Deep Dive on Amazon Aurora

Yoav Eilat
Senior Product Manager, Amazon Aurora and Amazon RDS
AWS

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon Aurora…

Enterprise database at open source price

Delivered as a managed service

コマーシャルデータベースの性能 と 可用性

オープンソースデータベースのシンプルさとコスト効果

MySQL, PostgreSQLと互換性がある

利用した分だけお支払い

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora performance & scalability

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

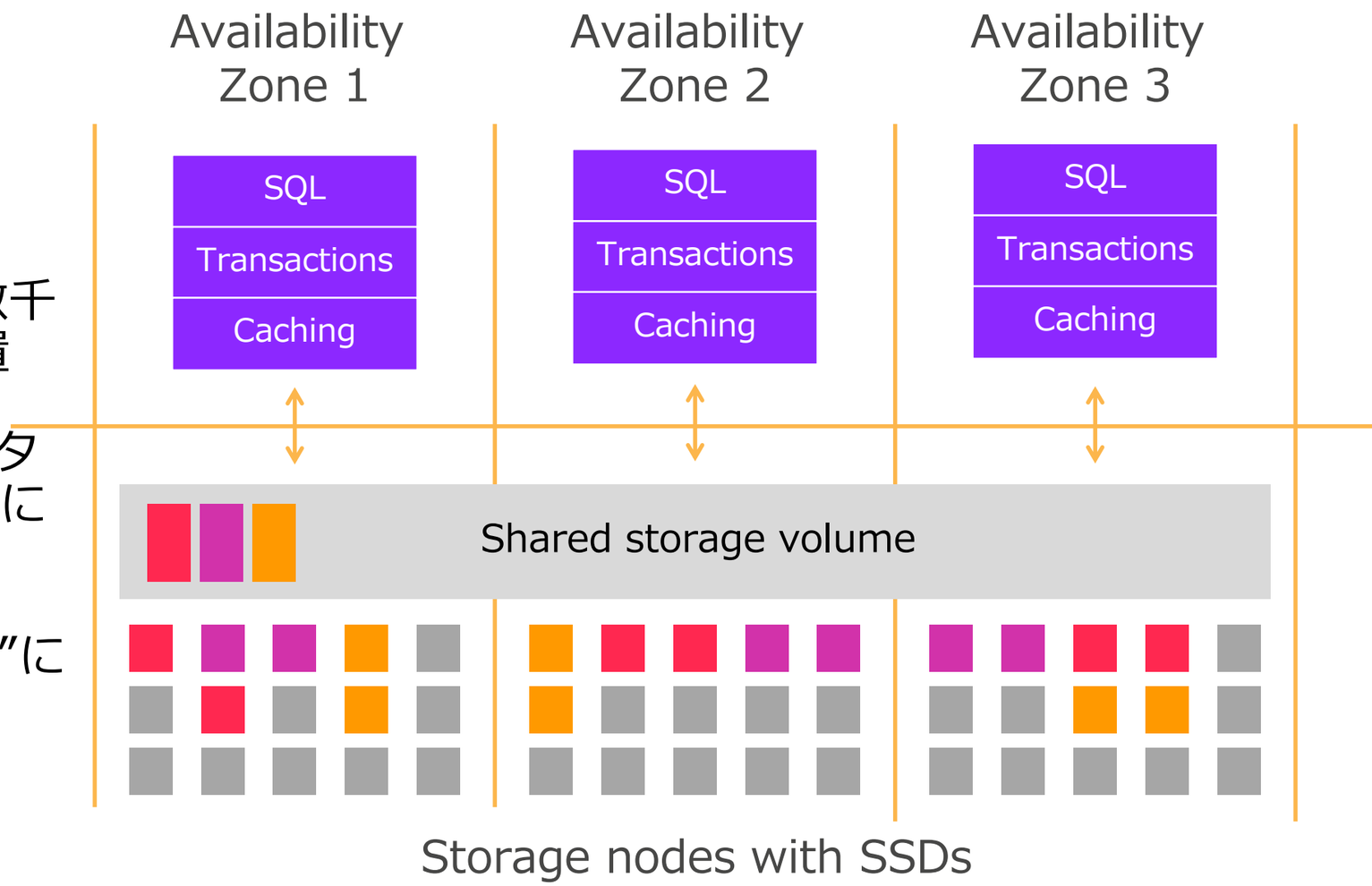
Aurora scale-out, distributed architecture

Amazon Aurora向けに作られた log-structured 分散ストレージシステム

ストレージボリュームは3AZに数百～数千インスタンスのストレージノードを配置

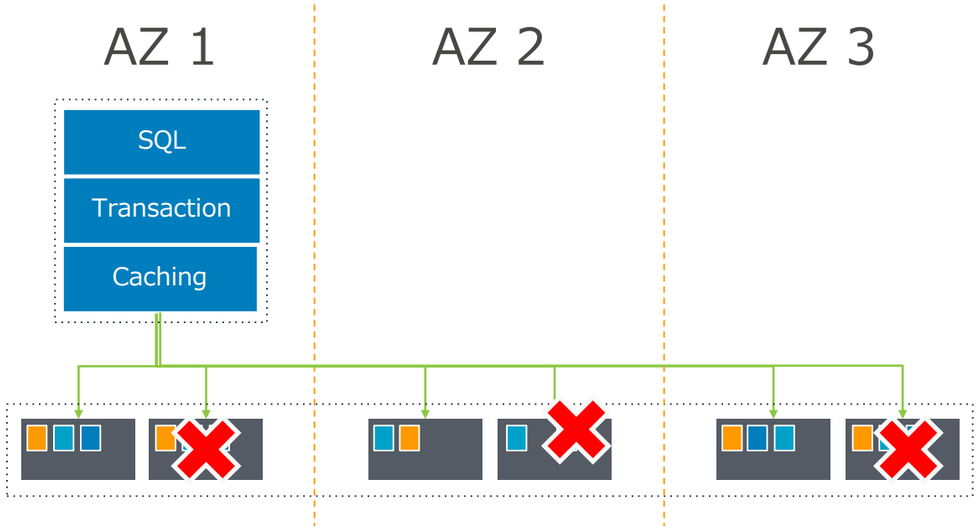
それぞれのAZに2つずつ、計6つのデータのコピーを保存することでAZ+1の障害に対応

データは10GBずつ“protection groups”に保存され、64TBまで自動的にスケールアップ

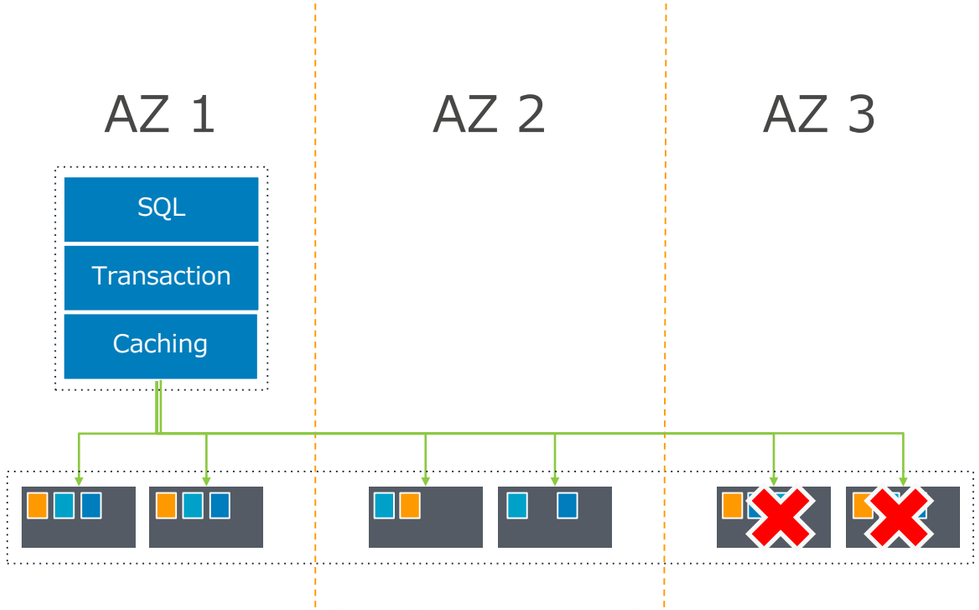


© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Six-way replicated storage



書き込み・読み込み可用性



読み込み可用性

- データは6ノードに非同期・並列で書き込む
- 書き込みクォーラムは4/6, 読み込みクォーラムは3/6ノード
- ノードの修復にはPeer to peer “gossip protocol”を利用

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Write quorum in action

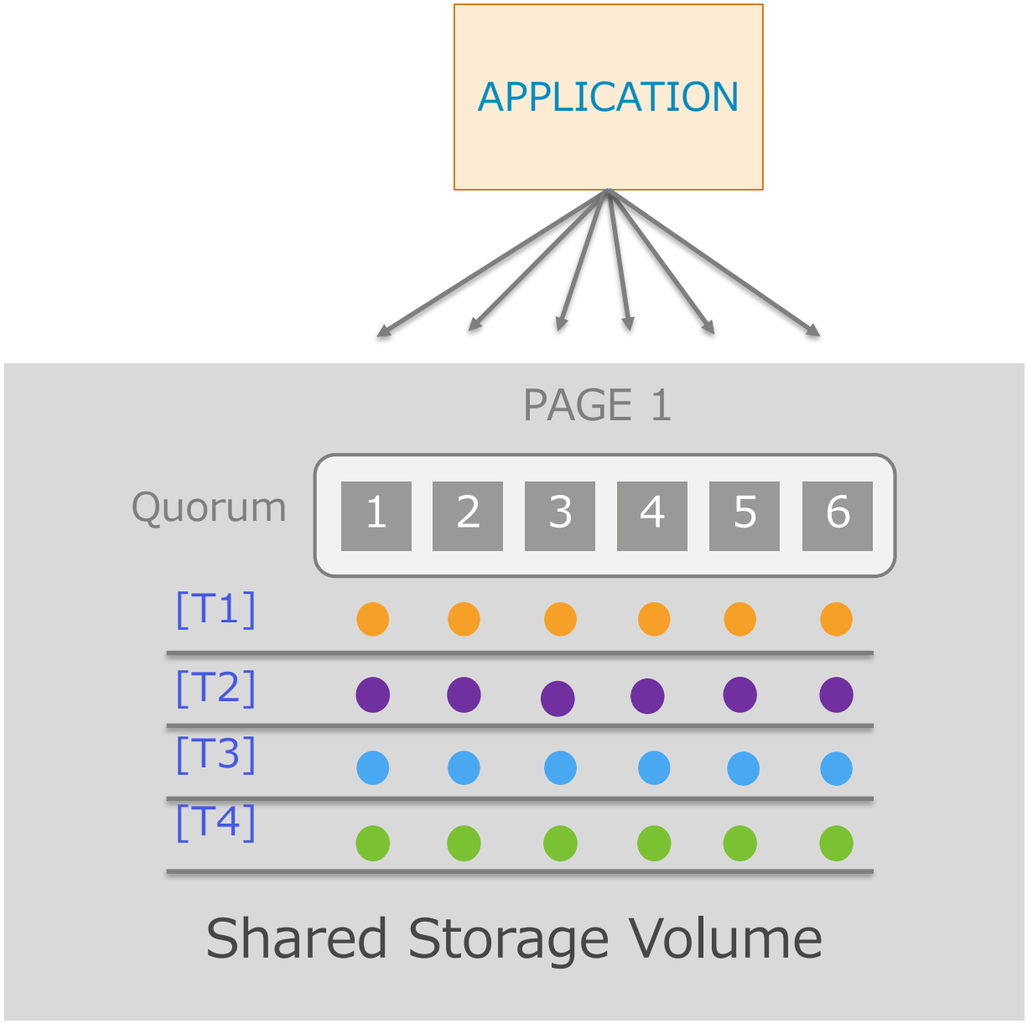
この例では4つのトランザクションからPage1書き込む

堅牢性とコミットを時系列で表現すると:

