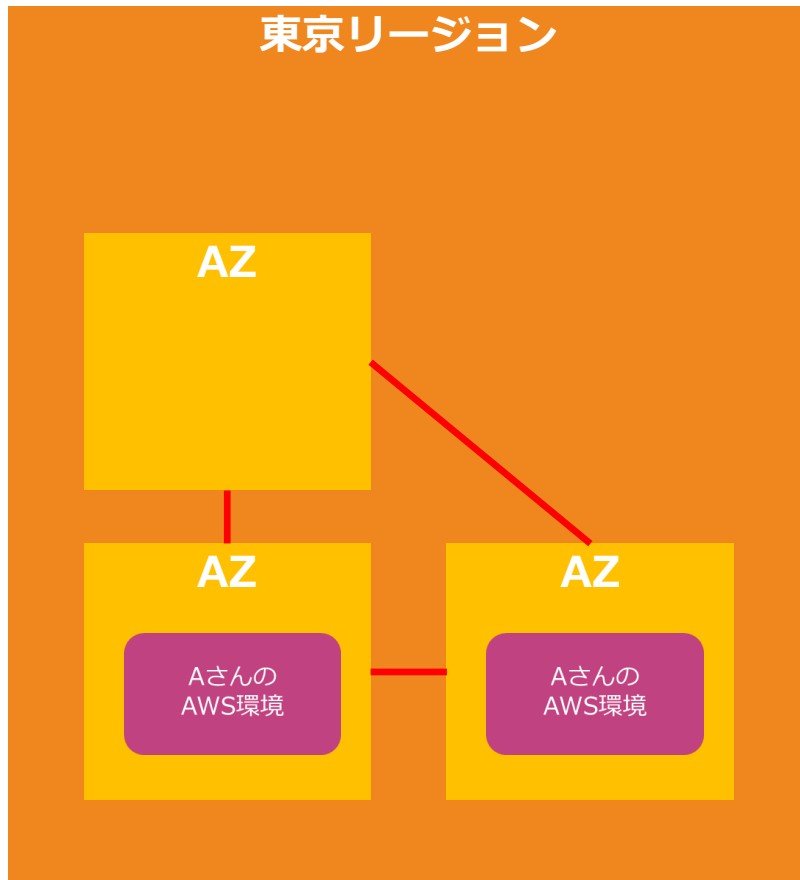


- ✓ 単一AZ内でしか共有されない設定などが多い
- ✓ 多くはAZ間で連携するための設定が必要



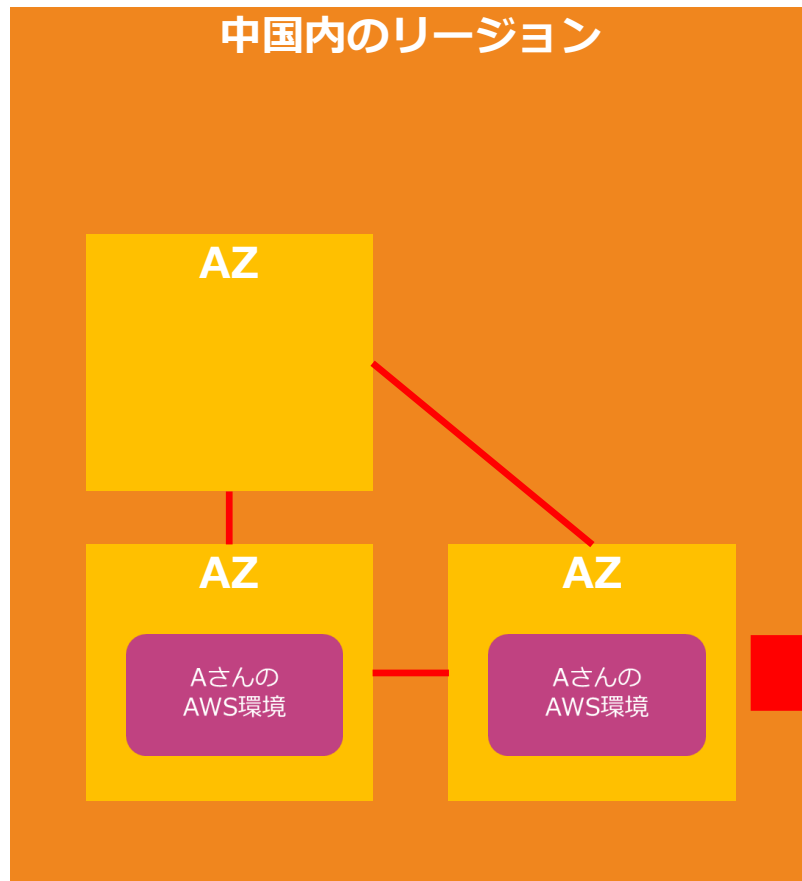
リージョンの選択

データやシステムに係る法律や社内規定を考慮し、基本的には自身の身近なリージョンを選択してAWSシステムを構築する



リージョンの選択

リージョンのある国の法律に影響される可能性も考慮する

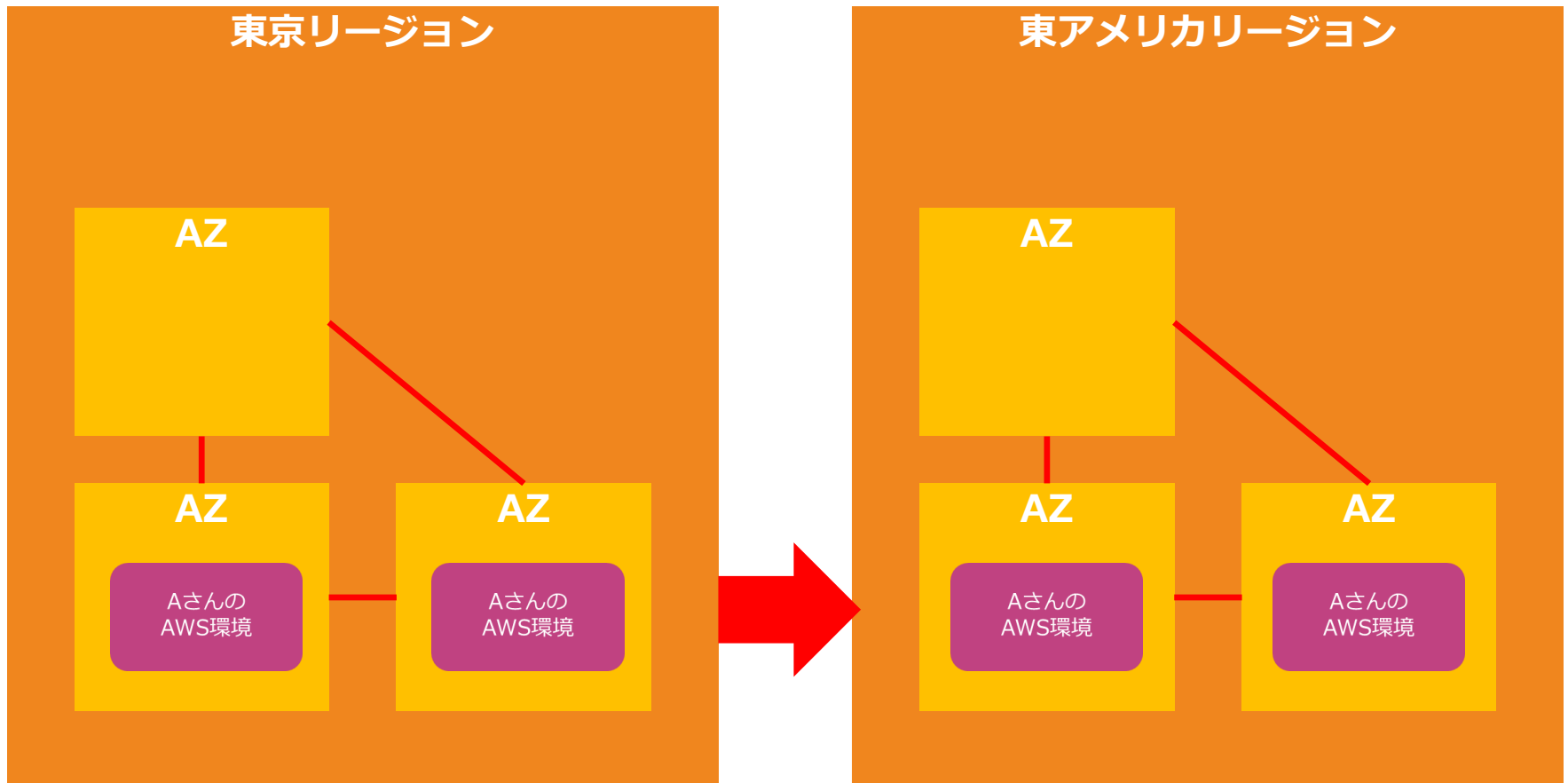


- ✓ 中国政府の要請に従いAWS中国はデータを提示する義務が生じる
- ✓ 中国内のデータの持ち出し制限がある



リージョンの選択

事業継続性計画（BCP）などの対策のためデータや予備システムとして別リージョンを利用する



リージョン間の連携

リージョン間では専用線接続とレプリカ作成ができるようになっている

【リージョン間で可能な連携】

- ✓ AWS Direct Connect Gateway経由で接続
- ✓ インターリージョンVPC Peeringで接続
- ✓ EC2リージョン間でのAMIコピー
- ✓ S3リージョン間のレプリケーション
- ✓ RDSリージョン間でのリードレプリカ
- ✓ DynamoDBリージョン間でのレプリカライブラリ
- ✓ Route53DNSフェイルオーバー



エッジロケーション

エッジロケーションはキャッシュデータなどを利用する際の更に小さなエンドポイントとなる拠点

- ✓ CloudFront
- ✓ AWS CDN



AWSの操作

AWSの操作方法は3つ：

AWS
マネジメント
コンソール

インスタンス操作
(SSH等)

AWS
CLI操作



AWSマネジメントコンソール

AWSマネジメントコンソールはWEB画面からAWSを設定できるGUIツール

