
AWS
re:Invent

DAT201

What’s new in Amazon Aurora

Tony Petrossian

GM, Amazon Aurora

AWS

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon Aurora

COMMERCIAL-GRADE CLOUD-NATIVE DATABASE

Delivered as a **managed** service



- Drop-in compatibility with MySQL and PostgreSQL
- Simplicity and cost-effectiveness of open-source databases
- Simple pay-as-you-go pricing
- Throughput and availability of commercial databases

Innovating with Aurora



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

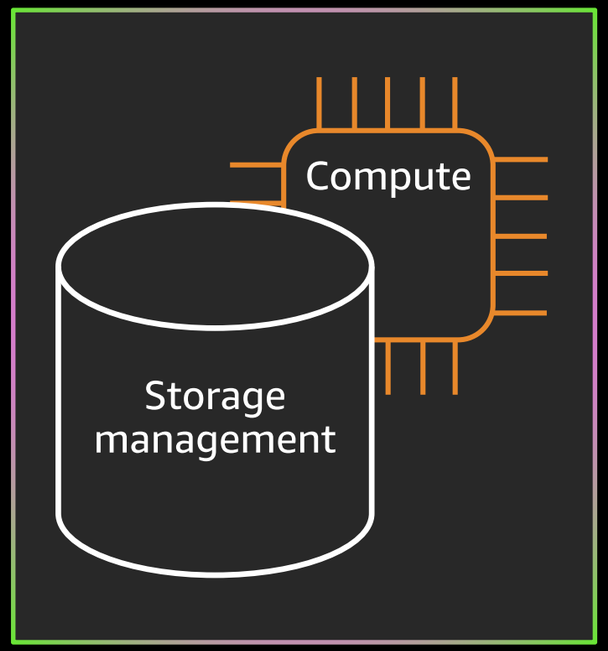
Database layers

SQL

Transactions

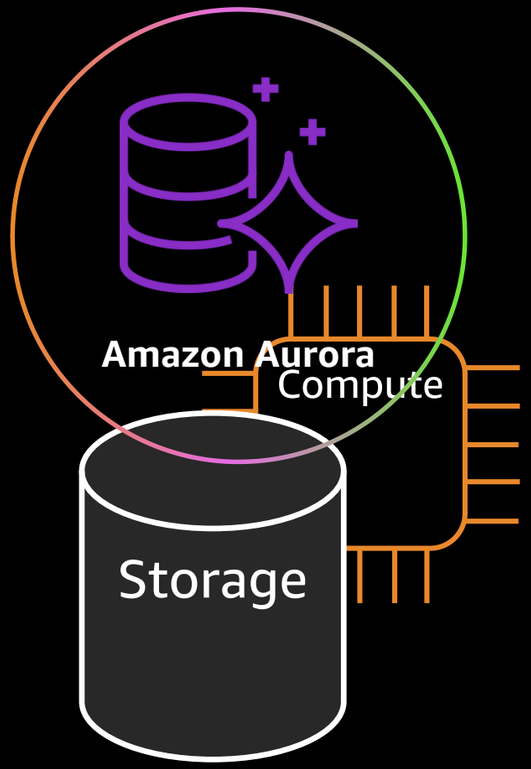
Caching

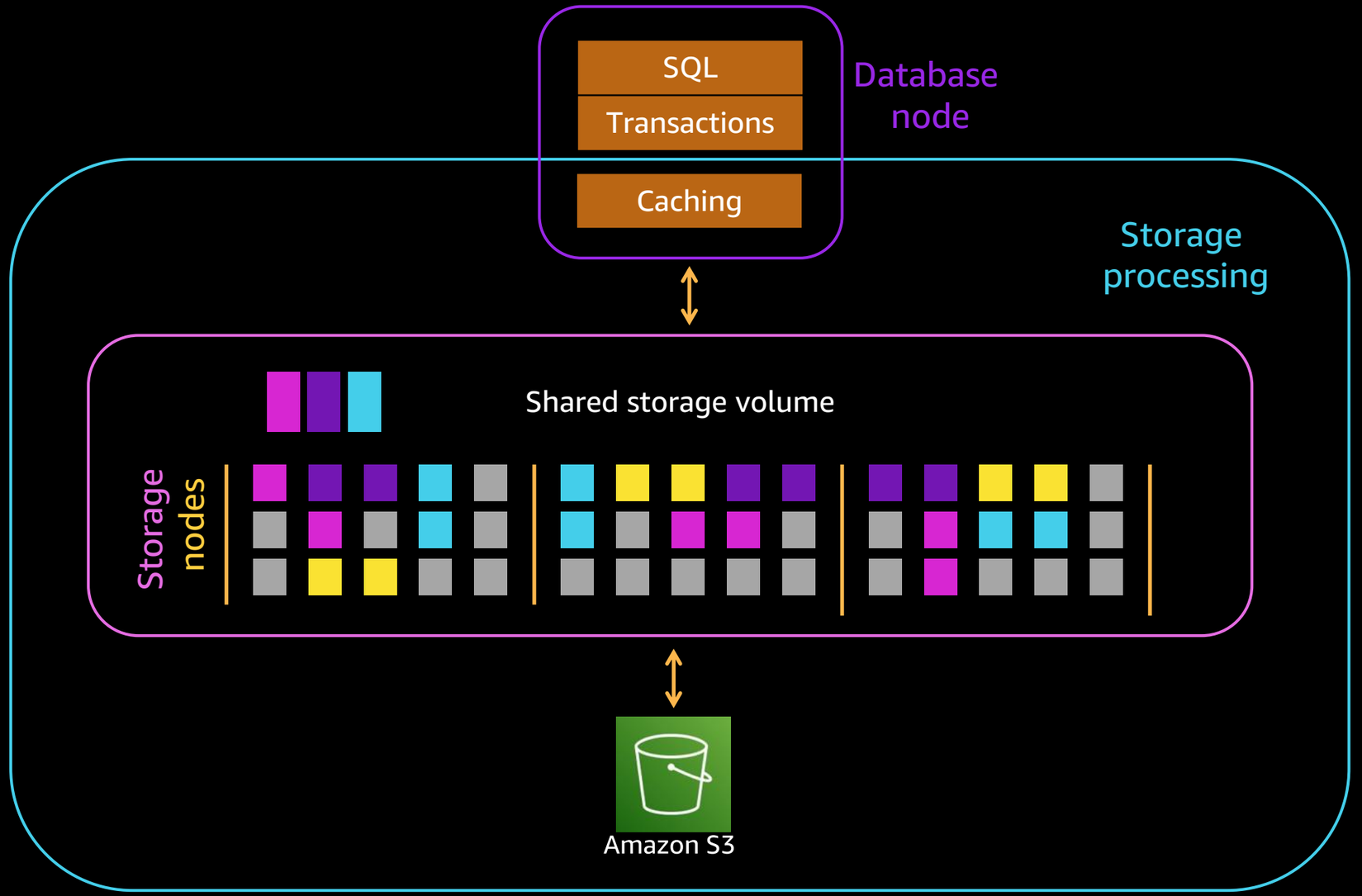
Logging and storage



Multiple layers of processing all on a single engine

Aurora decouples storage and query processing





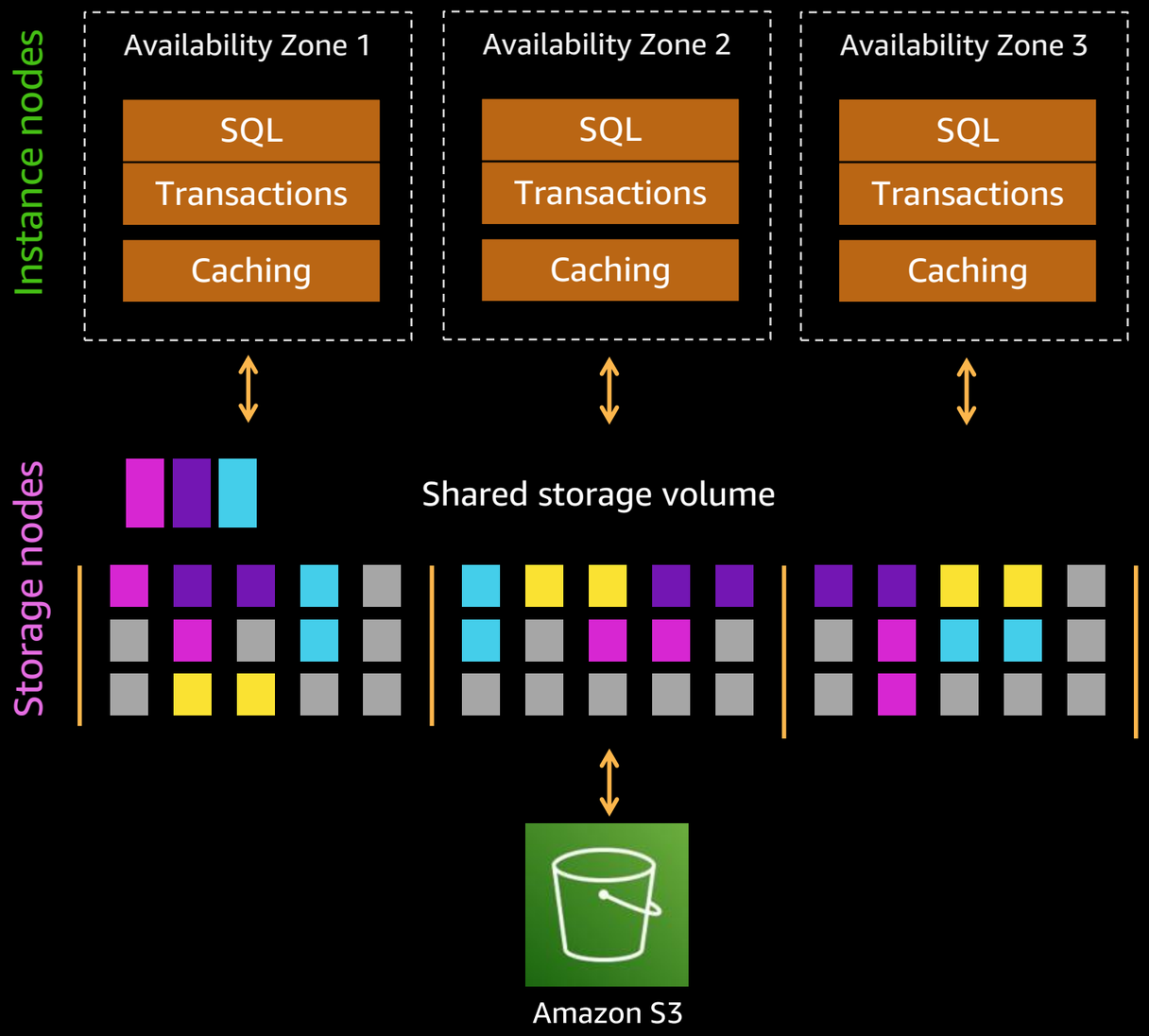
Scale-out, distributed storage processing architecture

Purpose-built log-structured distributed storage system designed for databases

Storage volume is striped across hundreds of storage nodes distributed over 3 different Availability Zones

6 copies of data, 2 copies in each Availability Zone to protect against AZ+1 failures

Data is written in 10 GB “protection groups,” growing automatically when needed