Transactions T1 T2 T3 T4

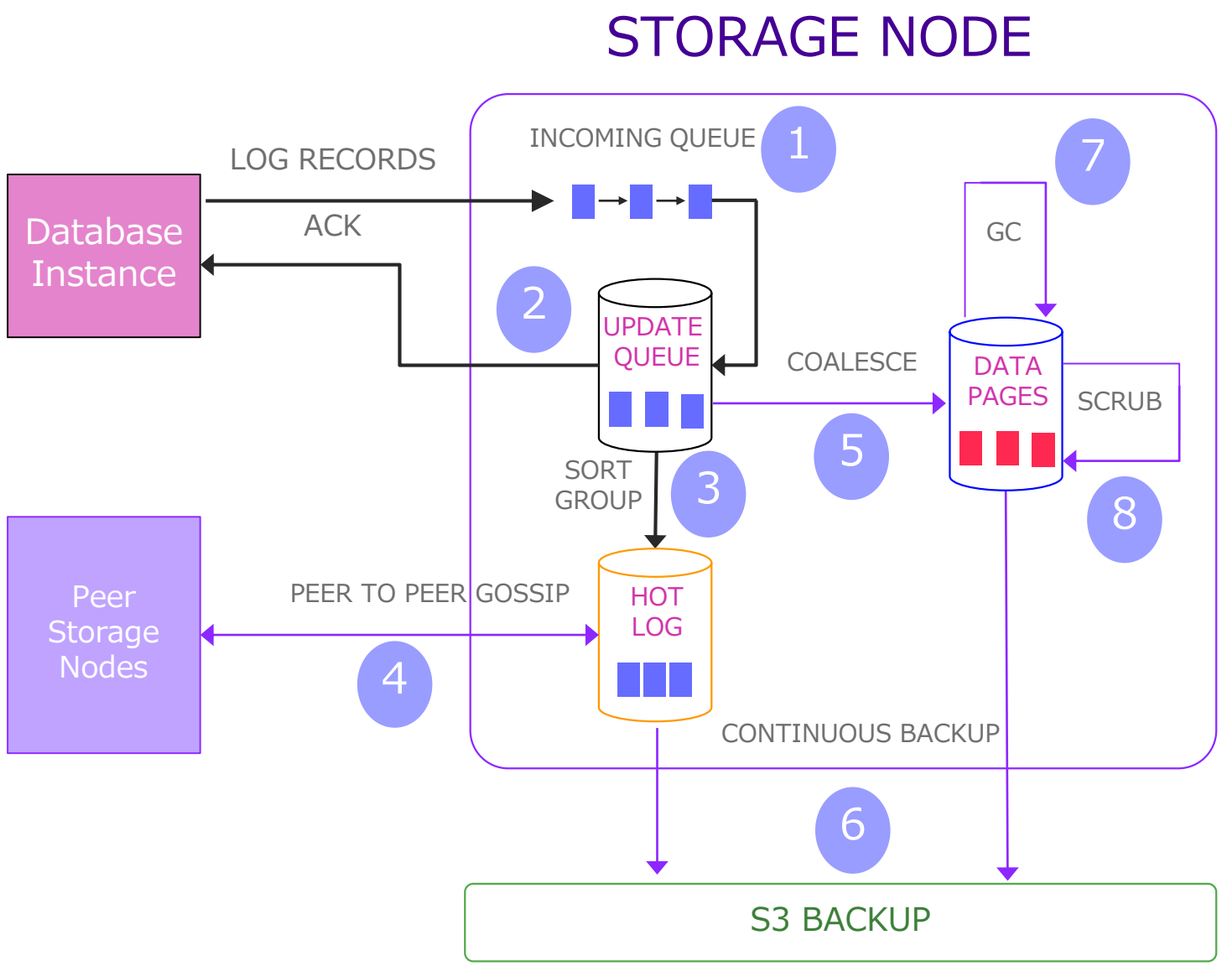
Durability: 4 5 6 0

Committed: ✓ ✓ ✓ ✓



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

I/O flow in Aurora storage node



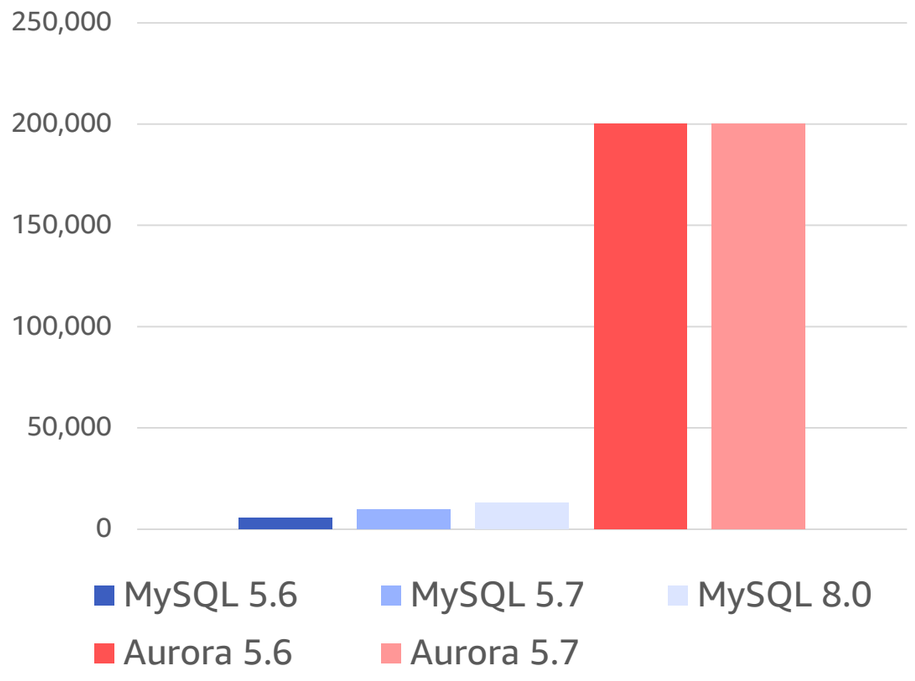
① ログレコードを受信してインメモリキューに追加
② ホットログとACKでレコードを保持
③ レコードを整理し、ログのギャップを特定
④ ギャップを埋めるためにゴシッププロトコルを利用
⑤ ログレコードを新しいページバージョンに結合
⑥ 定期的にログと新しいページのバージョンをS3に保存
⑦ 定期的に古いバージョンをガベージコレクション
⑧ ブロックのCRCコードを定期的に検証

Notes:
すべてのステップは非同期
ステップ1と2だけがフォアグラウンド待ち時間がある

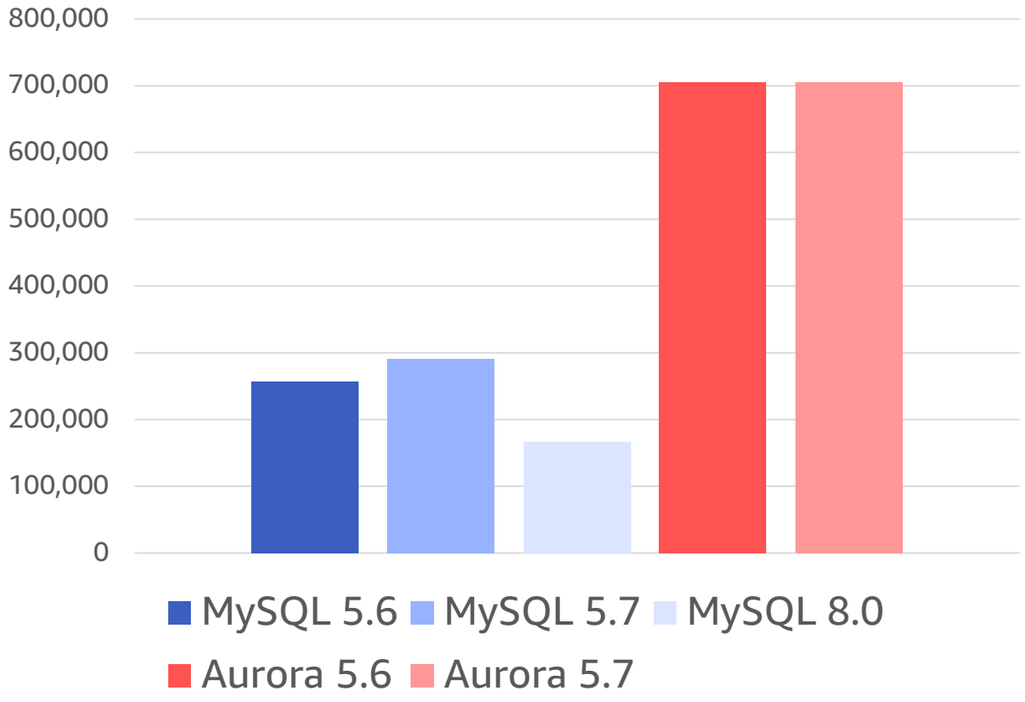
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Write and read throughput

Aurora MySQL is 5x faster than MySQL



Write Throughput



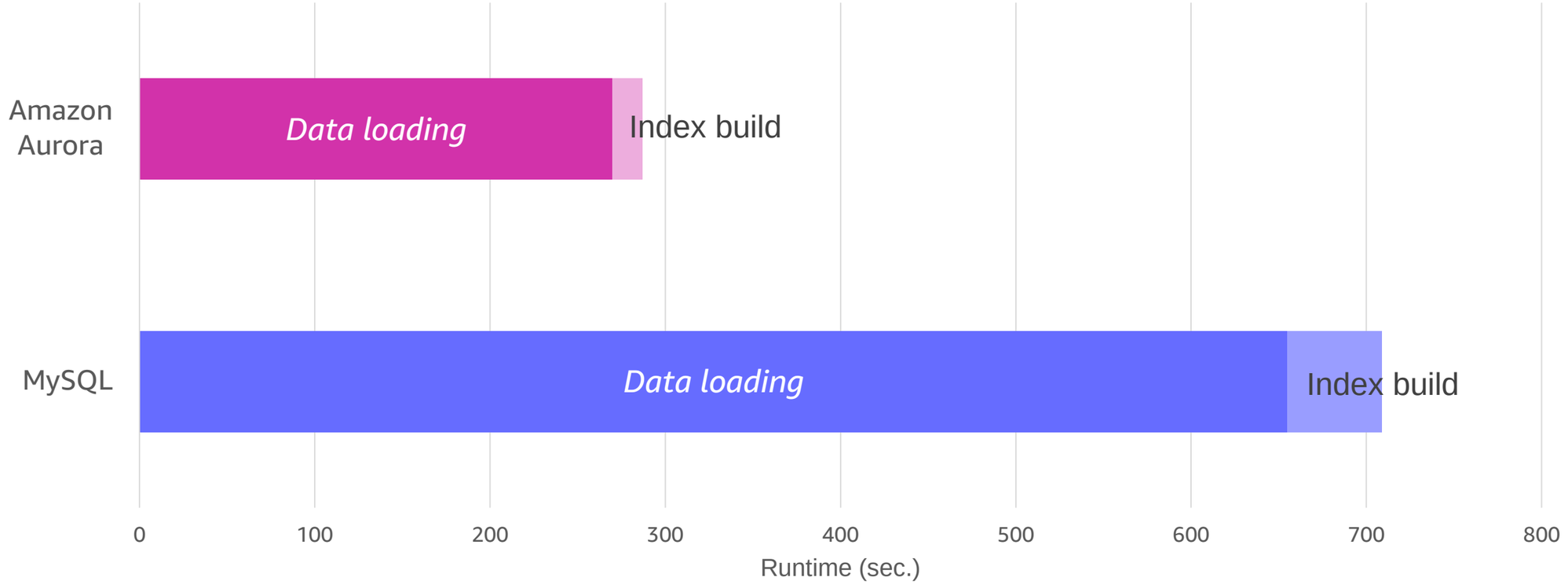
Read Throughput

Using Sysbench with 250 tables and 200,000 rows per table on R4.16XL

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Bulk data load performance

Aurora MySQL loads data 2.5x faster than MySQL

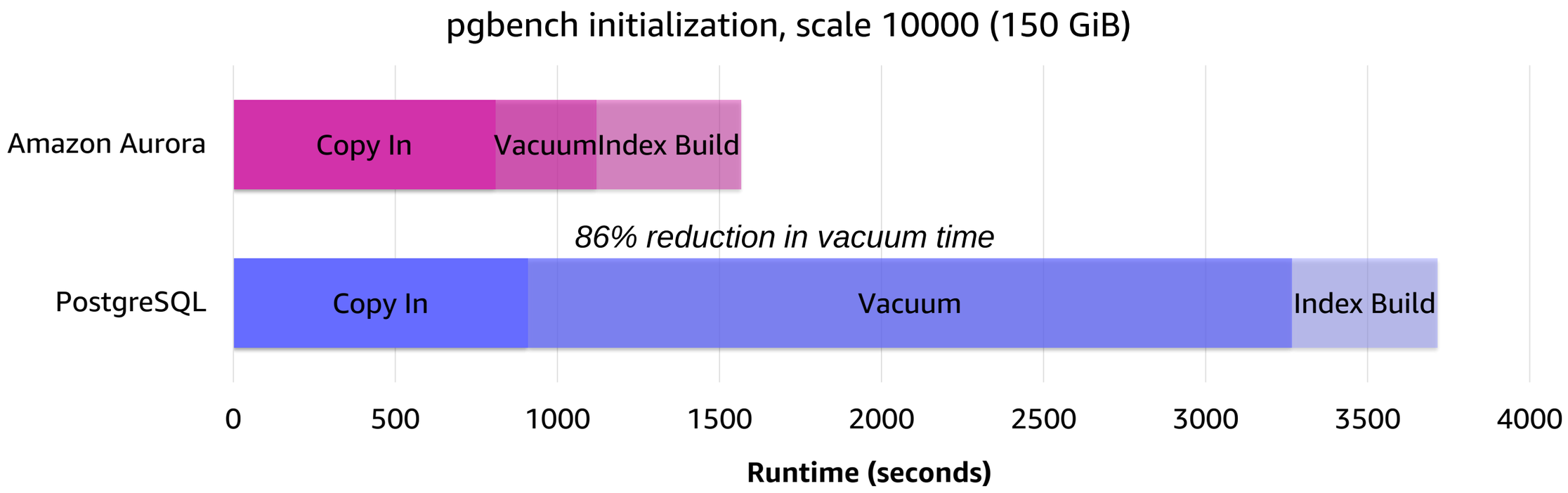


10 Sysbench Tables, 10MM rows per each

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Bulk data load performance

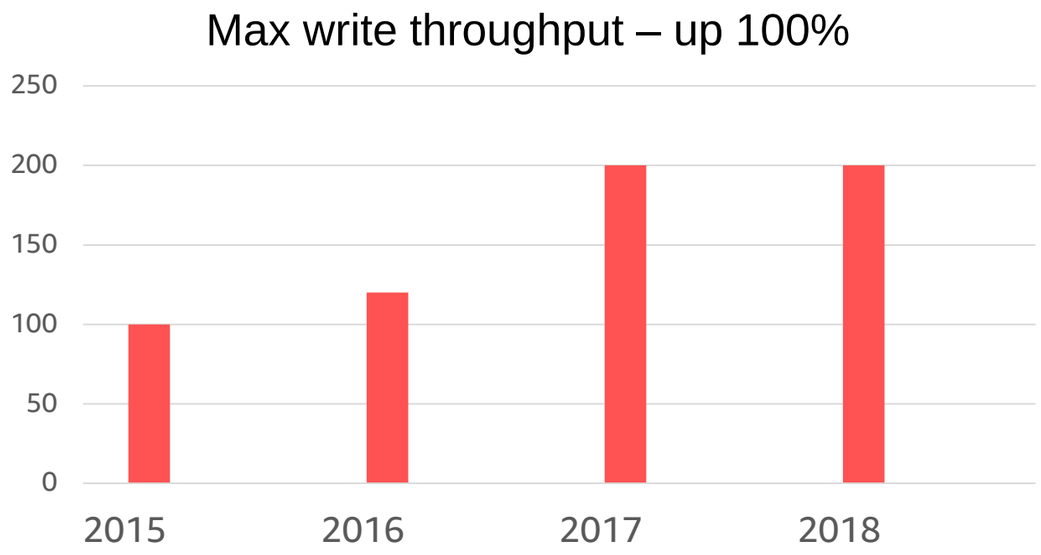
Aurora PostgreSQL loads data 2x faster than PostgreSQL