The screenshot displays the AWS Management Console interface for selecting an Amazon Machine Image (AMI). The top navigation bar shows the AWS logo and various service categories. The left-hand navigation menu lists various AWS services, including EC2, IAM, S3, and others. The main content area is titled 'ステップ 1: Amazon マシンイメージ (AMI)' and includes a search bar and a list of available AMIs. The AMIs are categorized by 'Quickstart' and include details such as the AMI ID, architecture, and root device type. The AMIs listed are:

- Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-02e680c4540db351e (64 ビット x86) / ami-0998858ab6ad47da8 (64 ビット Arm)
- Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0cd3dfa4e37921605
- SUSE Linux Enterprise Server 15 (HVM), SSD Volume Type - ami-0eb9f58db22854f8f
- Red Hat Enterprise Linux 7.6 (HVM), SSD Volume Type - ami-0b500ef59d8335eee (64 ビット x86) / ami-0302c1ecc74930ba5 (64 ビット Arm)
- Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0f65671a86f061cd1 (64 ビット x86) / ami-0f2057f28f0a44d06 (64 ビット Arm)

At the bottom of the console, there is a banner for Amazon RDS, encouraging users to try it out for their database needs.

インスタンス接続

設置したインスタンス（サーバー）の操作はWindowsサーバーやLinuxサーバーなどの標準に準じたソフトウェアを利用

Windowsサーバー向け操作GUI



Linuxサーバー操作SSHソフト

```
Connection to 52.17.142.65 closed by remote host.
Connection to 52.17.142.65 closed.
Ryans-iMac:Downloads ryankroonenburg$ clear

Ryans-iMac:Downloads ryankroonenburg$ ssh ec2-user@52.48.51.207 -i MyEC2Key.pem
The authenticity of host '52.48.51.207 (52.48.51.207)' can't be established.
ECDSA key fingerprint is SHA256:IDXGMYu3Wr1edahgRytDpND5XptqBM9aPfyLFhuLZW.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '52.48.51.207' (ECDSA) to the list of known hosts.

 _ _ | _ _ | _ _ |
 _ | ( _ / _ _ |
 _ _ | _ _ | _ _ |

Amazon Linux AMI

https://aws.amazon.com/amazon-linux-ami/2015.09-release-notes/
20 package(s) needed for security, out of 39 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-36-83 ~]$
```