Benefits of the Aurora model



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Aurora distributed storage provides:



Delivered as a **managed** service

- Redo log processing
- Instant crash recovery
- Fault-tolerant and self-healing storage
- Fast database cloning
- Database backtrack
- Database snapshots
- Continuous backups and point-in-time restore
- Storage automatic scaling independent of compute
- Read and write scalability
- Warm cache on database restart
- Low-latency replication

Aurora distributed storage provides:



So the database can process your transactions and SQL queries without impact from storage processing activities

- Redo log processing
- Instant crash recovery
- Fault-tolerant and self-healing storage
- Fast database cloning
- Database backtrack
- Database snapshots
- Continuous backups and point-in-time restore
- Storage automatic scaling independent of compute
- Read and write scalability
- Warm cache on database restart
- Low-latency replication

What’s new?



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Storage



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

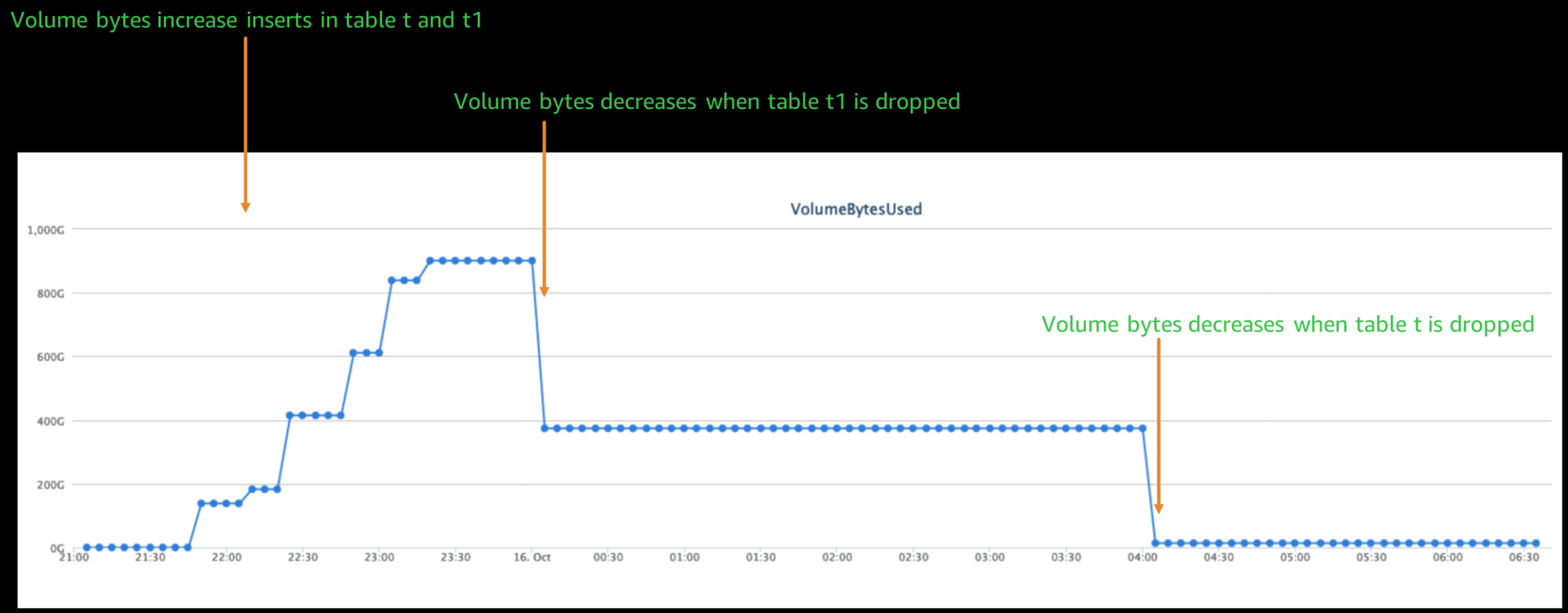
Database storage

Aurora now support database sizes of up to 128 tebibytes (TiB)