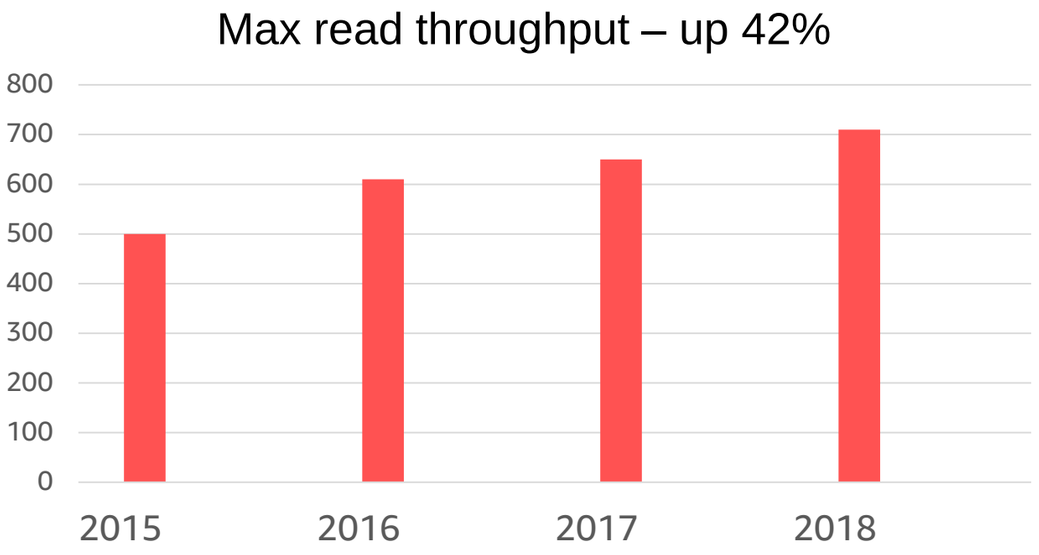


© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Continuous HW upgrades to boost performance

Aurora MySQL – 2015 to 2018





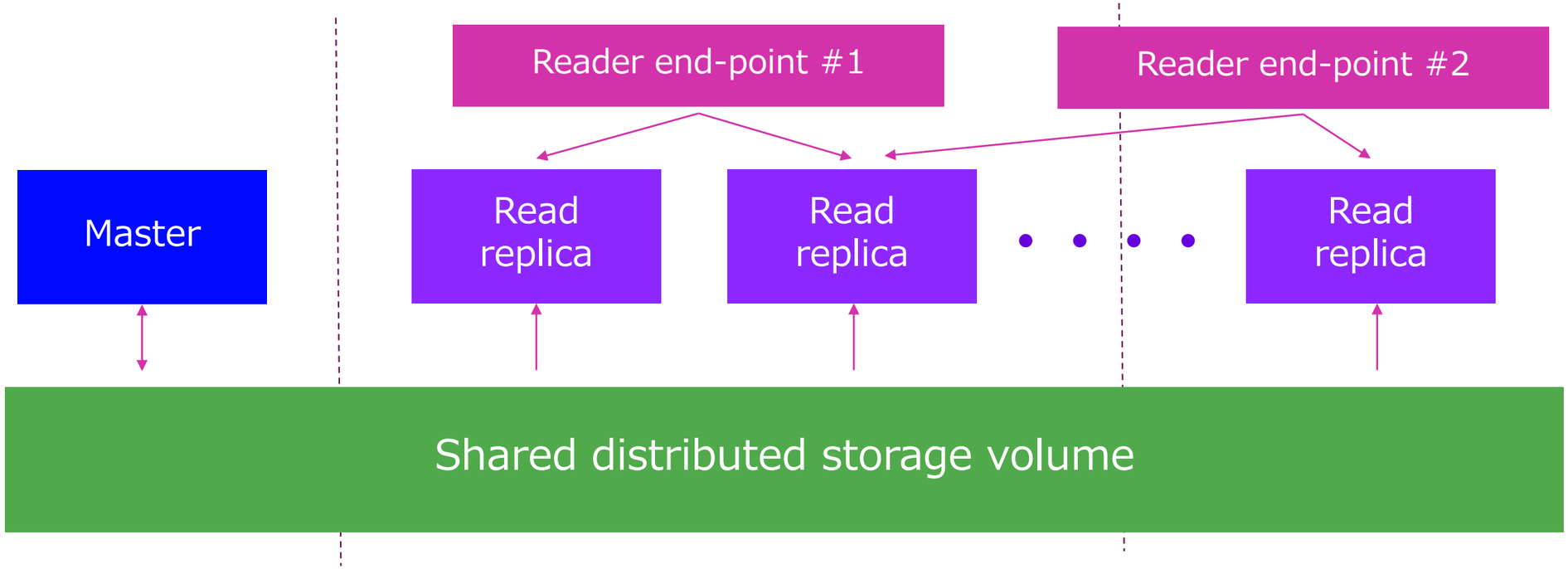
R3.8xl (2015)
32 cores, 256-GB memory

R4.16xl (2017)
64 cores, 512-GB memory

R5.24xl (new in 2019)
96 cores, 768-GB memory

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora Read Replicas



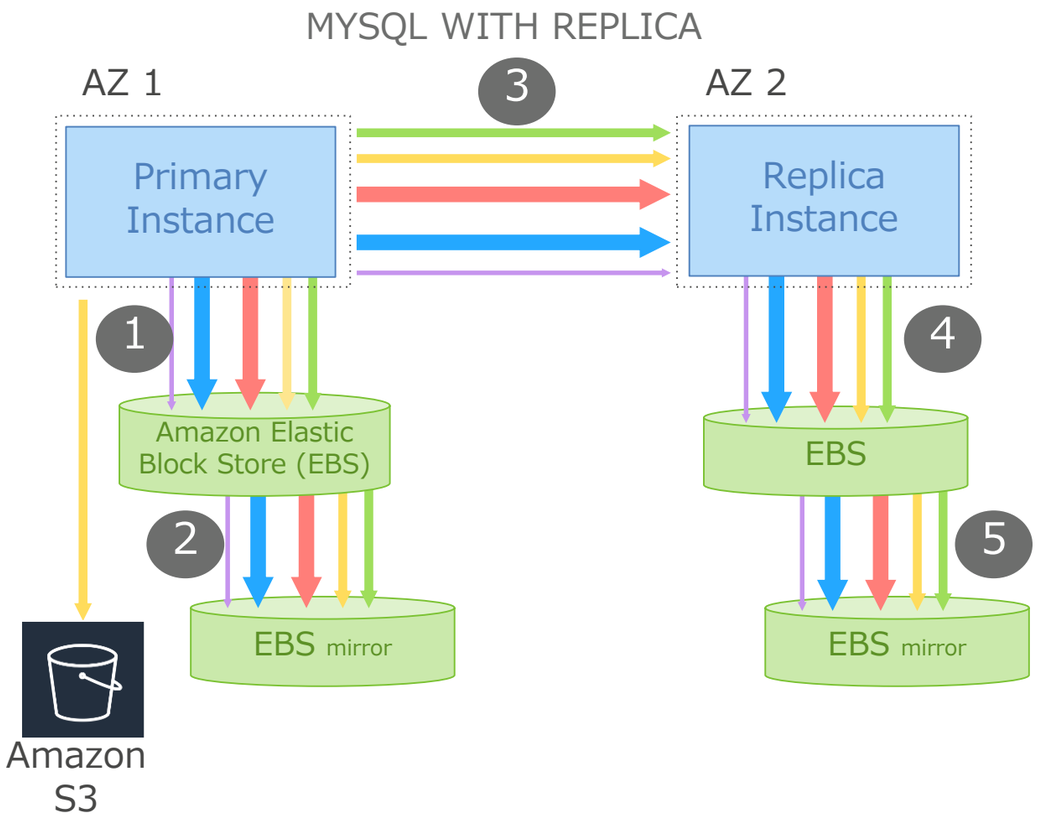
複数のアベイラビリティーゾーンに最大15の昇格可能なリードレプリカ

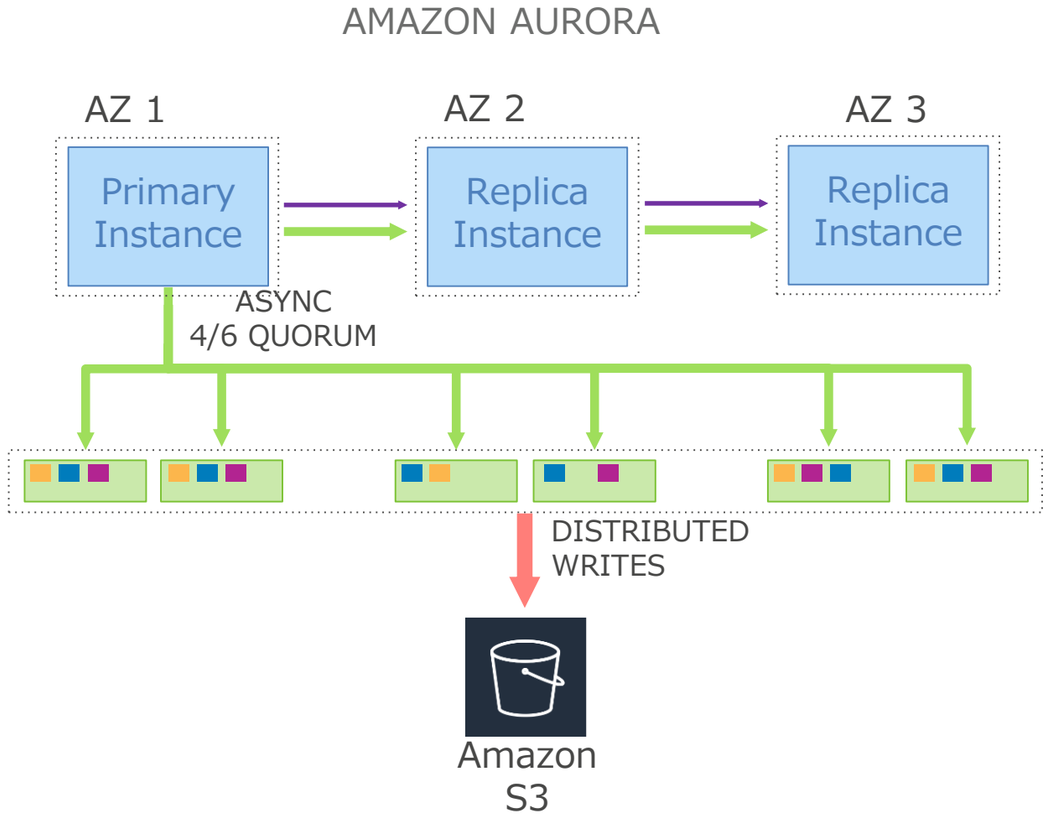
REDOログベースの（物理）レプリケーションにより、レプリカの遅延が低い（通常10ms未満）

フェイルオーバーの順序を設定可能なカスタムリーダーエンドポイント

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

MySQL vs. Aurora I/O profile





MySQL I/O profile for 30 min Sysbench run

0.78MM transactions

7.4 I/Os per transaction

Aurora IO profile for 30 min Sysbench run

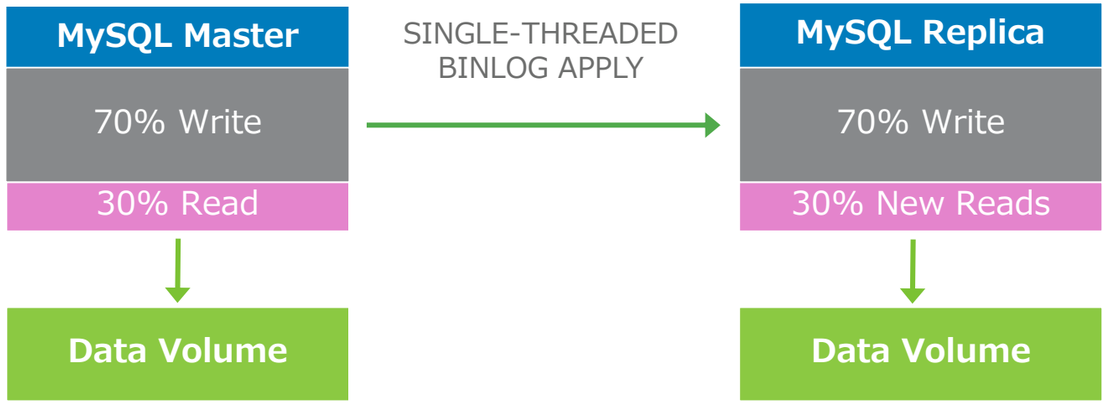
27MM transactions	35X	MORE
0.95 I/Os per transaction	7.7X	LESS