SSHソフトウェア

設置したサーバーなどにアクセスして内部設定やソフトウェア操作を実行する通信方式及びそのソフトウェア

```
Connection to 52.17.142.65 closed by remote host.
Connection to 52.17.142.65 closed.
Ryans-iMac:Downloads ryankroonenburg$ clear

Ryans-iMac:Downloads ryankroonenburg$ ssh ec2-user@52.48.51.207 -i MyEC2Key.pem
The authenticity of host '52.48.51.207 (52.48.51.207)' can't be established.
ECDSA key fingerprint is SHA256:iDXGMyu3Wr1edahgRytDpNDSXptqbM9aPfyLFhuL2Wo.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '52.48.51.207' (ECDSA) to the list of known hosts.

  __|  __|_  )
 _| (    /   Amazon Linux AMI
---| \___|___|

https://aws.amazon.com/amazon-linux-ami/2015.09-release-notes/
20 package(s) needed for security, out of 39 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-36-83 ~]$
```

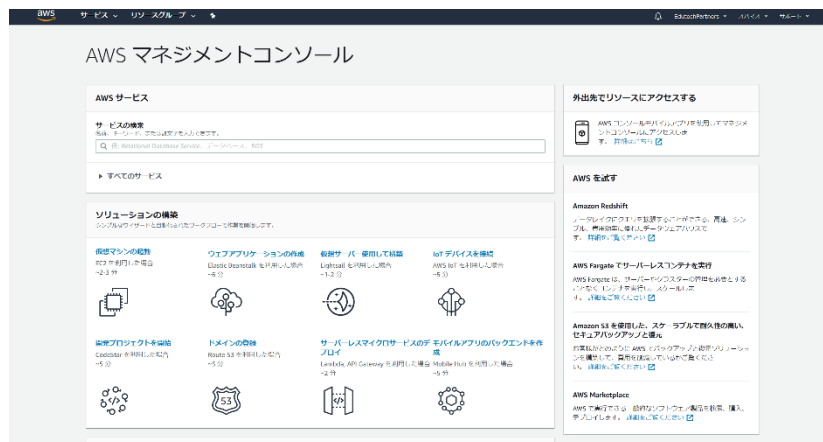
Udemy



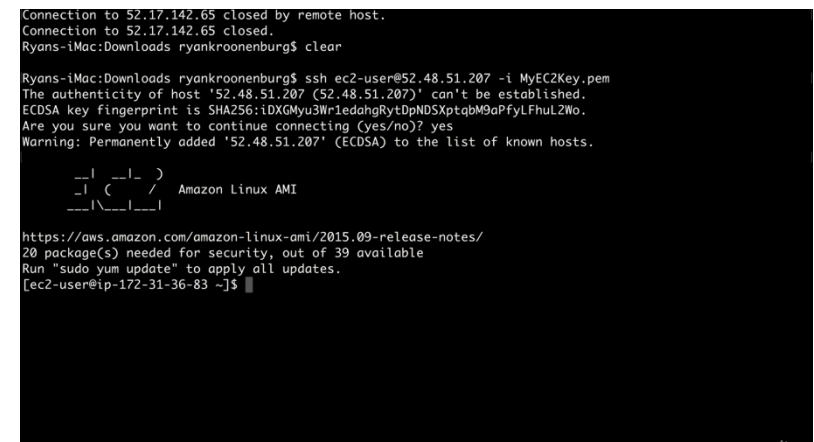
基本操作

AWSの設定・操作はまずはマネジメントコンソールでインスタンスなどを設置し、SSHで接続して操作するのが基本

【AWSマネジメントコンソール】
AWS上にインスタンスなどを設置

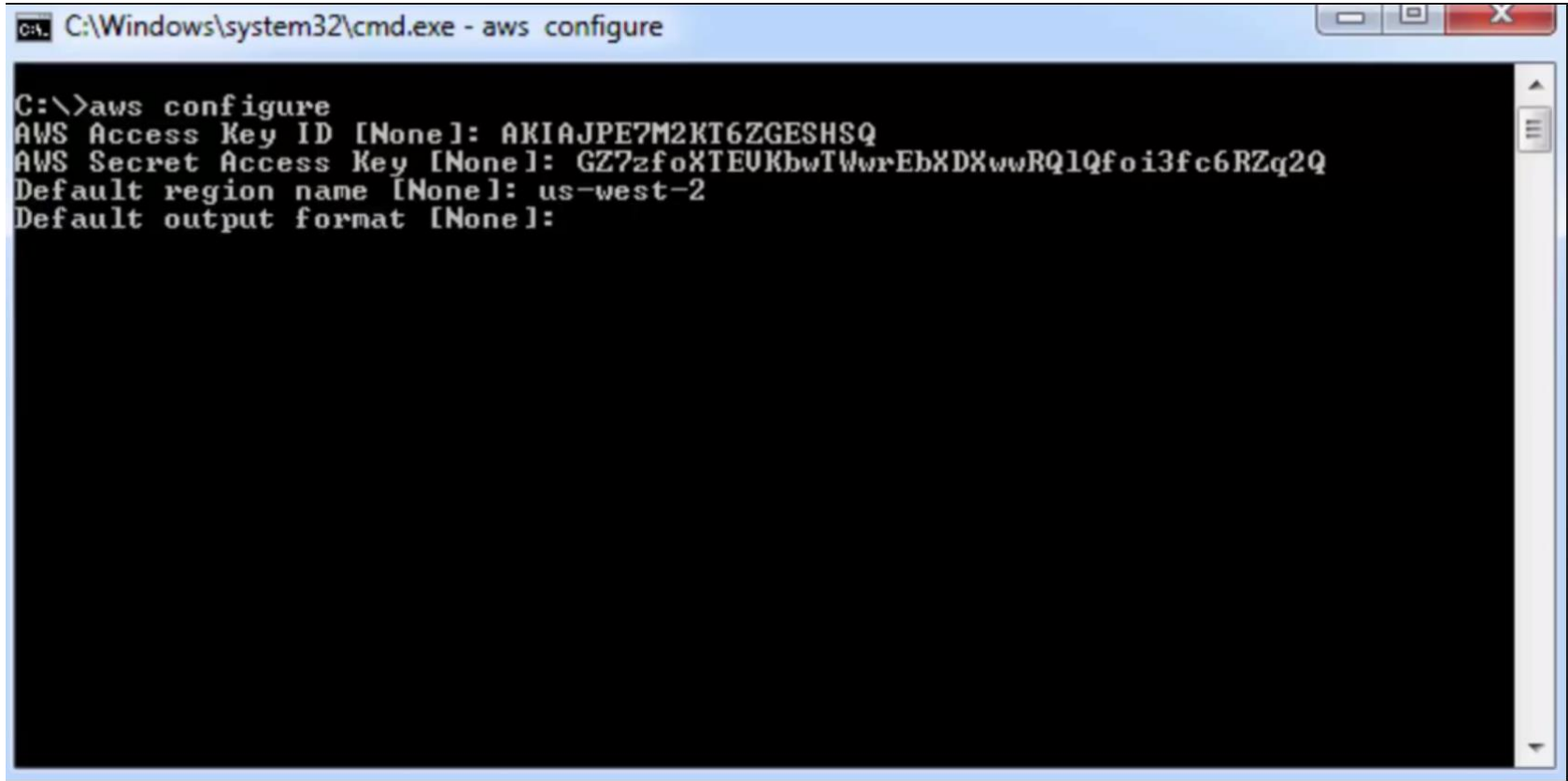


【インスタンス操作】
設置したインスタンス
に接続して操作する



AWS CLI

AWS CLIはコマンドラインから複数のAWSサービスを制御・管理することが可能



```
C:\Windows\system32\cmd.exe - aws configure

C:\>aws configure
AWS Access Key ID [None]: AKIAJPE7M2KT6ZGESHSQ
AWS Secret Access Key [None]: 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
Default region name [None]: us-west-2
Default output format [None]:
```