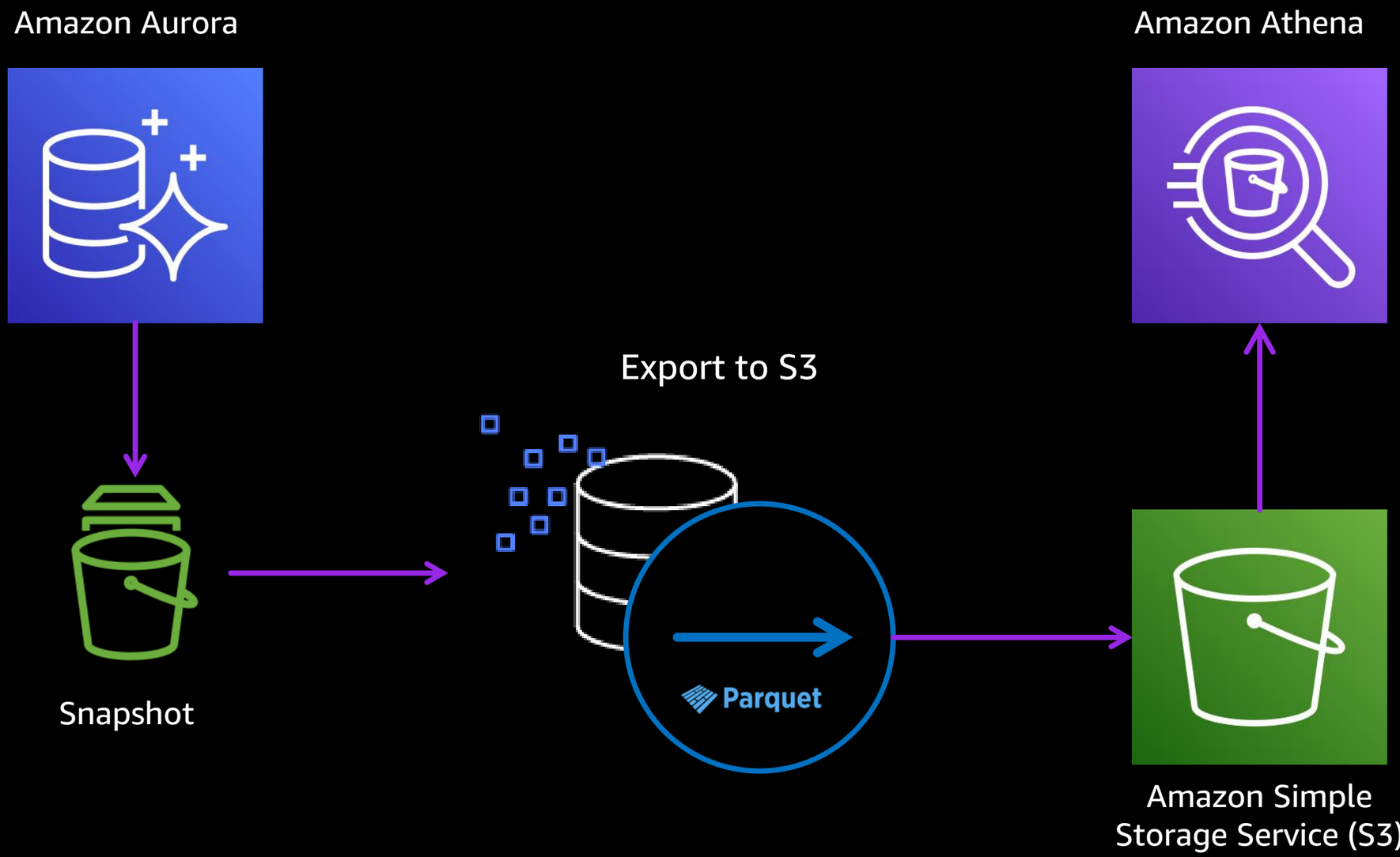


Database storage

Database volume storage now
automatically shrinks as data is deleted



Snapshot export to Amazon S3



AWS Backup support for Aurora

AWS BACKUP AUTOMATES YOUR BACKUPS ACROSS AWS SERVICES

Centralized backup management



Centralized console and consolidated API across AWS services



Meet business and regulatory backup compliance requirements

Policy- and tag-based backup solution



Create backup policies to define backup requirements

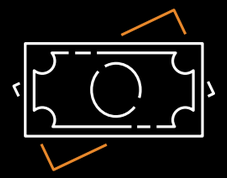


Common way to protect application data in the AWS Cloud and on premises

Automated backup scheduling



Automatically back up AWS resources according to the schedule policies



Simple and cost-effective

Automated retention and lifecycle management



Retain / expire backups or store in low-cost storage according to policies



Backup encryption (independent of source resource)

Compute



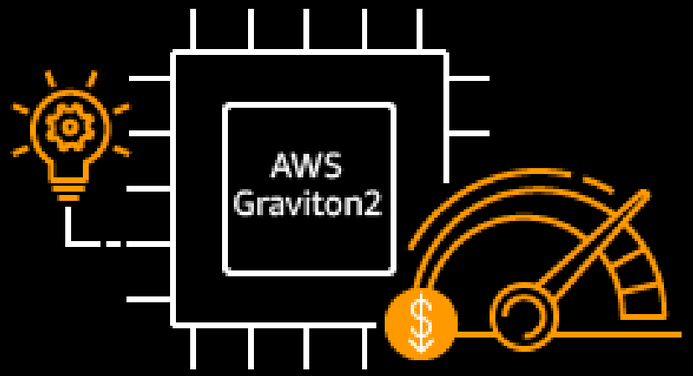
AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Graviton2-powered Aurora (preview)