TYPE OF WRITE

LOG · BINLOG · DATA · DOUBLE-WRITE · FRM FILES

Aurora read replicas are dedicated to reads

MYSQL READ SCALING

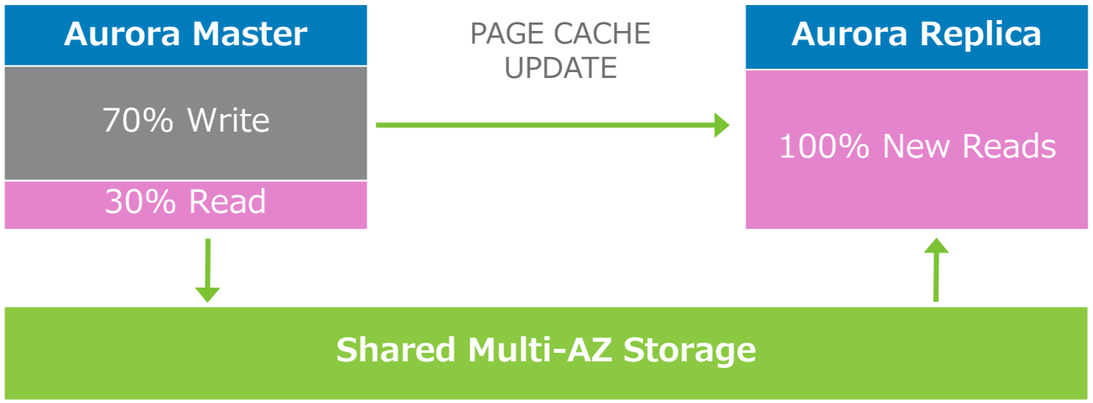


論理的な 完全な 変更

マスターと同じ書き込み ワークロード

独立した ストレージ

AMAZON AURORA READ SCALING



物理的な 差分の 変更

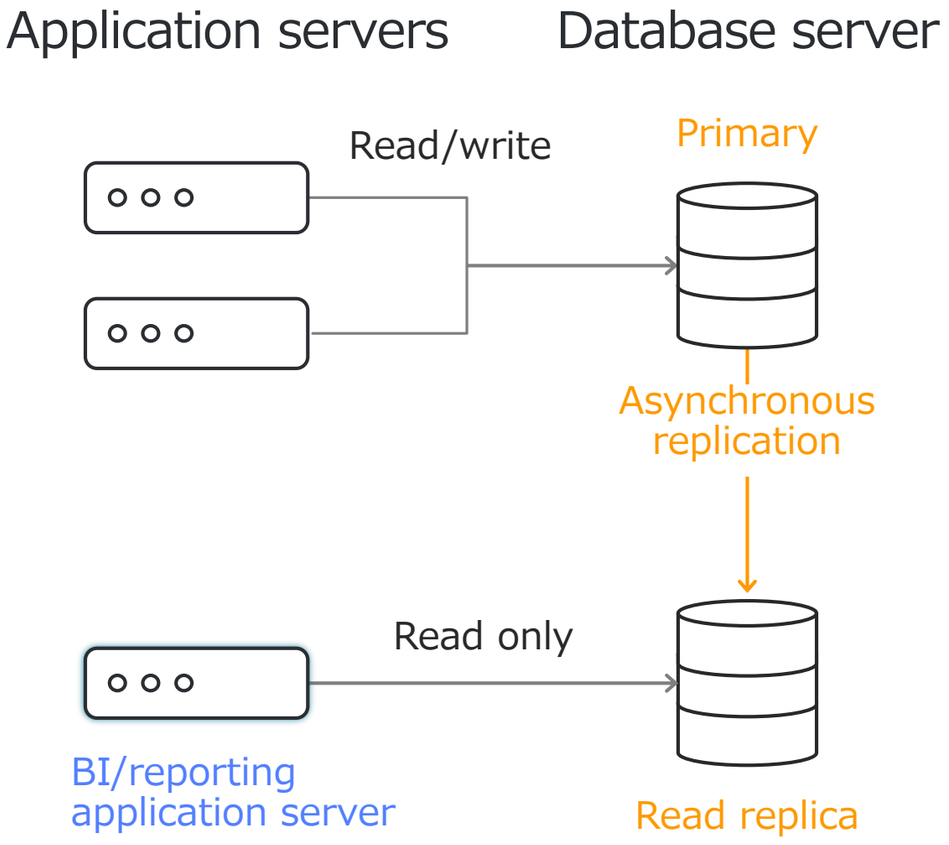
書き込みが発生しない レプリカ

シェアード ストレージ

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora read scaling options

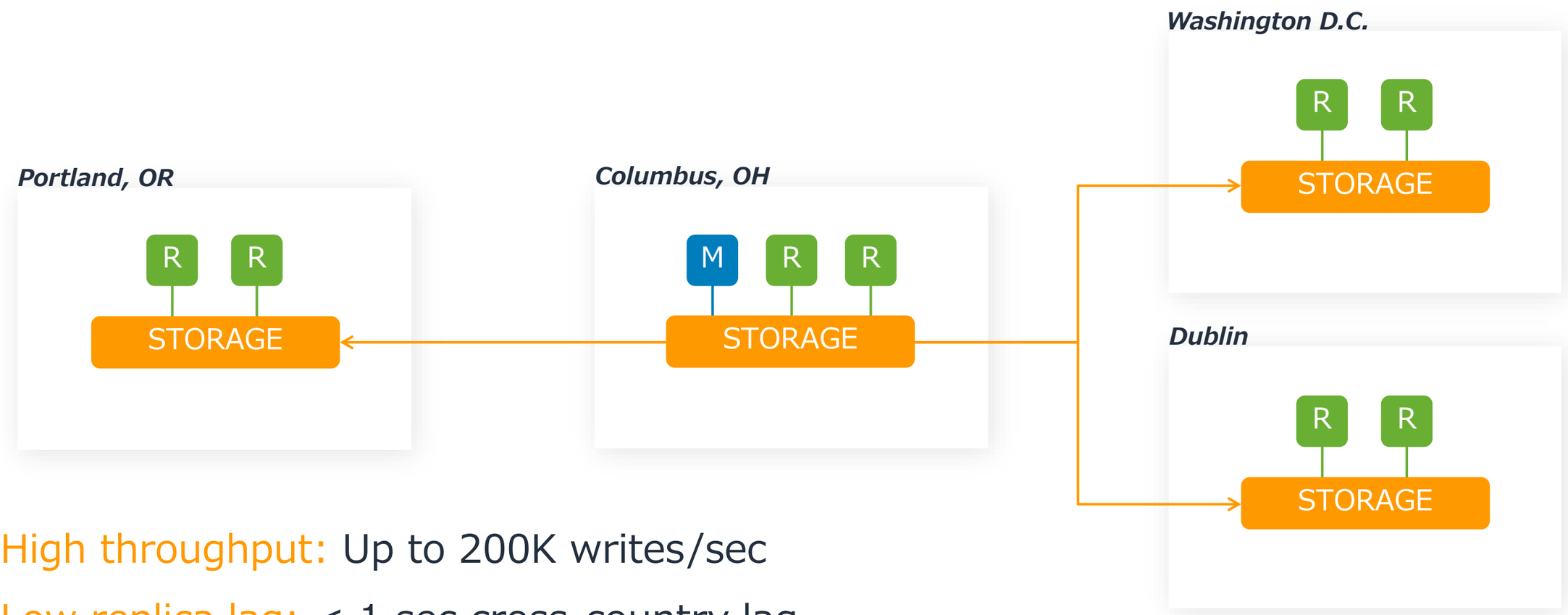
- クラスタ毎に昇格可能な15台のリードレプリカ
- 自動的に追加・削除を行うAuto-scalingレプリカ
- リージョンを跨いだリードレプリカ(Aurora Global Database)
- MySQLからbinlogを利用したレプリケーション



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora Global Database

Faster disaster recovery and enhanced data locality



High throughput: Up to 200K writes/sec

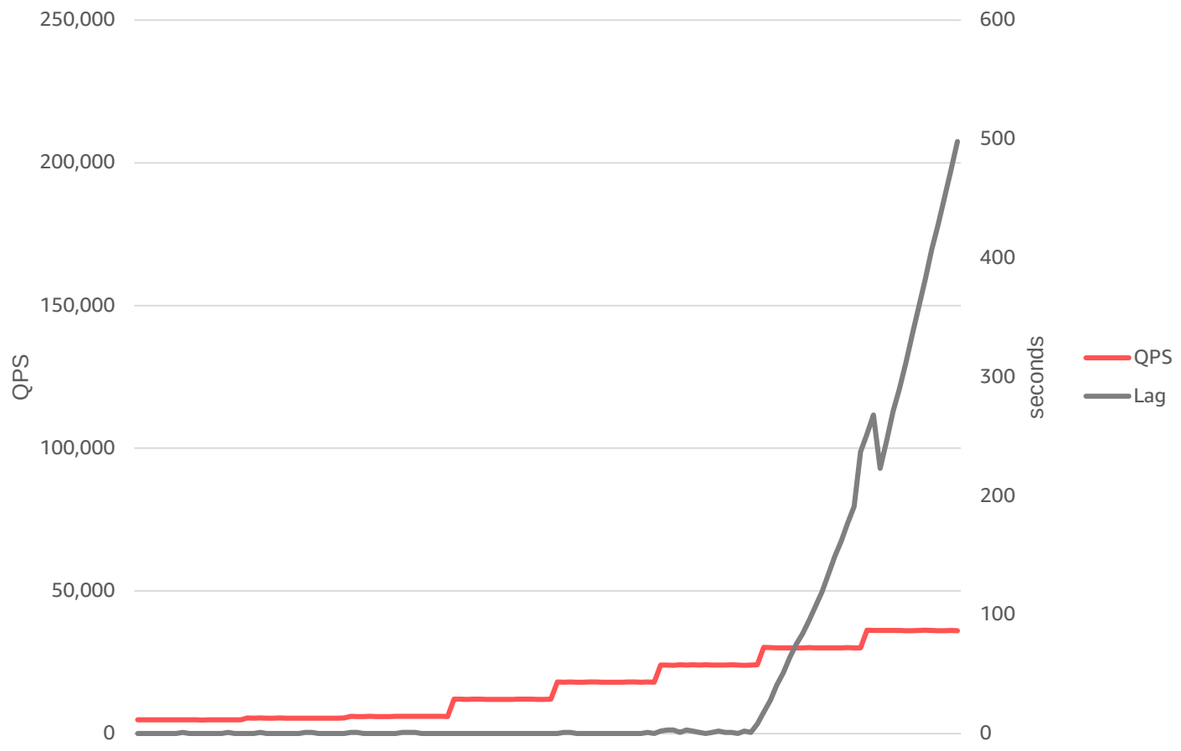
Low replica lag: < 1 sec cross-country lag

Fast recovery: < 1 min region unavailability

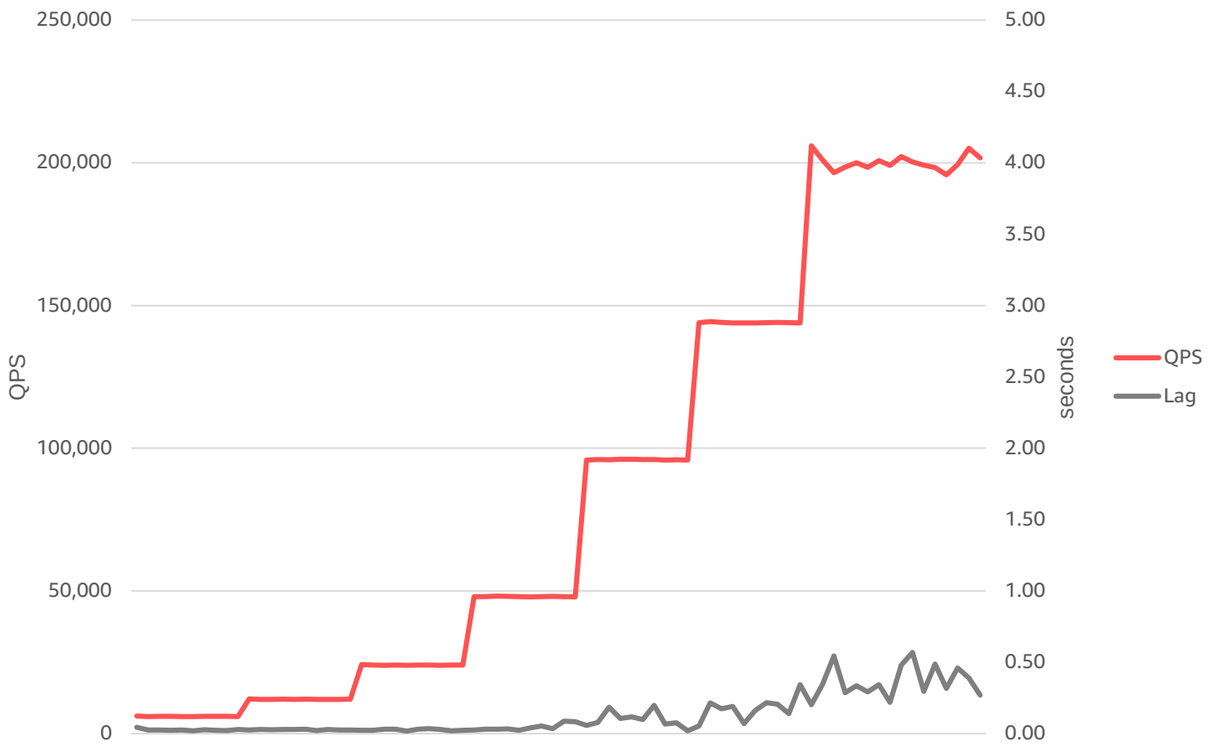
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Global replication performance

