



# AWS入門～第11回～

Amazon Auroraのご紹介



# 自己紹介

▶ 重本 尚志

▶ 略歴：

- ▶ 徳島大学工学部卒業（2008-03）
- ▶ 独立系IT企業に新卒として入社（2008-04）
- ▶ C#やJavaを中心（クラサバ多め）に案件を転々とする。
- ▶ 退職・独立（2017-01）
- ▶ 株式会社PUreatio設立（2018-12）

▶ 趣味：トレーディングカード収集

▶ 好きな食べ物：奈良漬、味噌ラーメン



# 目次

- ▶ Amazon Auroraとは
- ▶ Amazon Auroraの機能紹介



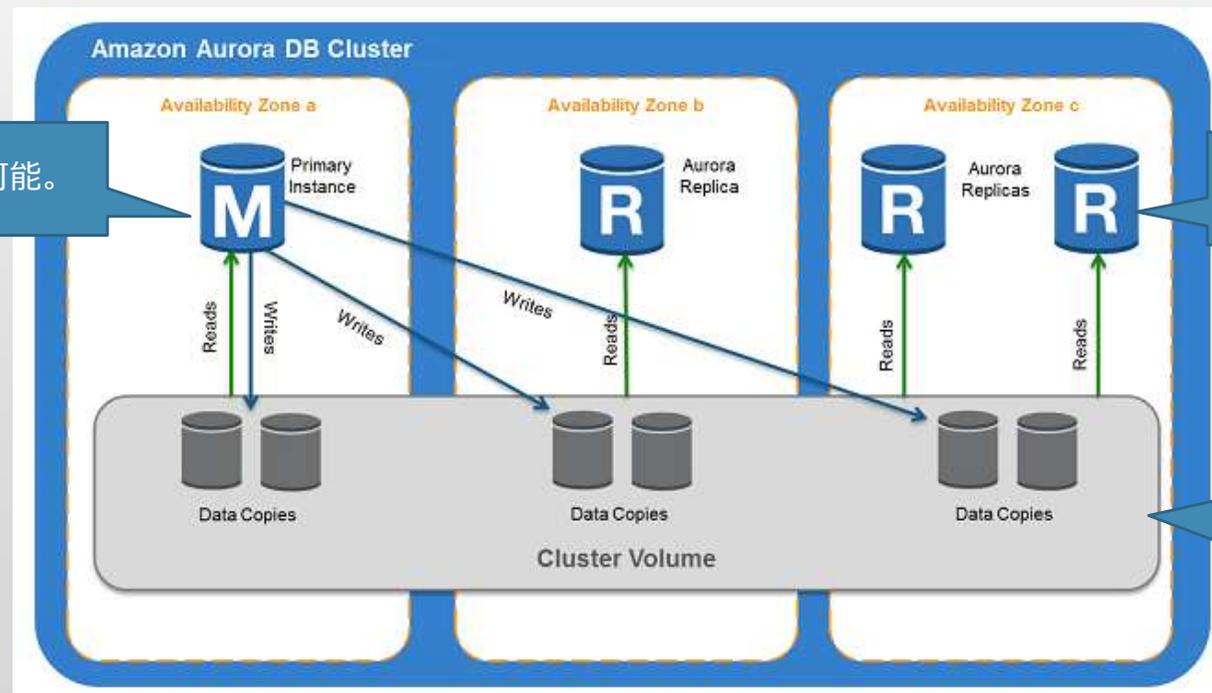
# Amazon Auroraとは

- ▶ AmazonWebServiceが提供するRDSのDBエンジンの1つ
  - ▶ 2014/10からサービスを提供している
  - ▶ 無償版（無償期間）は存在しない
- ▶ MySQL、PostgreSQLと互換性のあるリレーショナルデータベース
  - ▶ インスタンスもMySQL互換エディション、PostgreSQL互換エディションのそれぞれが存在している
  - ▶ すべてのバージョンやエンジンで互換性を保持しているわけではない

# Amazon Auroraとは

## Amazon Auroraの構成

- インスタンスと、そのデータを管理するクラスターボリュームに別れている。



Primaryはデータ変更が可能。

通常レプリカは読み取りのみ。

クラスターボリュームは3AZにまたがっている。  
10G単位の10G~128T割り当て済みストレージ。

# Amazon Auroraとは

## ▶ MySQL、PostgreSQLとの互換性

### ▶ MySQL（現在最新は8.0.28）

#### ▶ 5.6系、5.7系、8.0.23

▶ MySQL側の機能でも、Amazon Auroraではサポートされていない機能もある。

▶ [https://docs.aws.amazon.com/ja\\_jp/AmazonRDS/latest/AuroraUserGuide/Aurora.MySQL.CompareMySQL57.html](https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/AuroraUserGuide/Aurora.MySQL.CompareMySQL57.html)