Coming soon

GRAVITON INSTANCES COMING TO AURORA



AWS Graviton2

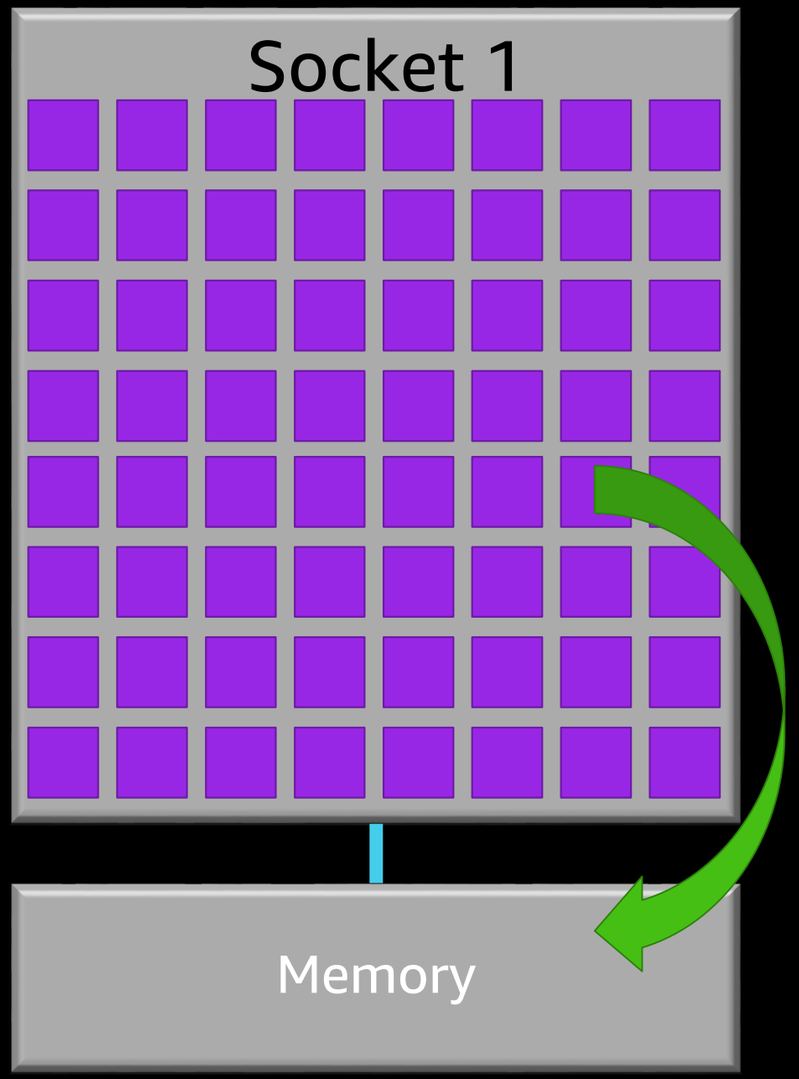
Powerful and efficient, modern applications

- Custom designed by AWS utilizing Arm Neoverse N1 cores
- Graviton2, fully compatible with all existing Aurora features – no porting needed
- Up to 35% better performance