Logical vs. physical replication



Logical replication with MTS



Physical replication

SysBench OLTP (write-only) stepped every 600 seconds on R4.16xlarge

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora Parallel Query

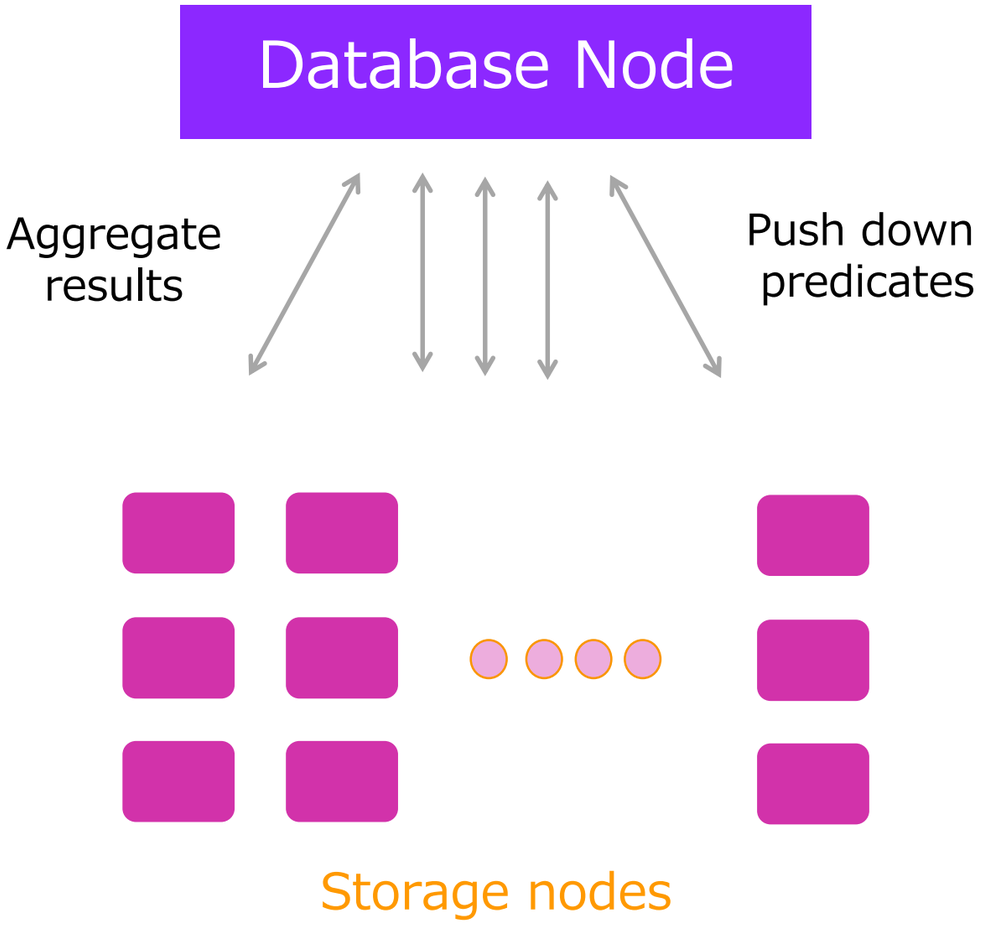
Speeding up analytical queries

Auroraストレージには数千のCPUが存在する

- クエリ処理をプッシュダウンして並列化
- 処理をデータの近くに移動することでネットワークトラフィックと待ち時間を削減

実現のためには多くのチャレンジ

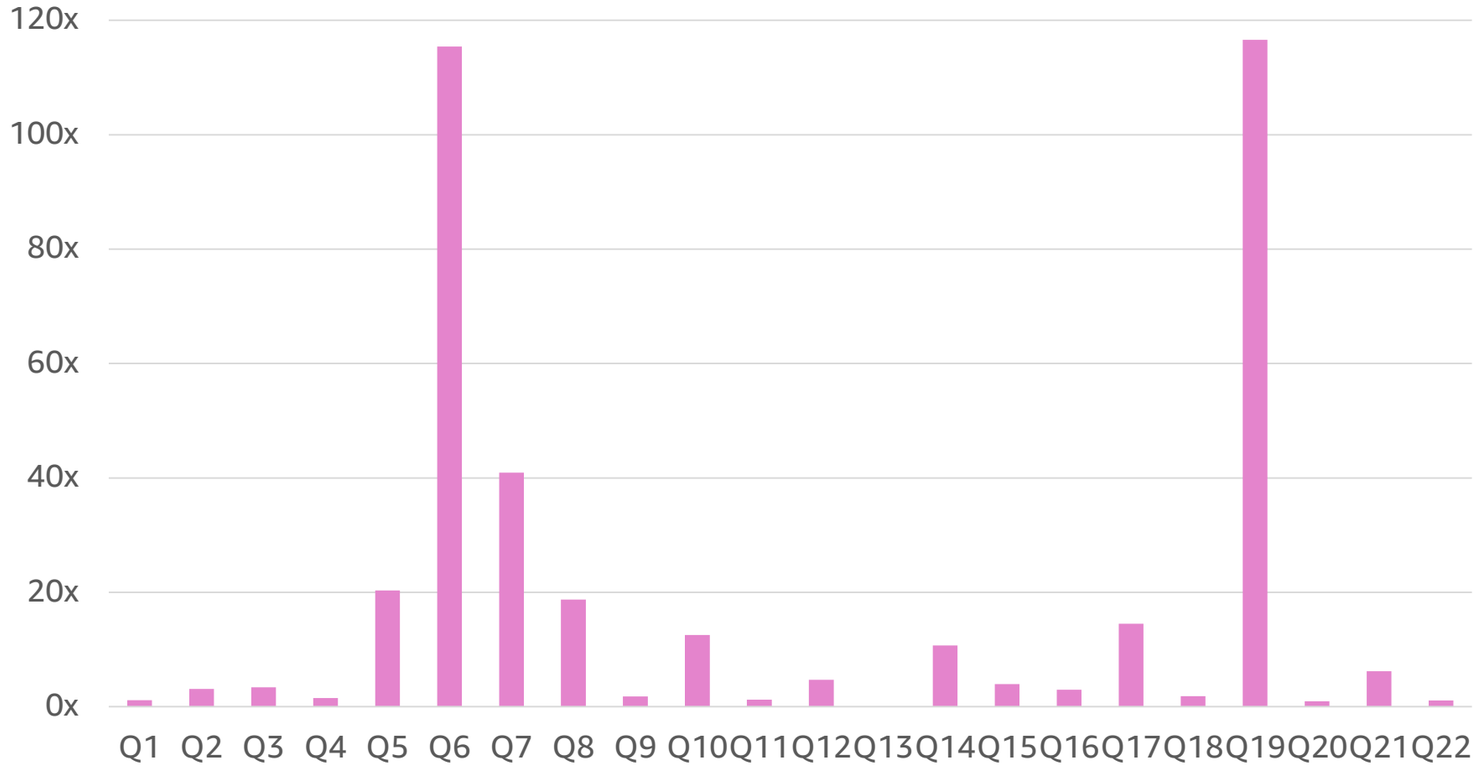
- データは範囲分割されていない - フルスキャンが必要
- データは実行中の可能性がある リードビューでは最新のデータを表示できない場合がある
- すべての機能をプッシュダウンできるわけではない



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Parallel Query performance results

Well-known decision support benchmark



Query response time reduction

- Peak speed up ~120x
- >10x speedup: 8 of 22 queries

NETFLIX

Auroraのパラレルクエリ機能をテストし、パフォーマンス向上は非常に優れていました。インスタンスタイプをr3.8xlargeからr3.2xlargeに減らすことができました。このユースケースでは、パラレルクエリは私たちにとって大きな利点でした。

Jyoti Shandil, Cloud Data Architect

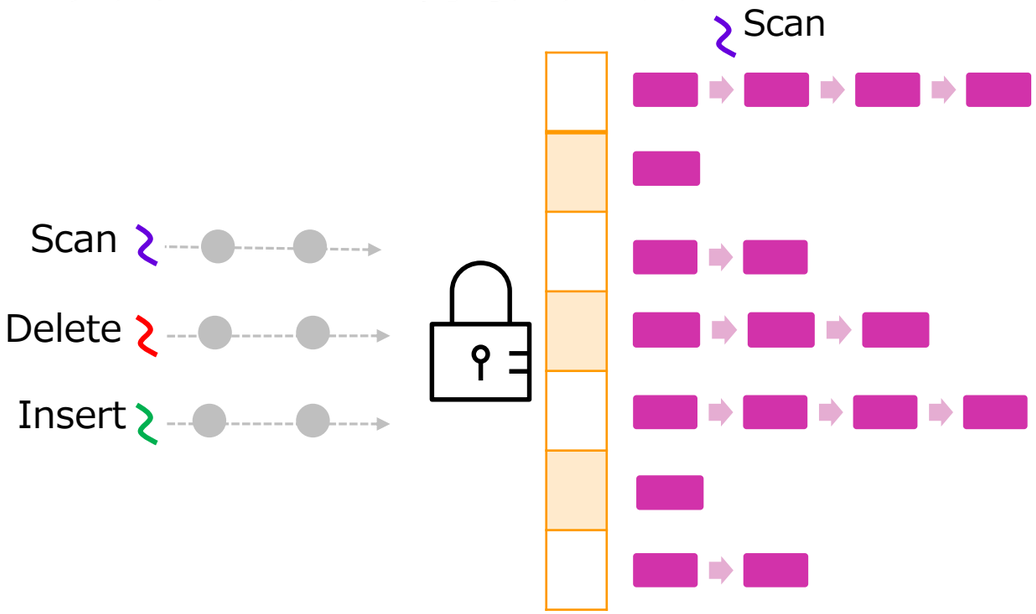
aws

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora lock management

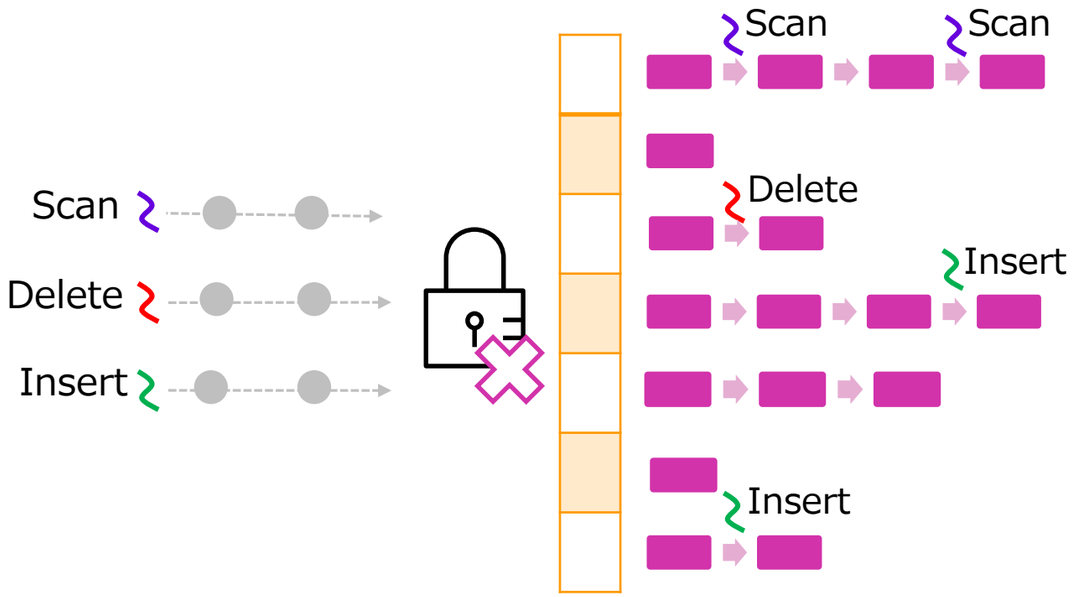
MySQL lock manager

MySQLと同じロックセマンティクス
ロックチェーンへの同時アクセス



Aurora lock manager

個々のロックチェーン内の複数のスキャナ
ロックフリーデッドロック検出

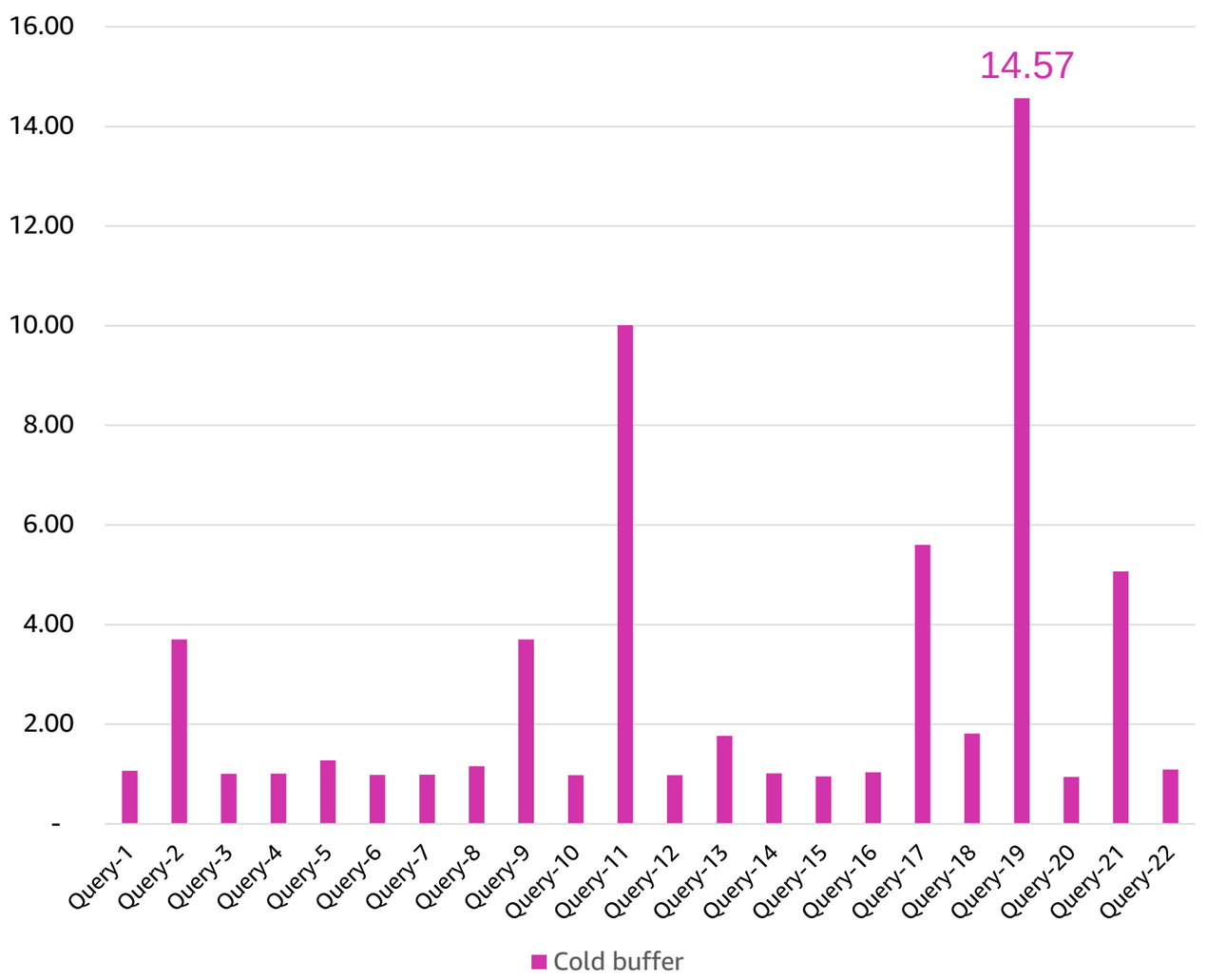


多くの同時セッションをサポートする、高い更新スループット

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Asynchronous key prefetch

- バッチ・キー・アクセス（BKA）結合アルゴリズム+マルチレンジ読み取り（MRR）最適化を使用したクエリに利用可能
- JOIN評価中にセカンダリ・プライマリインデックスを実行
- バックグラウンドスレッドを使用して、非同期でページをメモリにロード



Latency improvement factor vs. Batched Key Access (BKA) join algorithm. Decision support benchmark, R3.8xlarge