#### ▶ InnoDBストレージエンジン

▶ MyISAMストレージエンジンは対応していない

### ▶ PostgreSQL（現在最新は14.1）

▶ 10.11~10.18、11.6~11.13、12.4~12.8、13.3~13.4

# Amazon Auroraとは

## ▶ 料金比較

- ▶ Amazon Auroraのほうが、通常のMySQL、PostgreSQLより料金が少し高い。
- ▶ 料金体系も少し異なっている。

Amazon Aurora MySQL Edition

リージョン:	米国東部 (オハイオ) ↕
スタンダードインスタンス - 現行世代	
	時間あたりの料金
db.t4g.medium	0.073USD
db.t4g.large	0.146USD

MySQL

db.t4g.medium	0.065USD
db.t4g.large	0.129USD



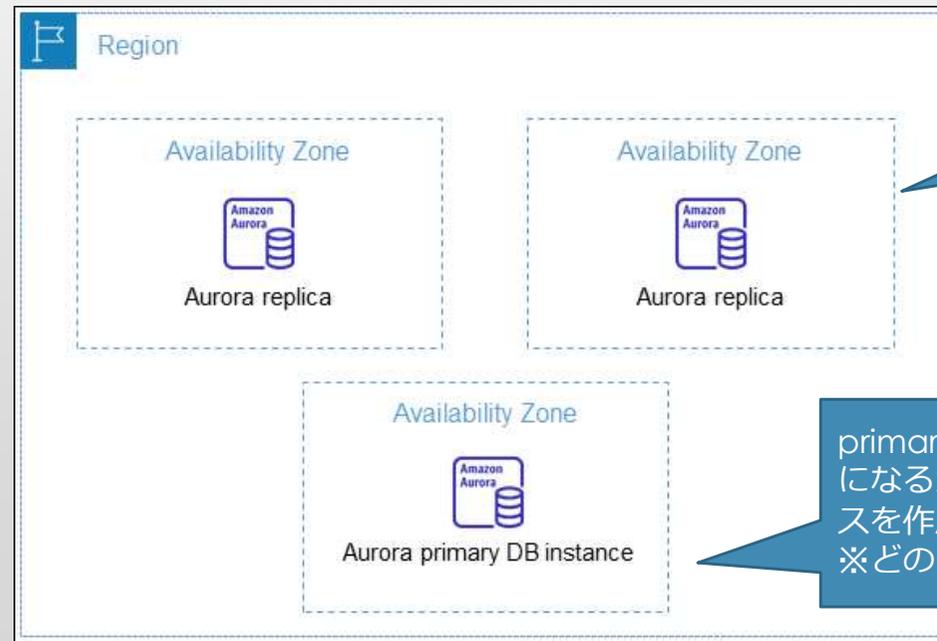
# Amazon Auroraとは

- ▶ Amazon Auroraの優れているところ
  - ▶ 処理速度が速い
    - ▶ MySQLの5倍のスループット
    - ▶ PostgreSQLの3倍のスループット
  - ▶ 高可用性、耐久性、安全性を備えている
    - ▶ ストレージのAutoScaling
    - ▶ AWS KMSを利用して、インスタンスの暗号化ができる
    - ▶ Performance Insightsが利用できる
    - ▶ CPUのモニタリング拡張ができる
    - ▶ Amazon CloudWatchにログがエクスポートできる
    - ▶ マイナーバージョンのアップグレードが自動で行える
    - ▶ etc...

# Amazon Auroraとは

- ▶ Amazon Auroraの高可用性・耐久性

- ▶ 3つのAZにレプリカ（最大15、マルチAZ）を作成、ユーザーデータを6個同期し、継続的にS3にバックアップしている。



primary以外は基本的に読み取り専用。

primaryに問題が発生した場合、別のレプリカがprimaryになるか、（レプリカがない場合は）新しいインスタンスを作成してそれをprimaryにする。  
※どのレプリカをprimaryにするかの優先付けは可能。