Graviton2 – Uniformity



R6g instance

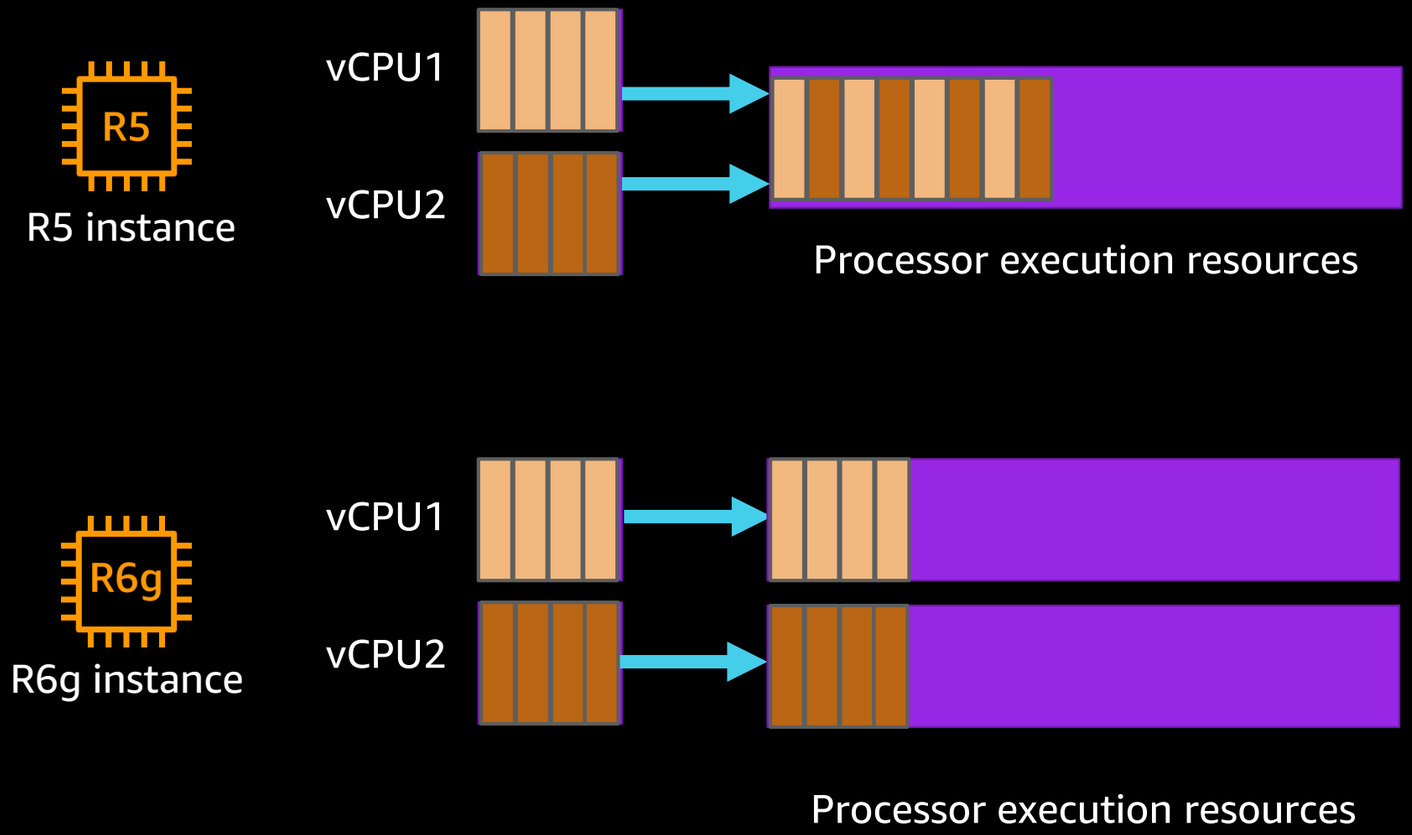


64-core

Single socket

Uniform memory access

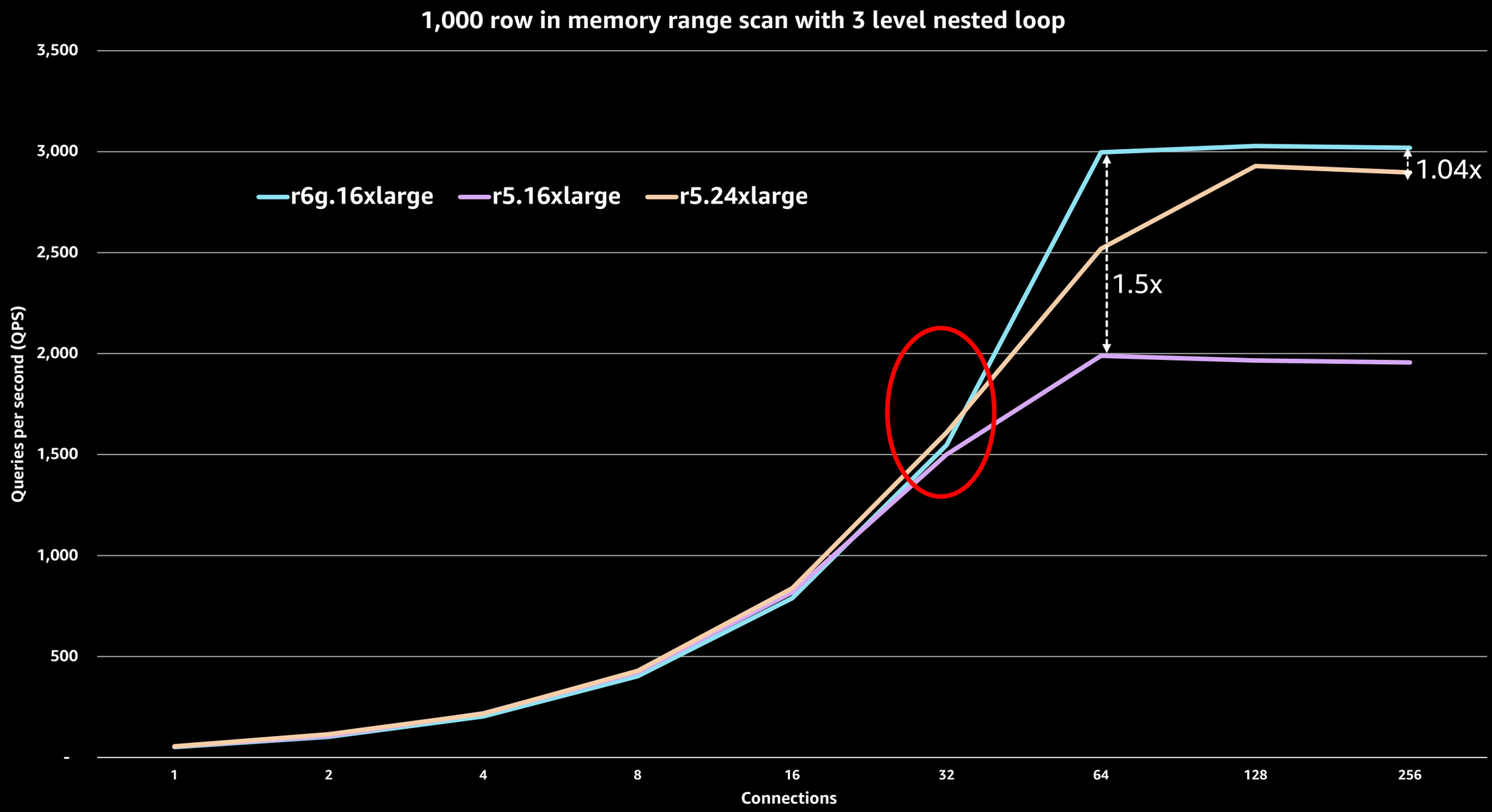
Graviton2 – vCPU



Every vCPU is a physical core

No simultaneous multi-threading (SMT)

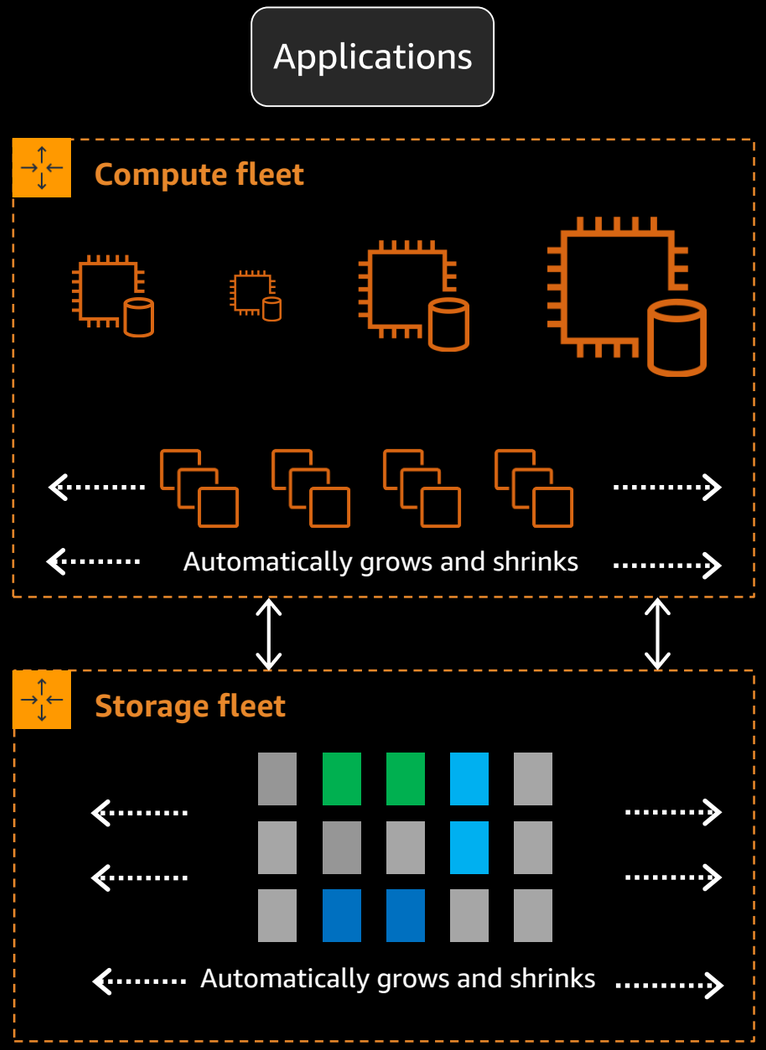
Graviton2


1,000 row in memory range scan with 3 level nested loop
3,500
3,000
2,500
2,000
1,500
1,000
500
-
Queries per second (QPS)
r6g.16xlarge
r5.16xlarge
r5.24xlarge
1.04x
1.5x
1
2
4
8
16
32
64
128
256
Connections

Amazon Aurora Serverless v2 (preview)