AKP used in queries 2, 5, 8, 9, 11, 13, 17, 18, 19, 20, 21

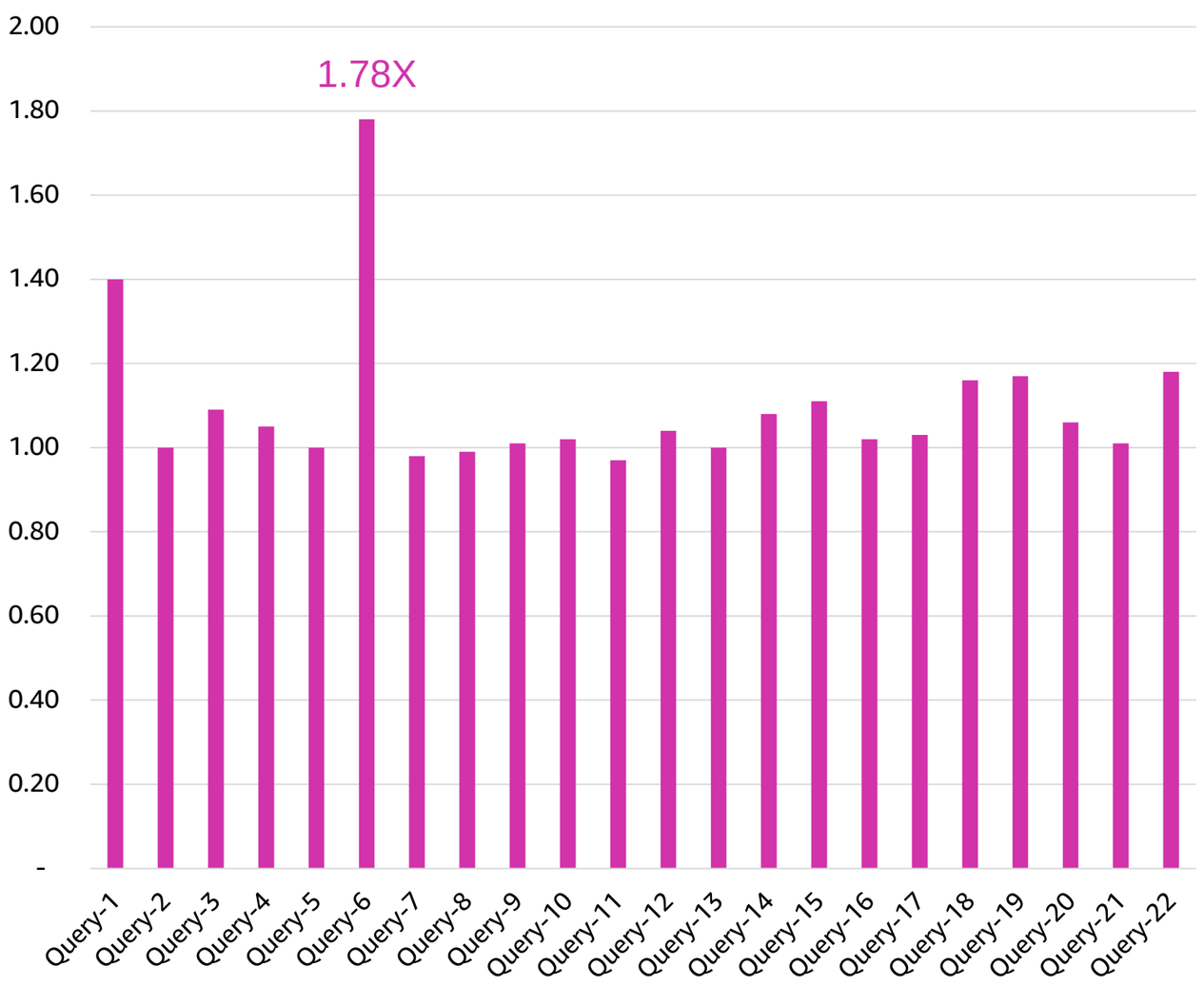
Batched scans

標準のMySQLでは1行毎に処理を行うためオーバーヘッドが大きくなる

- functionの読み出し
- ロックとラッチ
- カーソルの保存とフェッチ
- InnoDB から MySQL フォーマットへの変換

Amazon AuroraはInnoDB buffe poolからタプルをバッチで読み込む

- Table full scans
- Index full scans
- Index range scans



Latency improvement factor vs. Batched Key Access (BKA) join algorithm. Decision support benchmark, R3.8xlarge

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora availability & manageability

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

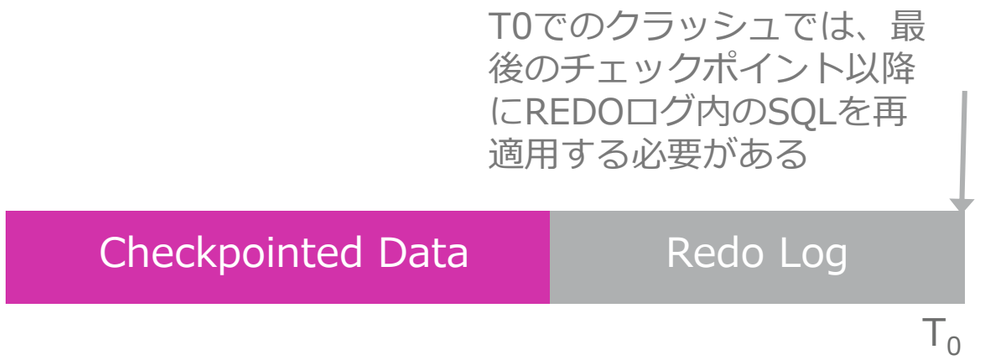
Instant crash recovery

Traditional database

最後のチェックポイント以降からログを適用する必要がある

チェックポイント間は通常5分

MySQLではシングルスレッド。大量のディスクアクセスが必要

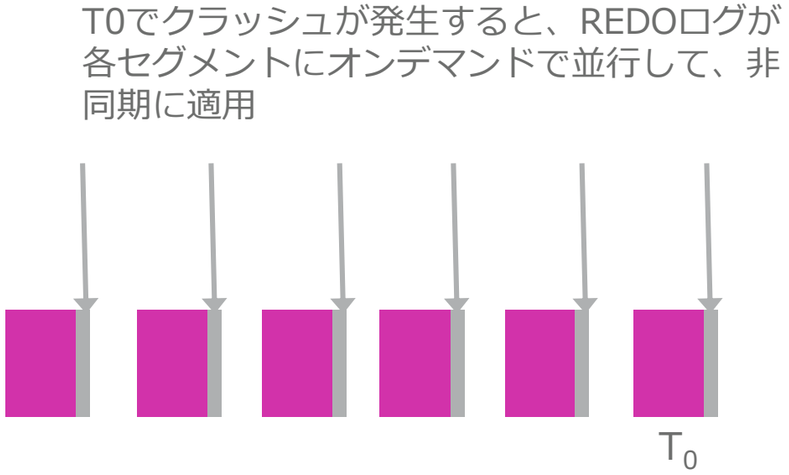


Amazon Aurora

ストレージがディスク読み取りの一部としてREDOレコードをオンデマンドで再適用

並列、分散、非同期

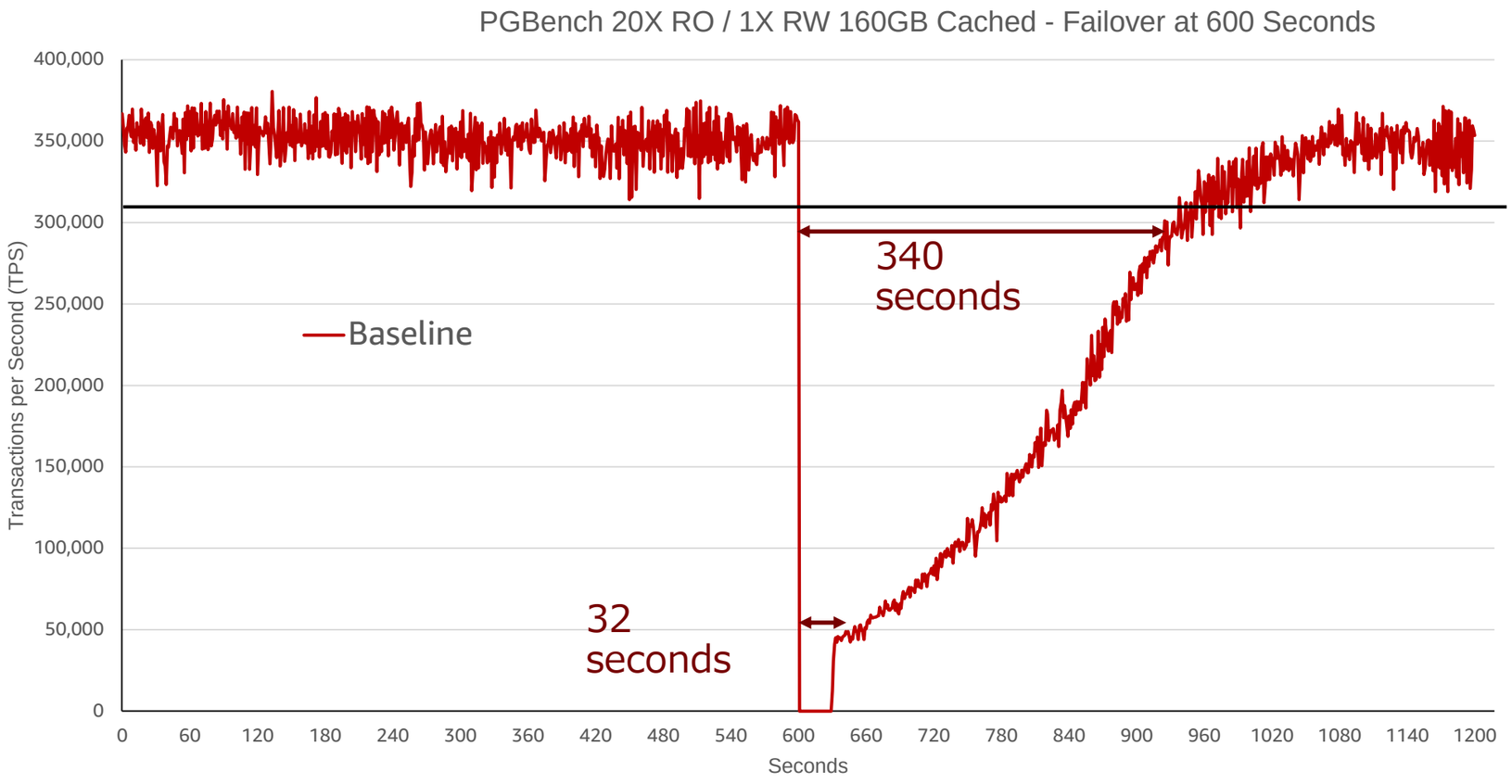
起動時に再適用なし



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

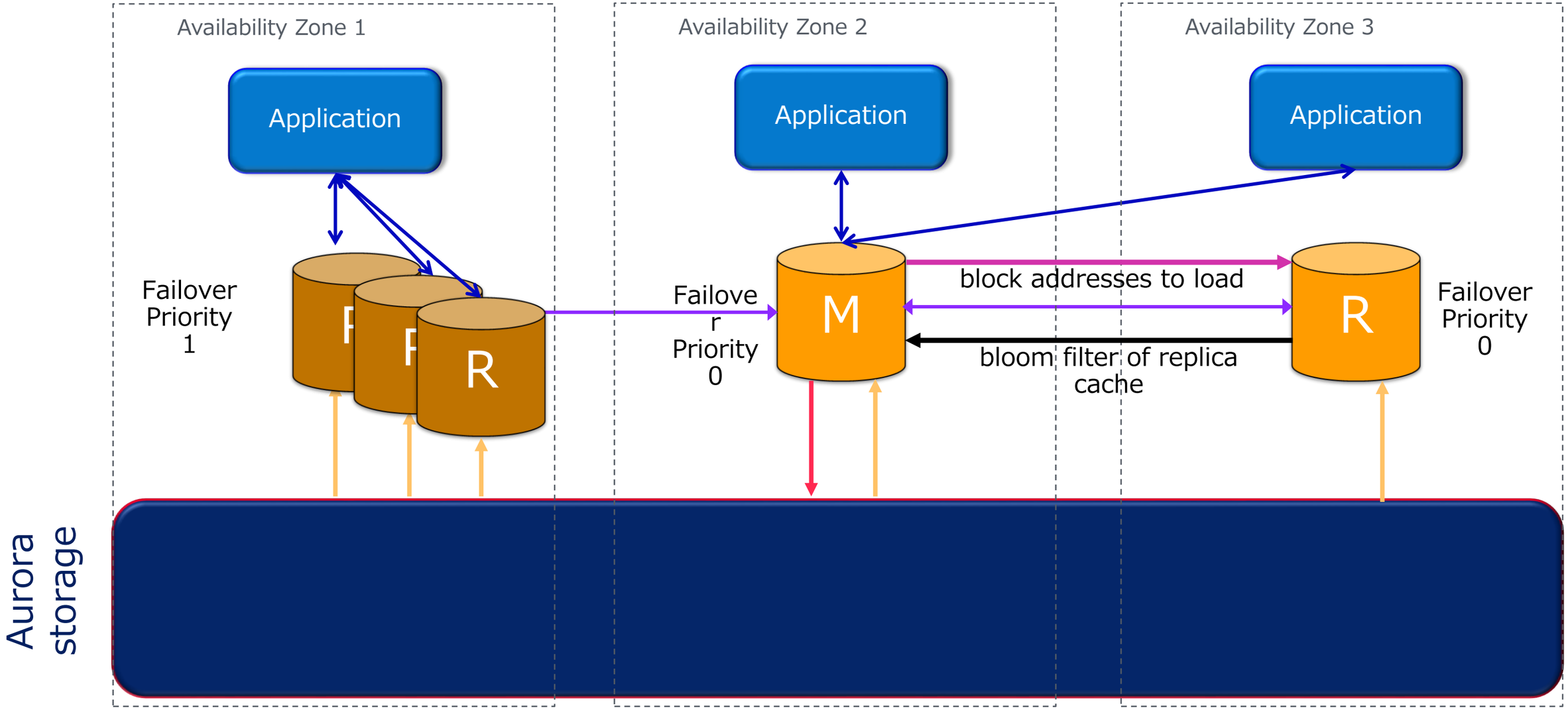
Cluster Cache Management – Aurora PostgreSQL

Avoiding a cold cache after failover



通常、データベースをバックアップしてキャッシュをウォームアップするにはしばらく時間がかかります。このフェールオーバーの例では、CCMがないと、DBの起動に32秒かかりましたが、パフォーマンスの90%を回復するには340秒でした