# Amazon Auroraとは

## ▶ 導入事例

- ▶ <https://www.slideshare.net/AmazonWebServicesJapan/aurora-amazon-aurora>
- ▶ <https://www.s-style.co.jp/jirei/case043.html>
- ▶ <https://www.cloud-koubou.jp/case/%E6%A0%AA%E5%BC%8F%E4%BC%9A%E7%A4%BE%E7%BF%BB%E8%A8%B3%E3%82%BB%E3%83%B3%E3%82%BF%E3%83%BC-%E5%B0%8E%E5%85%A5%E4%BA%8B%E4%BE%8B%E3%81%AE%E3%81%94%E7%B4%B9%E4%BB%8B/>



# Amazon Auroraの機能紹介

- ▶ バックトラック
- ▶ グローバルデータベース
- ▶ 並列クエリ

# Amazon Auroraの機能紹介

## ▶ バックトラック

- ▶ Auroraのデータをバックアップデータを使わずに特定の状態まで戻せる機能。
  - ▶ MySQLのみ利用可能、一部のバージョンでは利用できない。
  - ▶ 特定のテーブルのみの操作はできない。
- ▶ バックトラックに指定した時間が、正確に一致しない場合がある。
  - ▶ Auroraが指定時間に最も則した時間を判断するため。
- ▶ 最大指定可能時間は72時間。

### バックトラック

Backtrack では、DB クラスターをすばやく特定の時点に巻き戻すことができ、別の DB クラスターを作成する必要はありません。 [情報](#)

#### バックトラックを有効にする

Backtrack を有効にすると、バックトラックのための変更の保存について課金されます。

#### ターゲットバックトラックウィンドウ

バックトラックウィンドウはさかのぼることができる時間を決定します。Aurora はその時間枠をサポートするために十分なログ情報の保持を試みます。 [情報](#)

時間 (最大  
72)

# Amazon Auroraの機能紹介

## ■ グローバルデータベース

- 複数リージョンに跨って、1つのデータベースを利用できる機能。

- 一部のバージョンでは利用できない。

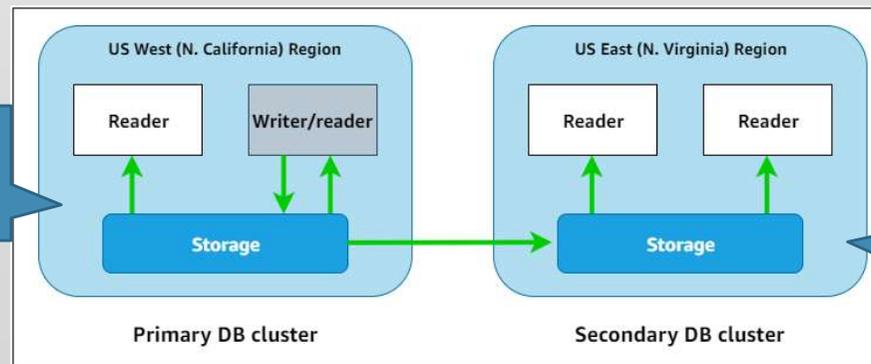
- MySQL、PostgreSQLの両方で利用可能。

- バックトラックには対応していない。

- 構成要件が存在する。

- [https://docs.aws.amazon.com/ja\\_jp/AmazonRDS/latest/AuroraUserGuide/aurora-global-database-getting-started.html#aurora-global-database.configuration.requirements](https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/AuroraUserGuide/aurora-global-database-getting-started.html#aurora-global-database.configuration.requirements)