NEW

PREVIEW



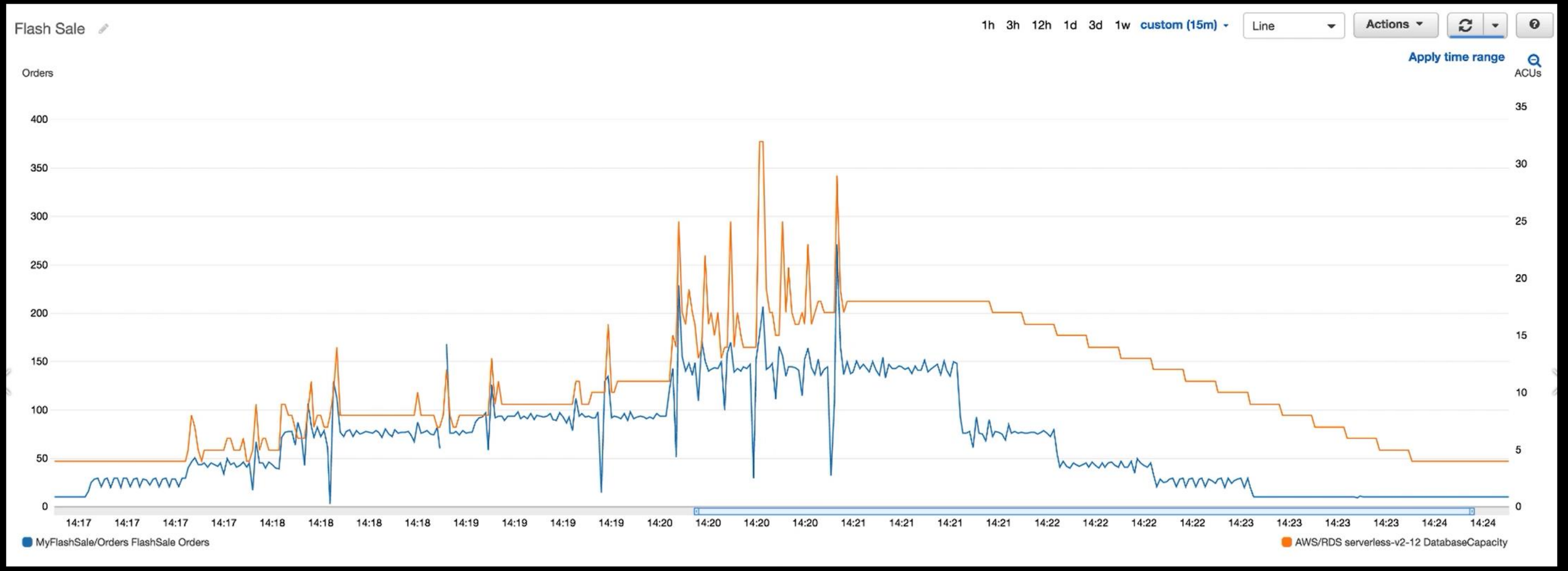
Scale in fine-grained increments to provide just the right amount of database capacity

Scale instantly, from hundreds to hundreds of thousands of transactions, in a fraction of a second

Up to 90% cost savings when compared to provisioning for peak load

Full breadth of Aurora capabilities including parallel query, global database, read replicas, backtrack, and multi-AZ support

Amazon Aurora Serverless v2



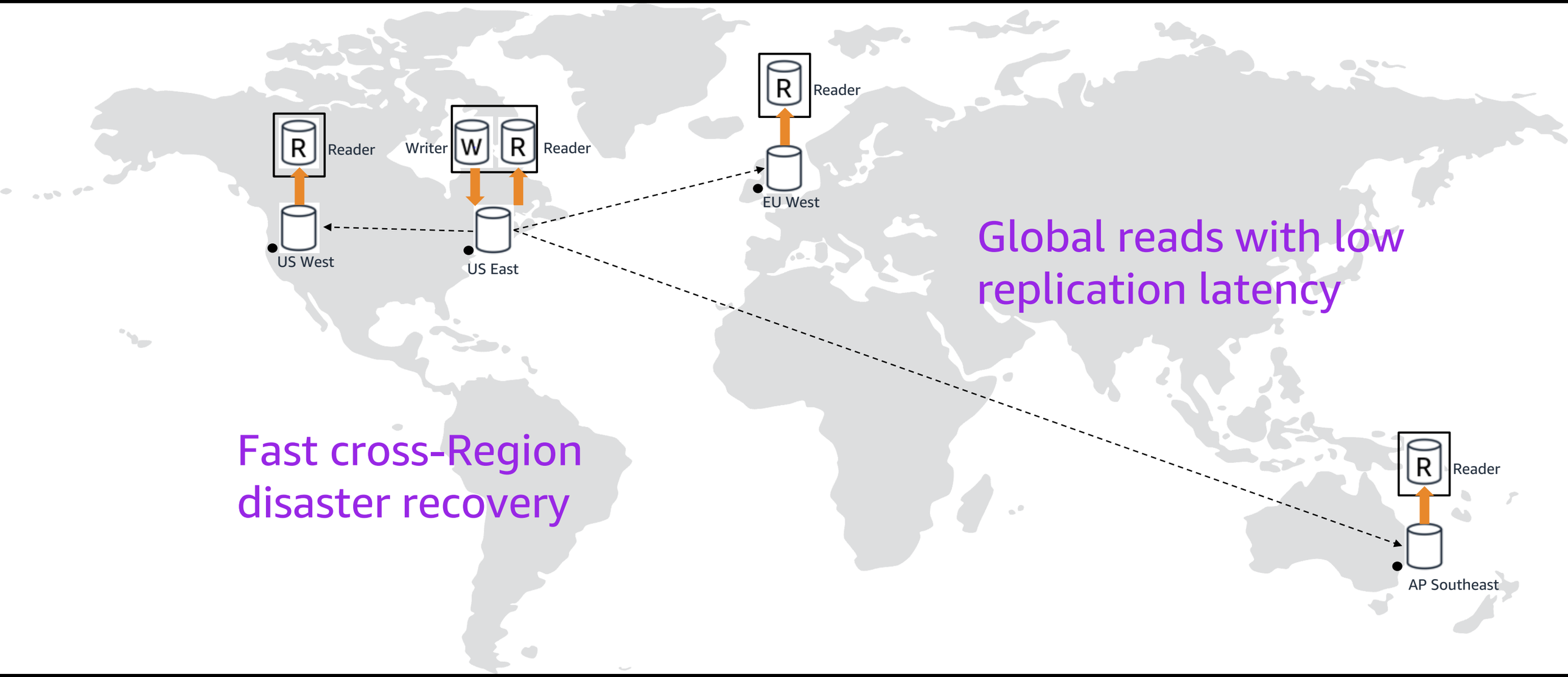
Global Database



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

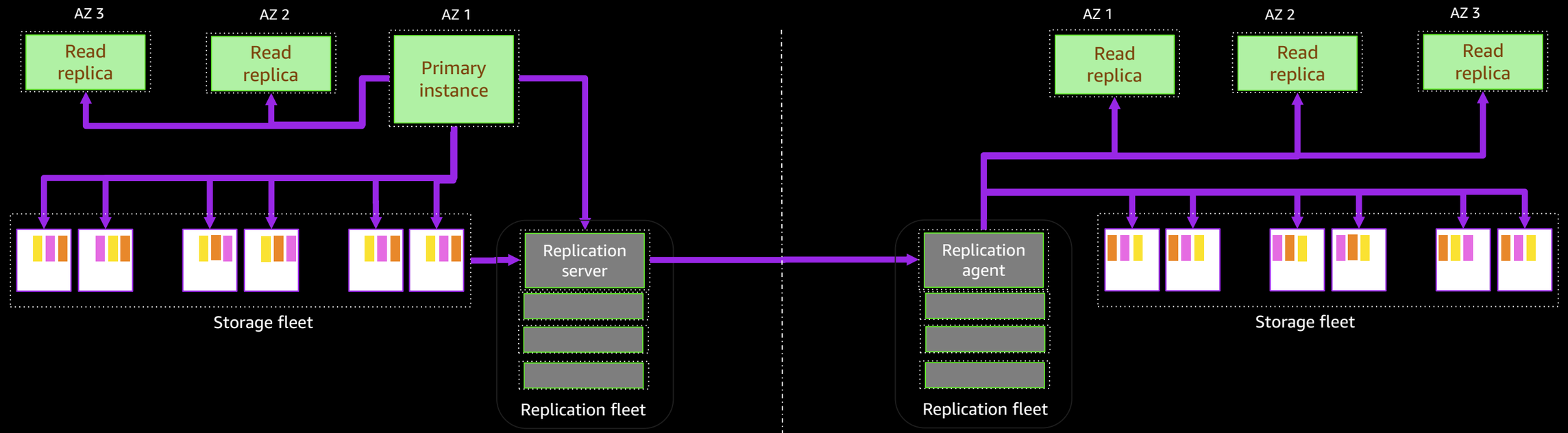
Use cases


R
Reader
US West
Writer
W
R
Reader
US East
R
Reader
EU West
R
Reader
AP Southeast
Global reads with low replication latency
Fast cross-Region disaster recovery