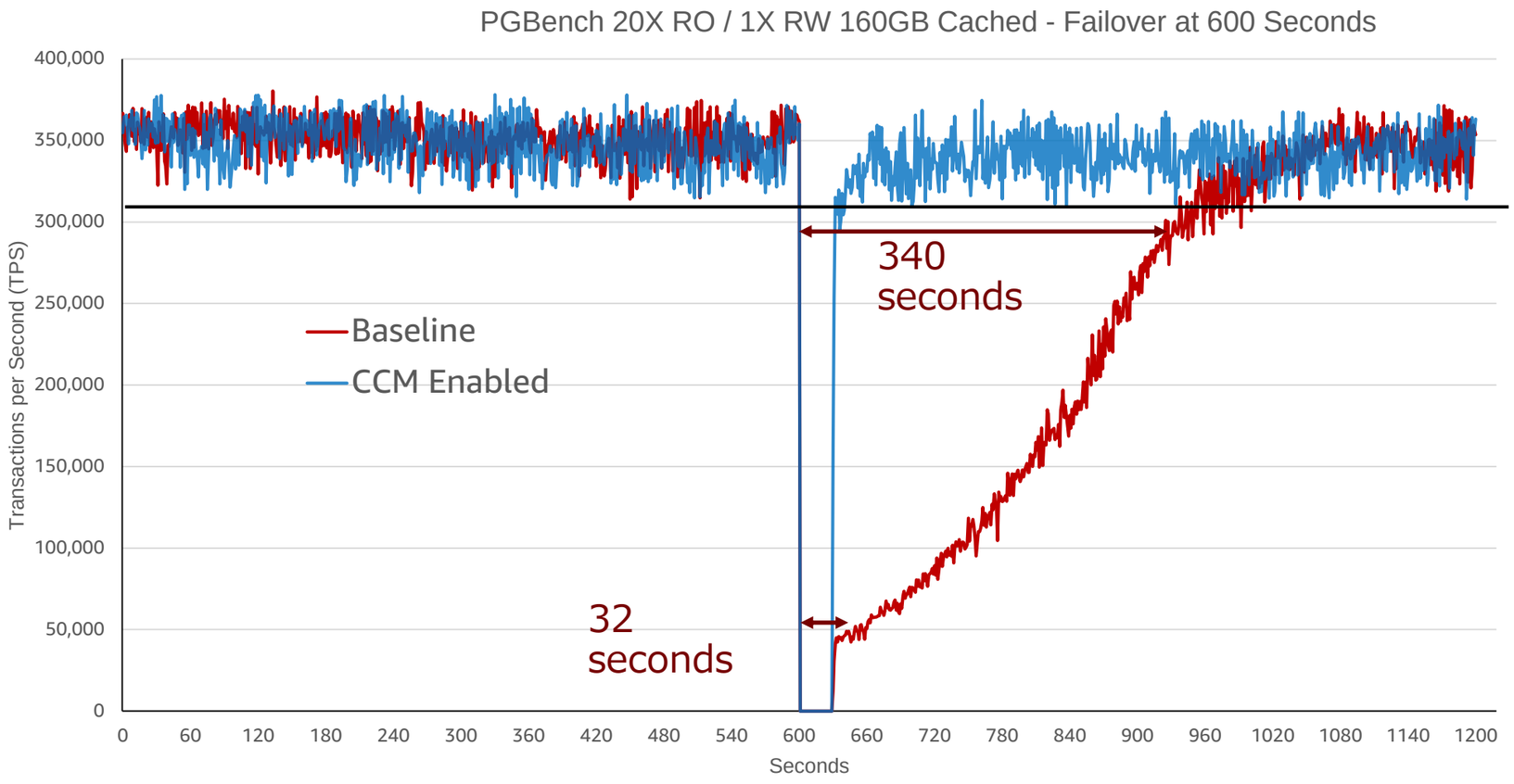
How Cluster Cache Management works



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Cluster Cache Management – Aurora PostgreSQL

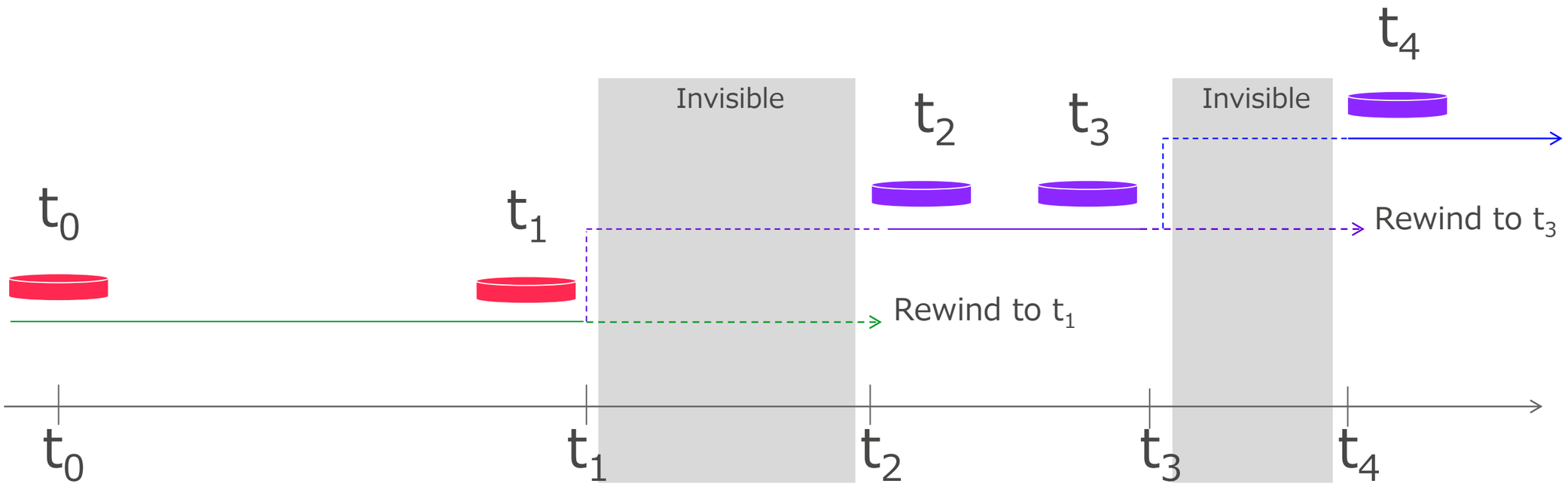
Failover to an (almost) fully warm cache



CCMが有効になっていると、データベースはウォームアップされたキャッシュにフェイルオーバ。フェイルオーバーから32秒後に、パフォーマンスの90%に復帰

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora Backtrack

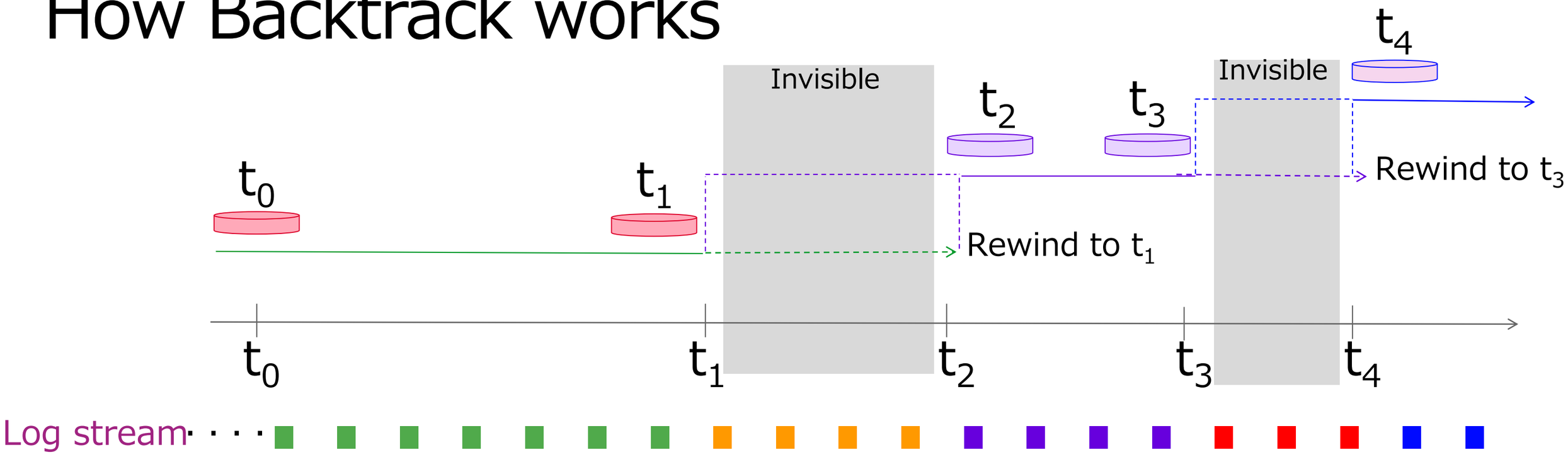


Backtrackにより、バックアップからの復元を必要とせずにデータベースを特定の時点に戻すことが可能

- 意図しないDMLまたはDDL操作から回復
- Backtrackは破壊的ではないため、正しい時点を見つけるために複数回Backtrackすることが可能
- QA（テスト実行の間にDBを巻き戻す）にも役立ちます

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How Backtrack works



ログは、LSNツリー内の分岐に基づいて表示または非表示にする

- t_2でBacktrackを実施し、t_1 へ巻き戻す場合は オレンジ ログを不可視にする
- t_4 でBacktrackを実施し、t_3 へ巻き戻す場合は 赤ログを不可視にする

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

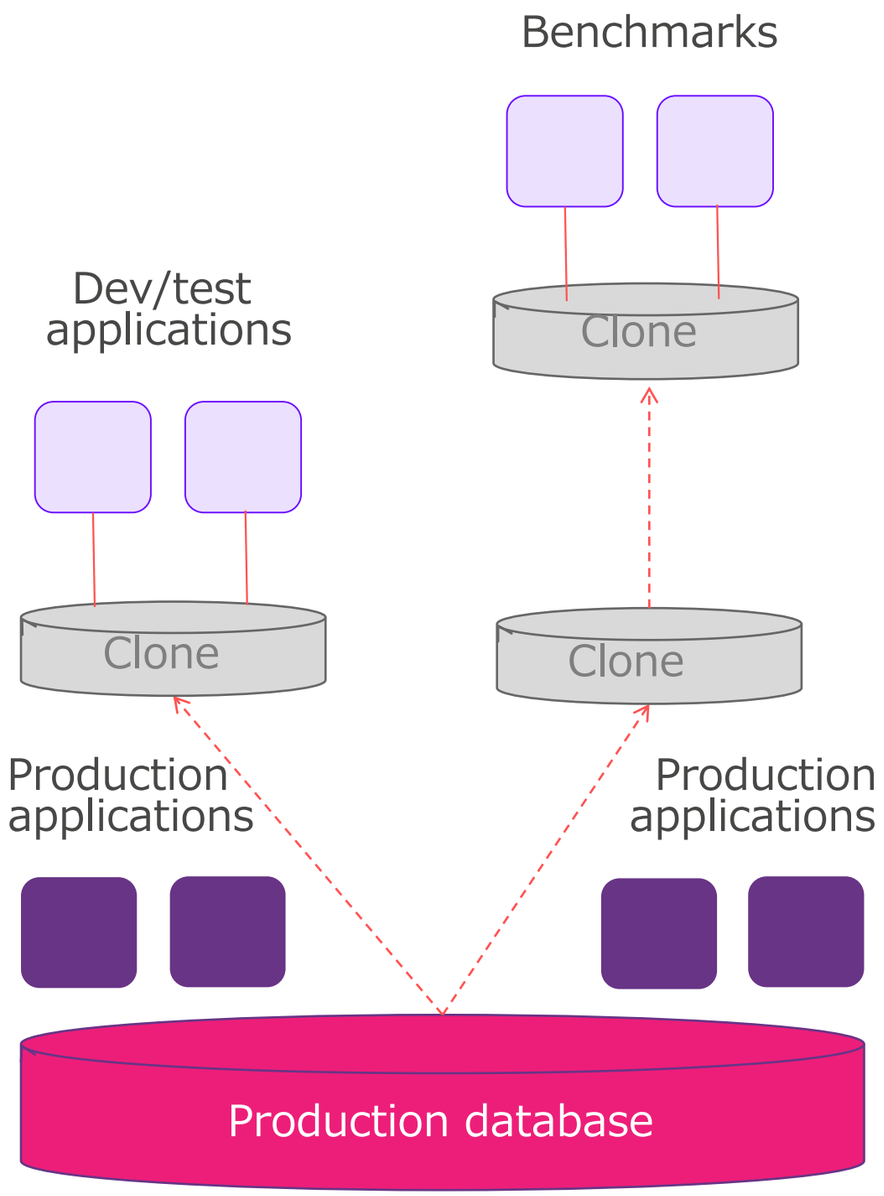
Fast Database Cloning

重複したストレージコストなしでデータベースのコピーを作成するCreation of a クローンはほぼ瞬間的なもの - データをコピーしない

- データのコピーは書き込み時にのみ行われる - 元のボリュームデータとクローンされたボリュームデータが異なる場合がある
- 最大15クローンまで作成可能

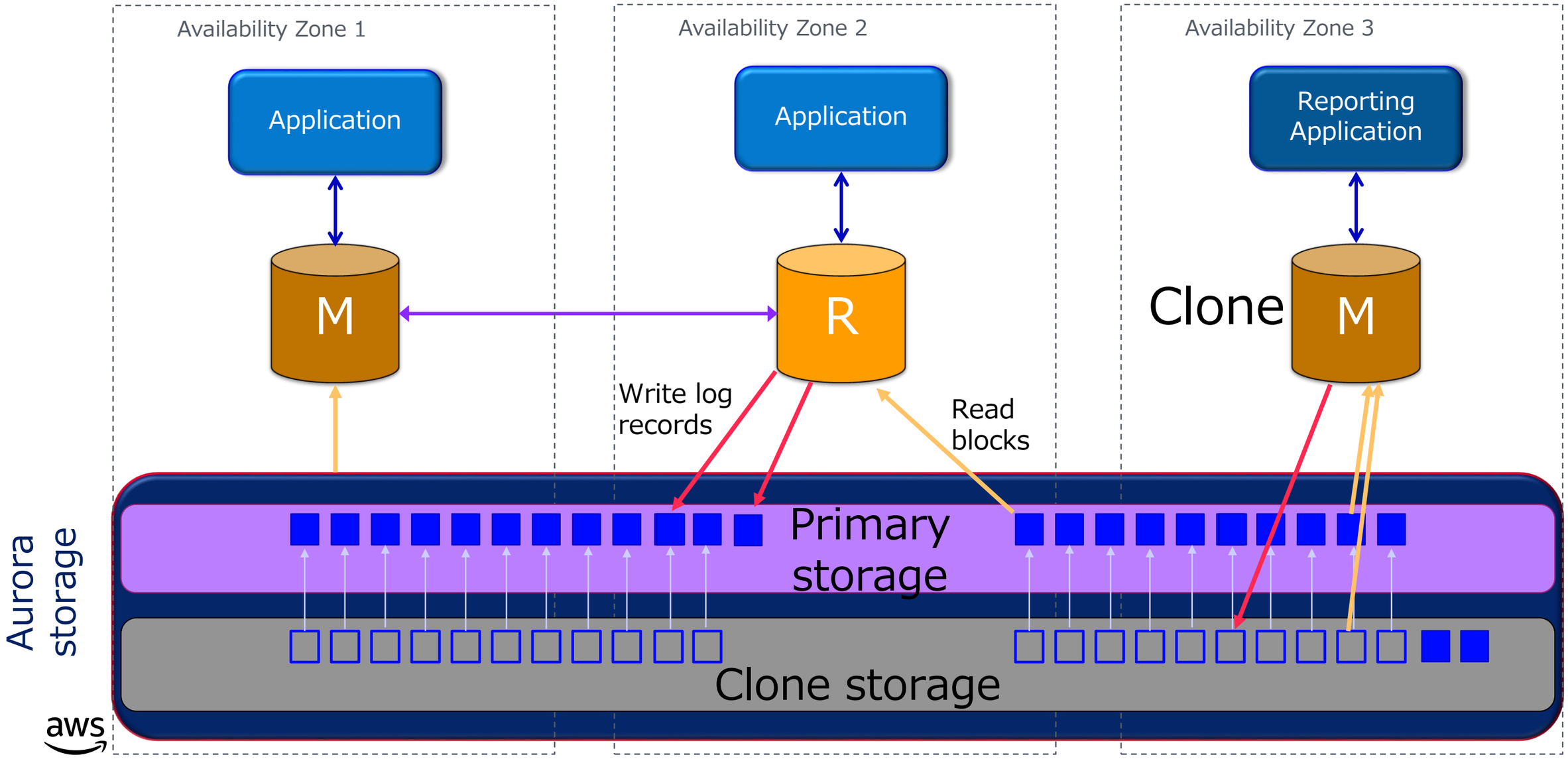
ユースケース

- 本番のデータベースをテスト目的で利用
- データベースの再構成
- 本番システムに影響を及ぼさず、分析目的でスナップショットDBを作成



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How Fast Database Cloning works