Primaryは読み取り/書き込み可能。

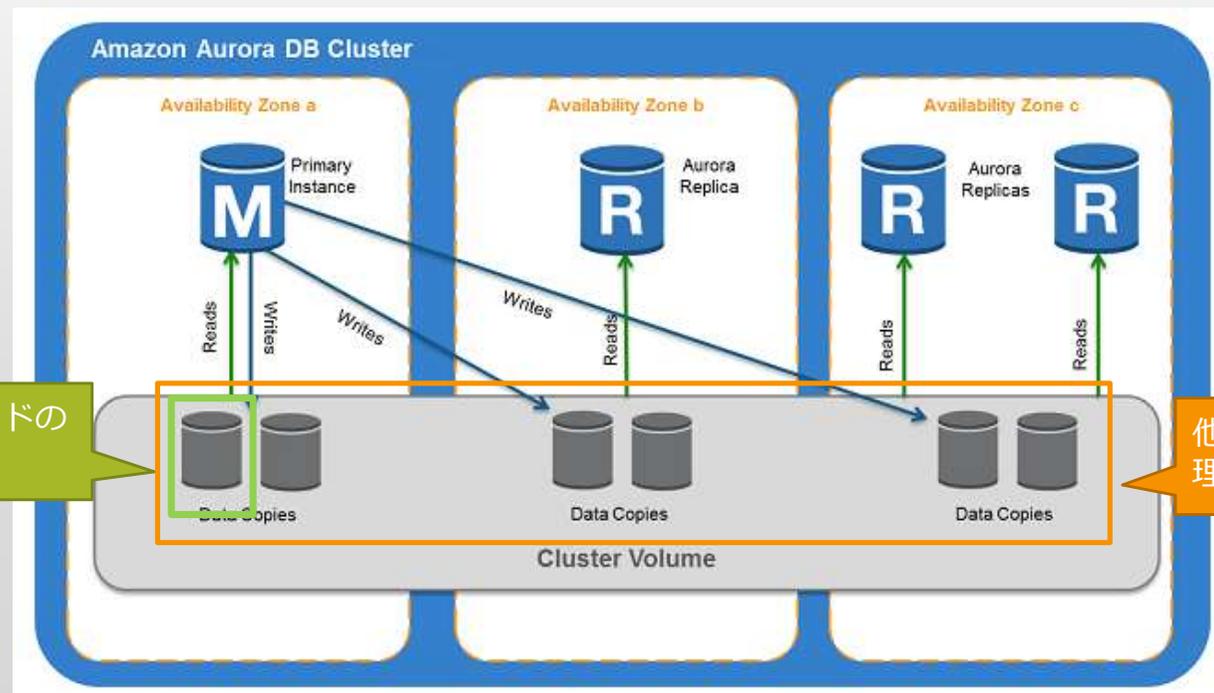


読み取り専用Secondaryは、最大5つ。SecondaryのAuto Scaleはサポートされていない。

# Amazon Auroraの機能紹介

## ■ 並列クエリ

- 複数のストレージノードに分析処理を分散することで、性能向上を図る機能。



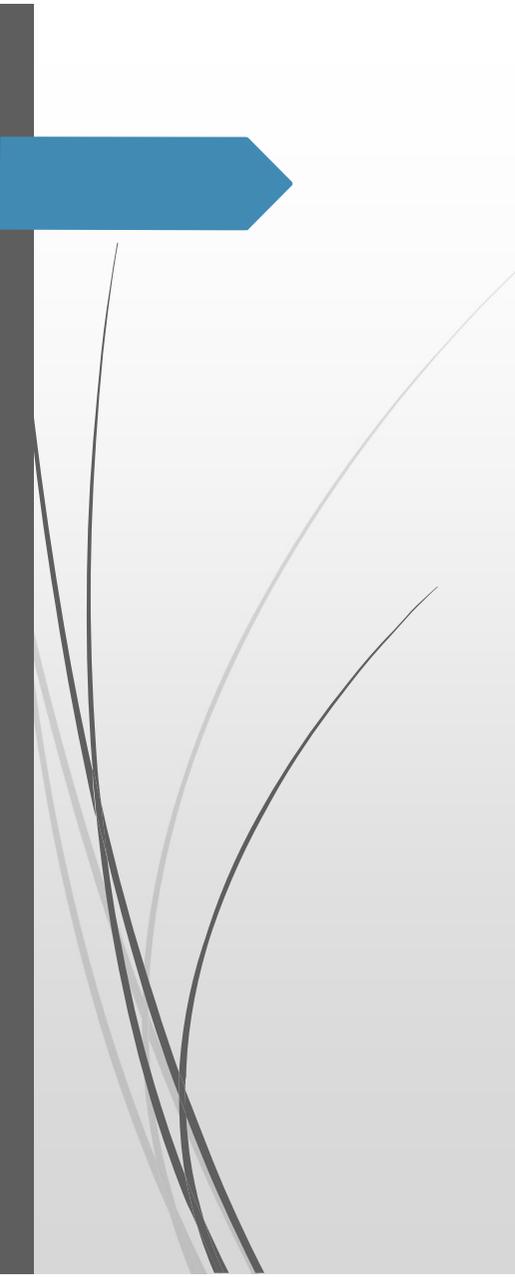
普段はヘッドノードのみで処理する。

他のノードにも分析処理を移譲する。



## 最後に

- ベストプラクティスがAWSのドキュメントに記載されている
  - [https://docs.aws.amazon.com/ja\\_jp/AmazonRDS/latest/AuroraUserGuide/Aurora.BestPractices.html](https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/AuroraUserGuide/Aurora.BestPractices.html)



**ご清聴ありがとうございました**