Global reader instances – Up to 90

READ SCALABILITY WITH GLOBAL DATABASES

Aurora Global Database supports up to 90 reader instances spread across 6 Regions without impacting the database writer performance in the primary Region



Aurora Global Database – Managed RPO

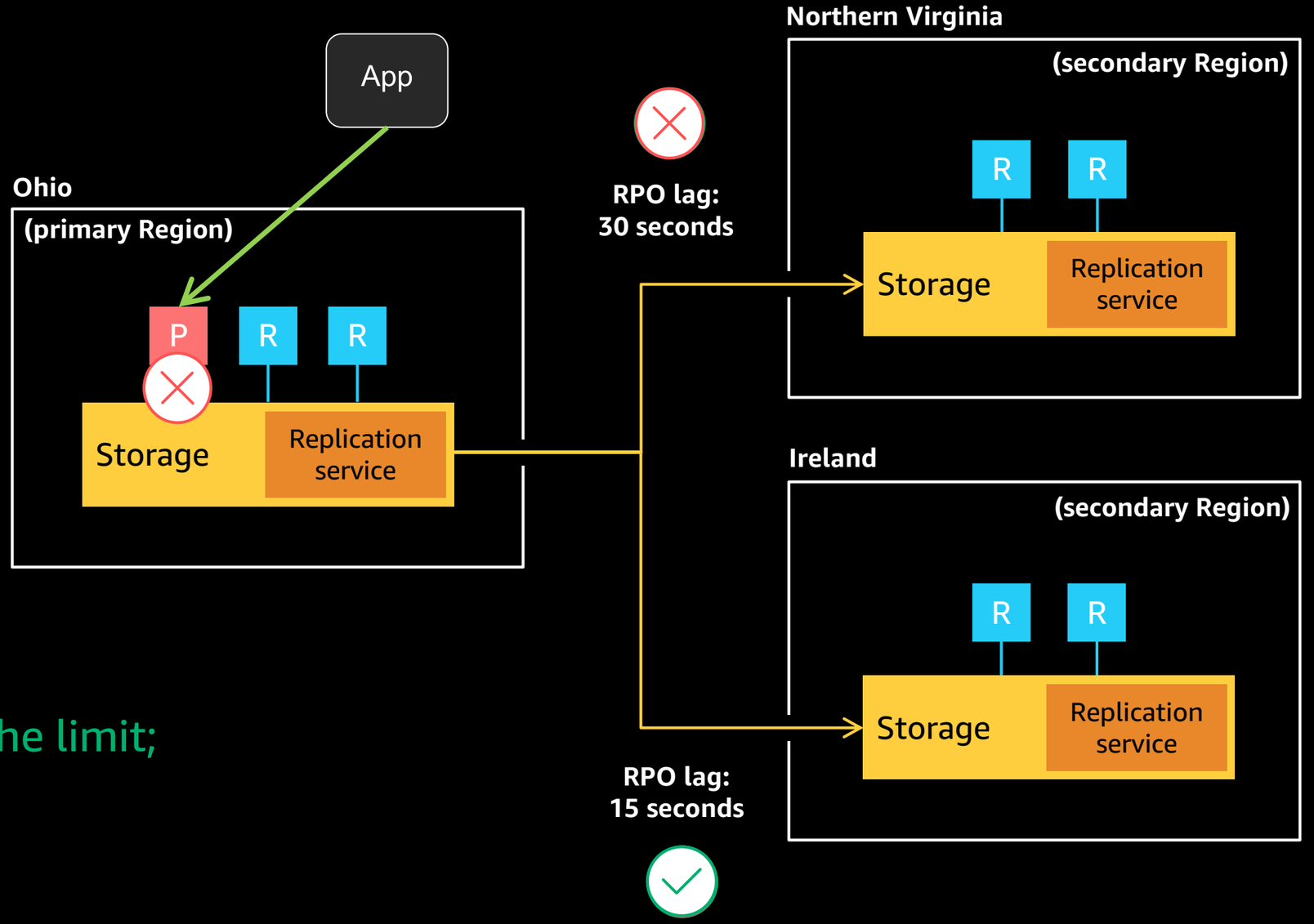
For applications with **critical RPO requirements**

You define the maximum RPO that you allow

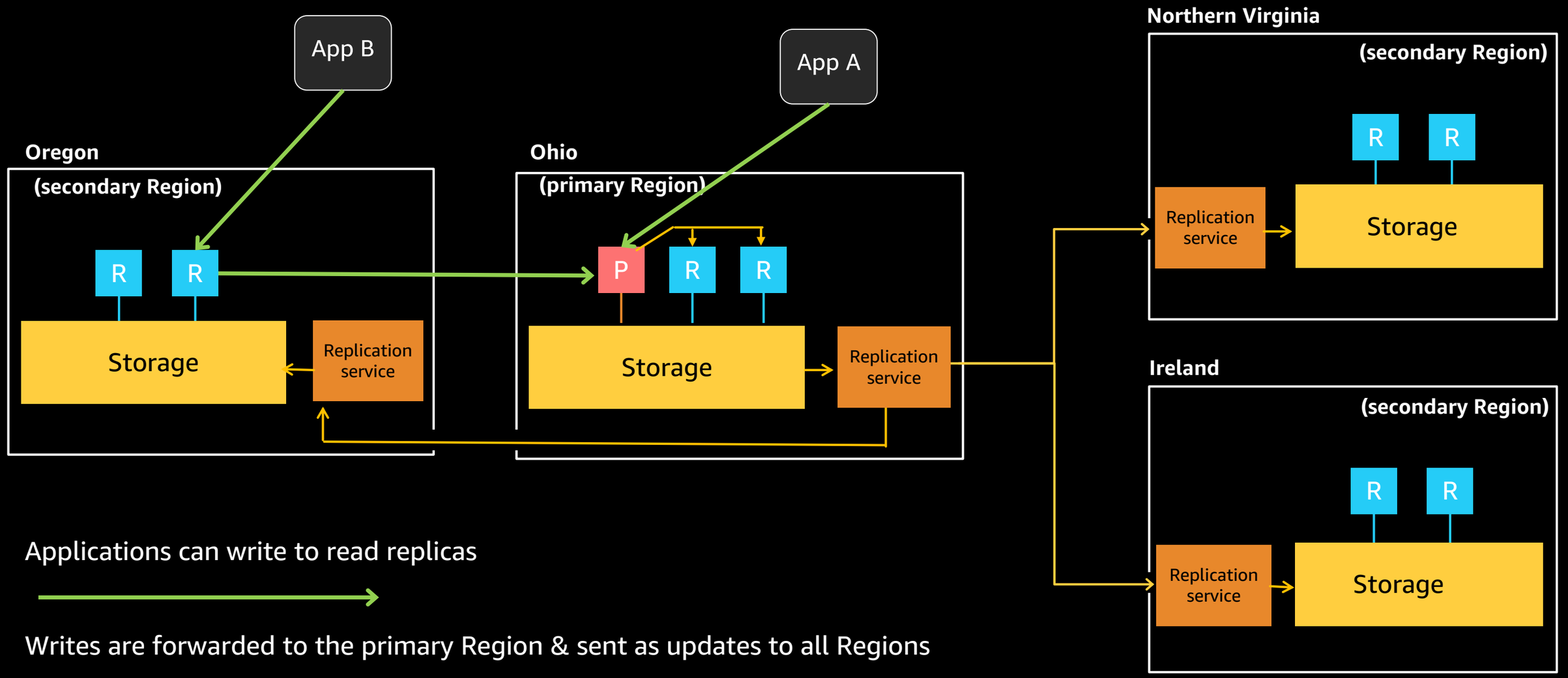
If RPO lag in **all** secondary Regions exceeds the limit, Aurora pauses writes until **at least one** Region catches up