Aurora Serverless

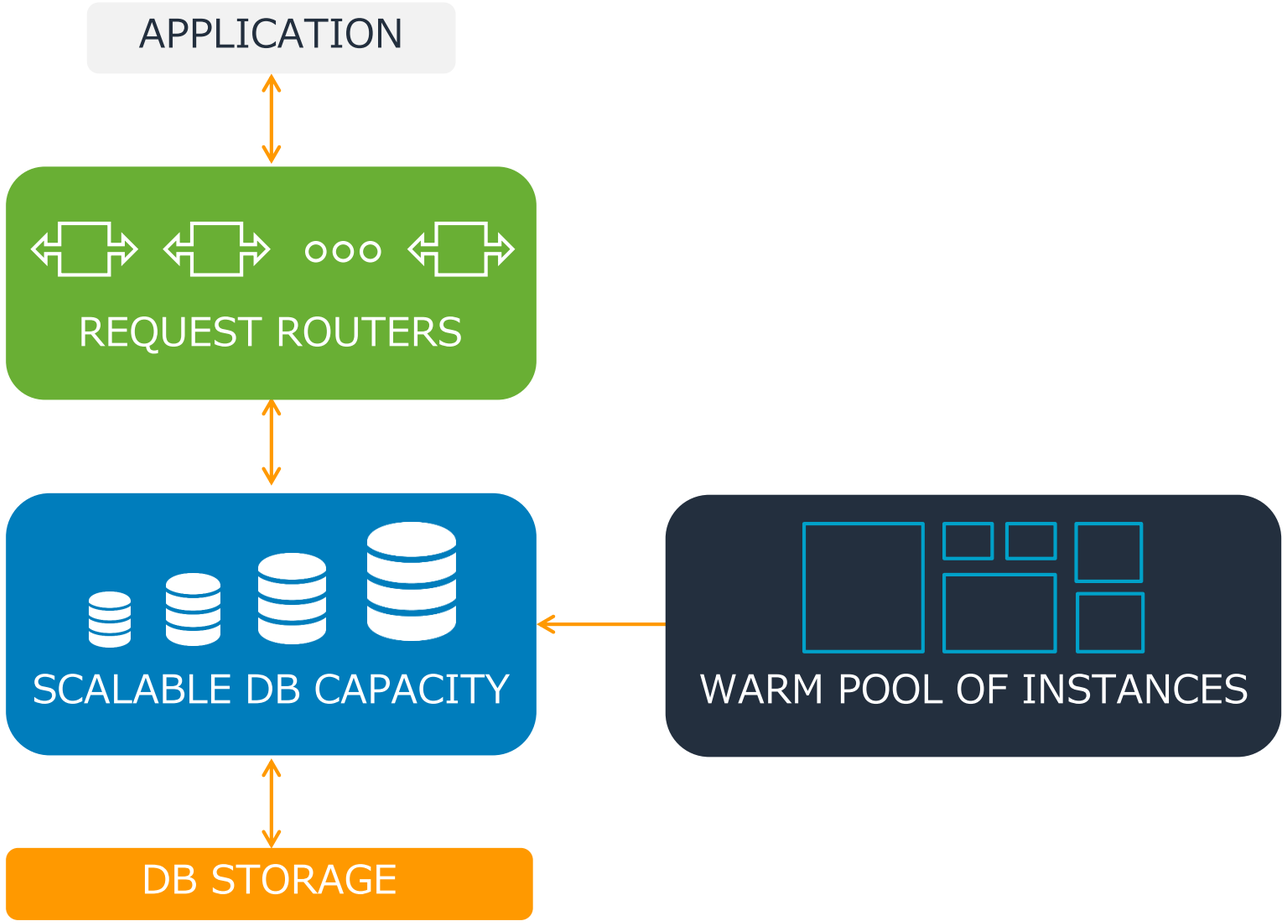
オンデマンドで起動し、使用していないときはシャットダウン

自動的にスケールアップ・ダウン

スケール時にアプリケーションへのインパクトなし

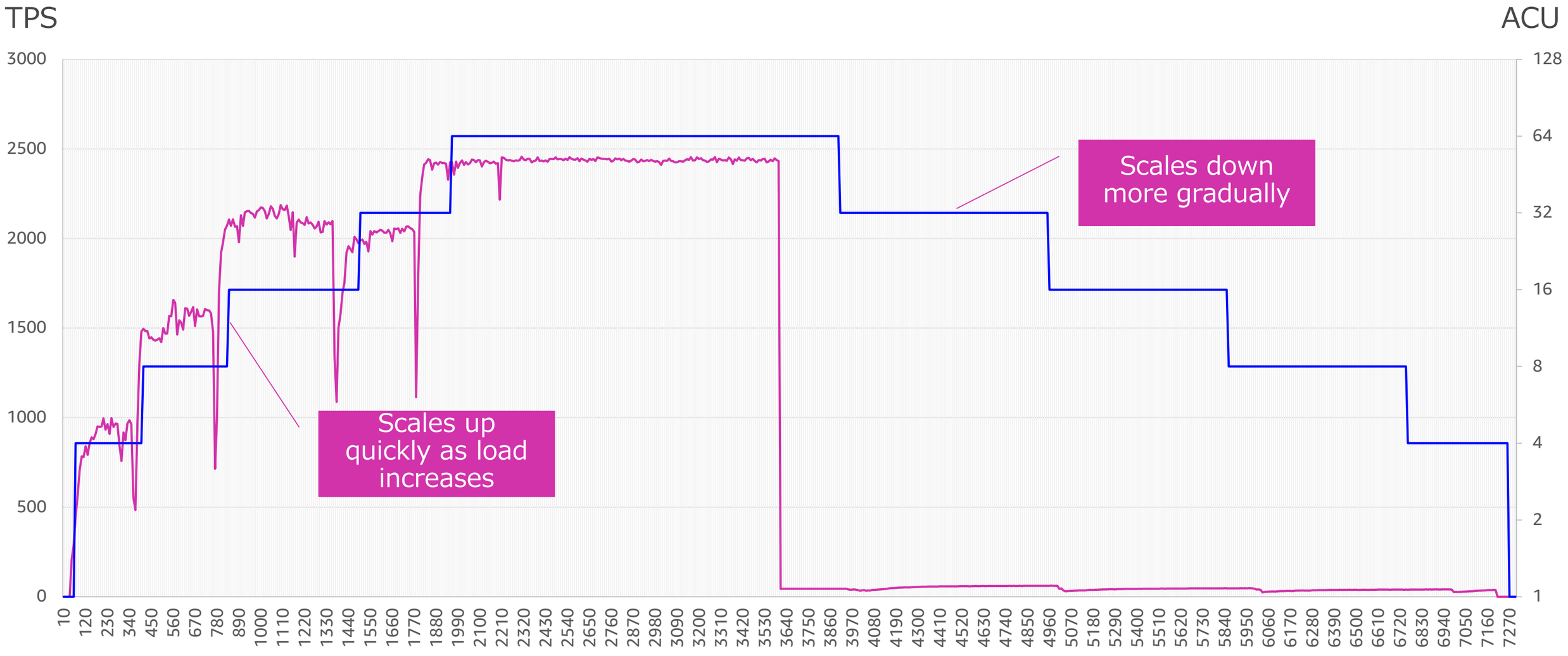
1分を最低時間に1秒単位の課金

使用頻度が低い、予測不可能、または周期的なワークロードに最適



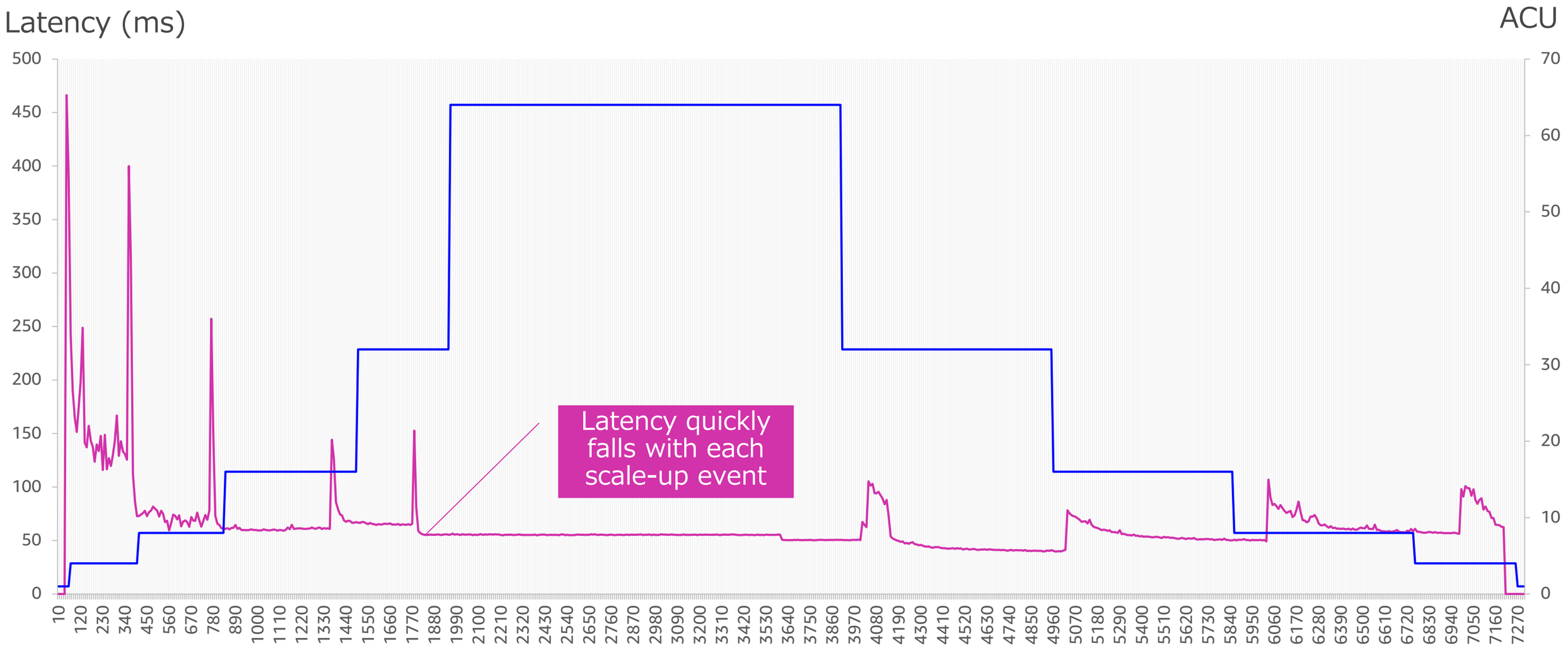
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora Serverless: scale up and down with load



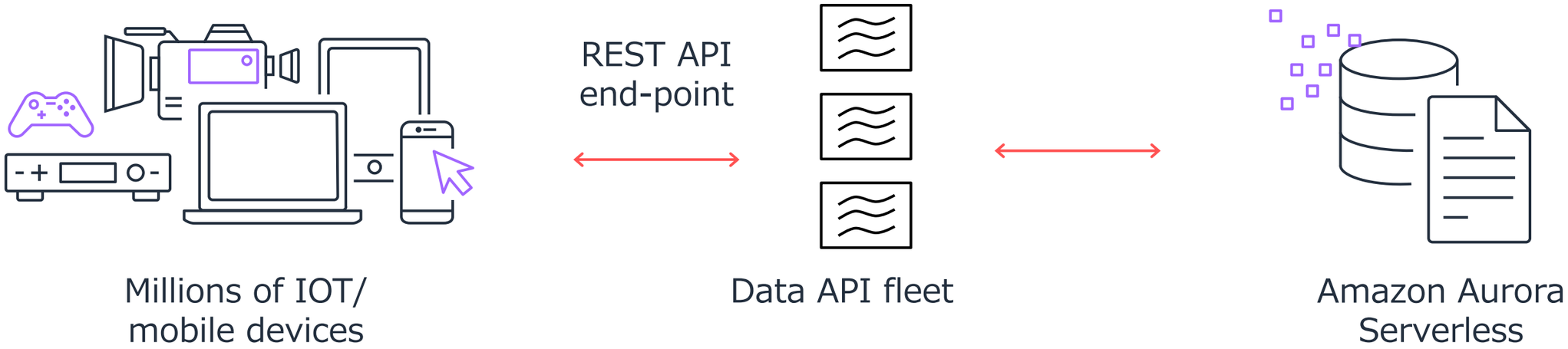
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora Serverless: stabilizing latency quickly



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon RDS Data API for serverless apps



Serverless apps often have restrictions

- DBへの限られたネットワーク接続
- DBへの持続的な接続なし
- 限られたリソースを持つスモールクライアント（例：IoT）

RDS Data API provides:

- HTTP経由でアクセス可能なパブリックエンドポイント
- クライアント設定なしでアクセス

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon runs on Aurora

Amazon fulfillment centers

3/29に、すべてのAmazonフルフィルメントセンター（FC）をOracleからAmazon Auroraに移行しました

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

本セッションのFeedbackをお願いします

お手元のサミットガイドブックの表紙、受講票にも記載している
『QRコード』からご回答ください。
もれなく素敵なAWSオリジナルグッズ＆アイスをプレゼントします。

涼感マフラータオル（巾着入り）

プレゼントの引き換えは、EXPOエリア内アンケートコーナー・出口付近のいずれかにお越しください。

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Yoav Eilat

yeilat@amazon.com

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.