Let's see an example where we set RPO = 20 seconds

→ Lag in Ireland back under the limit; writing has been resumed



Aurora Global Database: Write Forwarding



Applications can write to read replicas

Writes are forwarded to the primary Region & sent as updates to all Regions

Any Region can be promoted to be the primary

Aurora Global Database: Managed Failover

Coming soon

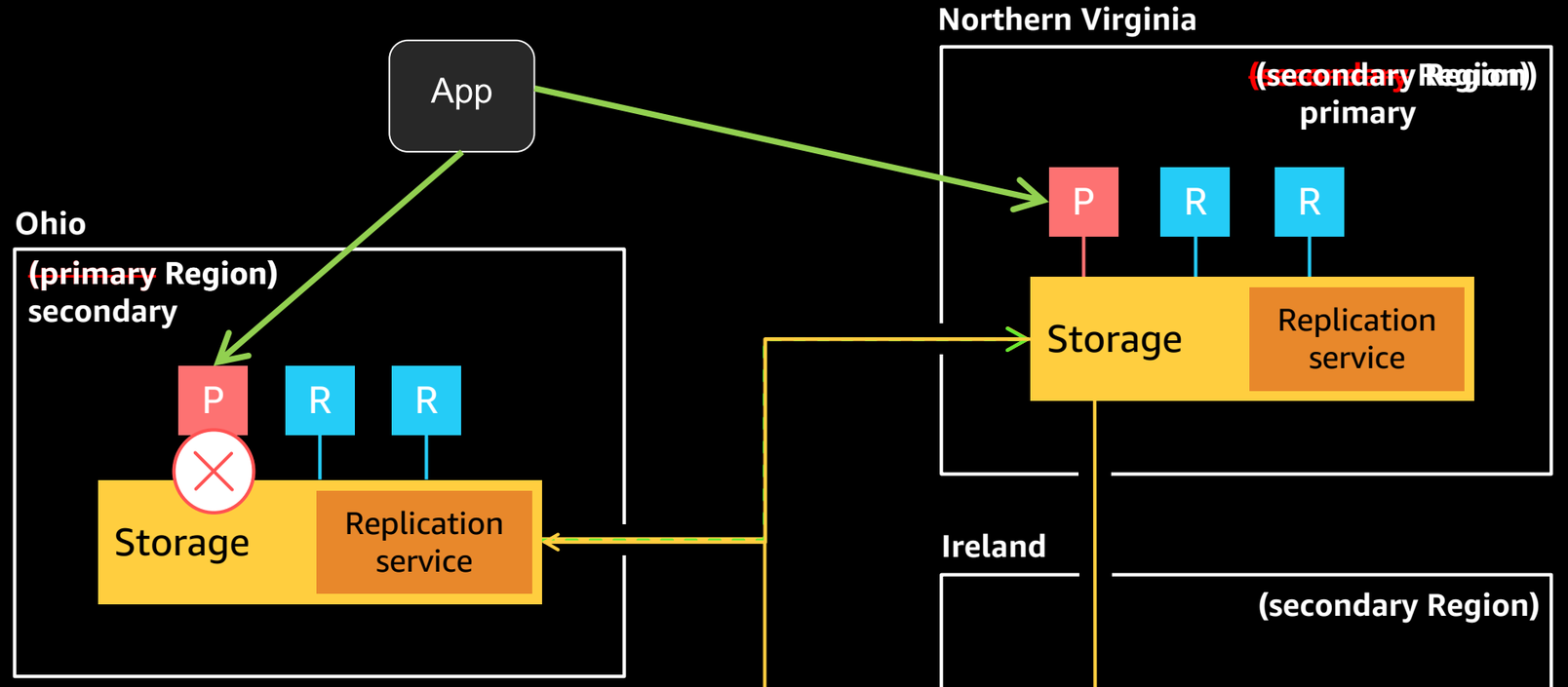
Automated way to promote secondary Region to primary

For testing DR setup or relocating primary Region

Fully maintains the topology

RPO = 0; writes are stopped until new primary catches up

In this example, we'll promote N. Virginia to primary



→ We click to promote N. Virginia
→ Ohio writes are stopped
→ N. Virginia catches up
→ Other Regions attached to N. Virginia
→ Writes now available in N. Virginia
→ Point your app at the new endpoint and go

Versions



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon Aurora versions

MySQL compatibility



- Automated minor version upgrades
- In place upgrades from Aurora MySQL 5.6 to 5.7 (coming soon)
- Aurora MySQL 8.0 (coming soon)

PostgreSQL compatibility



- Automated minor version upgrades
- In-place upgrade
  - 9.6.9 to 10.11
  - 10.7 to 11.7
  - 11 to 12 (coming soon)
- PostgreSQL 12 (coming soon)

New for PostgreSQL



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Introducing Babelfish for Aurora PostgreSQL

NEW

PREVIEW

Run SQL Server applications on PostgreSQL with little to no code changes

Keep existing queries



Translation layer enables Aurora PostgreSQL to understand Microsoft SQL Server's proprietary T-SQL

Accelerate migrations



Lower risk and complete migrations faster, saving you months to years of work

Freedom to innovate



Run T-SQL code side-by-side with new open-source functionality and continue developing with familiar tools

Babelfish for PostgreSQL (Open-Source Project)

NEW

PREVIEW

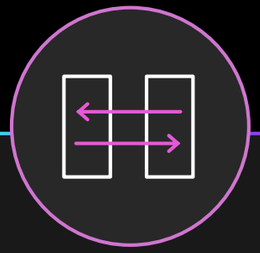
Coming in 2021, an open-source project for Babelfish source code

Customize and add new features



Contribute to help steer the direction of Babelfish

Apache 2.0 license



Use it for any purpose, innovate and distribute your modifications with confidence

Available on GitHub in 2021



Is community-driven and provides transparency into the feature roadmap

Babelfish for PostgreSQL design tenets

GUIDING PRINCIPLES



No compromises on correctness

Database calls either work or return an error



Wire protocol compatibility

Applications work without changing database drivers



Interoperability

Use PostgreSQL functionality from T-SQL and
T-SQL functionality from PostgreSQL code

Related sessions

DAT301

Deep dive on Amazon Aurora with PostgreSQL compatibility



DAT302

Deep dive on Amazon Aurora with MySQL compatibility



DAT401

Amazon Aurora storage demystified



Thank you!

Tony Petrossian

@tonypetrossian



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.