AWS re:Invent

DAT367-R2

Running SQL Server on Amazon RDS and migrating to Aurora MySQL

Richard Waymire
Outbound Principal Architect
Amazon Web Services

Mahesh Kansara
Database Engineer
Amazon Web Services

AWS re:Invent

aws

Agenda

Run the AWS Schema Conversion Tool to convert an SQL Server schema to Amazon Aurora PostgreSQL

Use AWS Database Migration Service (AWS DMS) to migrate your SQL Server database to Aurora PostgreSQL

Perform data updates on SQL Server and monitor them replicating

Related breakouts

DAT367-R, R1, R2 - Running SQL Server on Amazon RDS and migrating to Aurora MySQL

Start your AWS CloudFormation template restore

- Your first step in this lab is to login to Event Engine using the credential shared with you

- Directions for the lab: https://tinyurl.com/sv8pfet

Let’s talk

Join us in the *[track name]* Networking Lounge at *[location]* on *[day]*, December *[date],* from *[start time]* to *[end time]*.

SQL Server on AWS

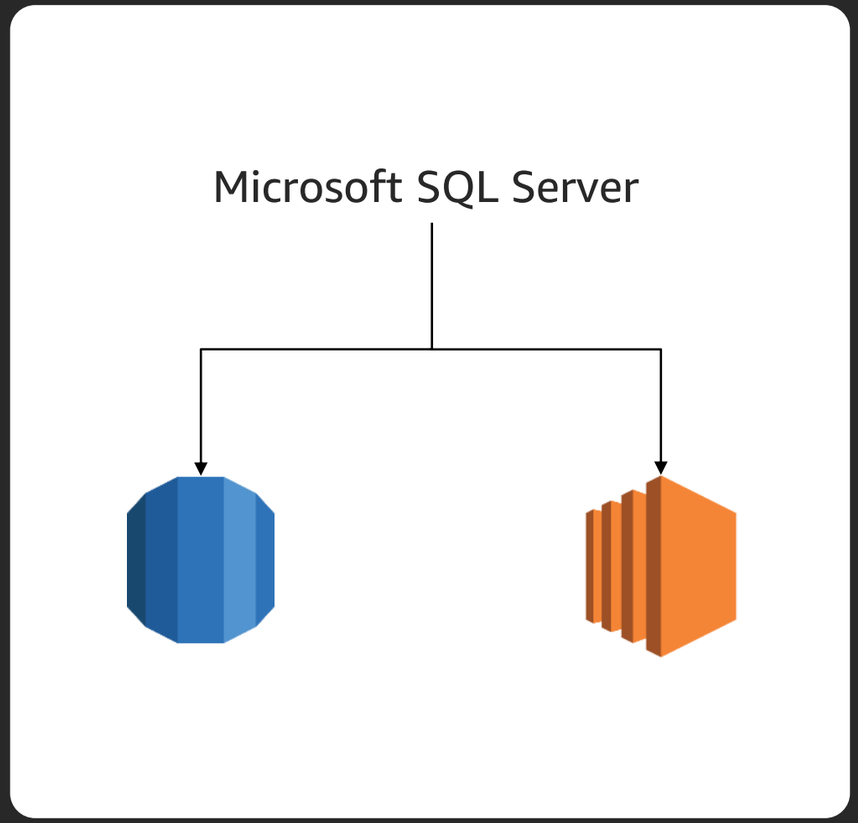
AWS re:Invent

aws

Choose the best service for your needs

Amazon RDS SQL Server

- Managed physical infrastructure
- Managed DB install and backups
- Managed OS and patching
- Managed high availability and scaling



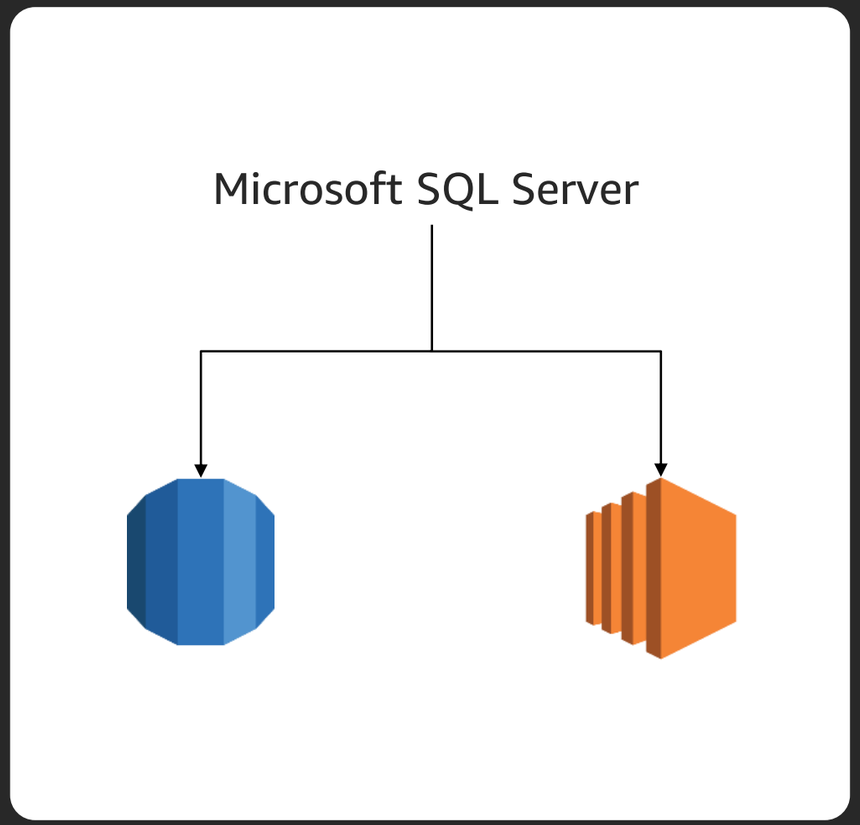
SQL Server on Amazon EC2

- Managed physical infrastructure
- Managed OS installation
- Managed scaling
- OS-level control

Choose the best service for your needs

Amazon RDS SQL Server

Your responsibility
- App optimization and tuning
- Deployment orchestration



SQL Server on Amazon EC2

Your responsibility
- App optimization and tuning
- Deployment orchestration
- Monitoring and recovery
- High availability
- Backups
- DB & OS patching

SQL Server features at a glance

	Amazon RDS	Amazon EC2
Versions supported:	2012, 2014, 2016, 2017	All
Editions supported:	Express, Web, Standard, Enterprise**	All
High availability:	AWS-managed	Self-managed; AlwaysOn, Mirror, Log Ship
Encryption:	Encrypted storage using AWS KMS (all editions); TDE support	
Authentication:	Windows & SQL authentication	
Backups:	Managed automated backups	Maintenance plans & third-party tools
Maintenance:	Automatic software patching	Self-managed

SQL Server EC2 vs. Amazon RDS: Which should I use?

	EC2	RDS
License included	✓	✓
BYOL	✓	
Full control over the instance	✓	
Automated backups		✓
Self-managed AlwaysOn availability groups	✓	
AWS-managed Multi-AZ deployment		✓

Migrating using AWS DMS and AWS SCT

Flexible, powerful migration tooling

Schema conversion

Data replication

Data warehouse migration

Breadth of data sources

Relational

SQL Server, Oracle, SAP ASE, Db2 LUW, SQL Azure

Non-relational

Cassandra

Supported in SCT for planning/conversion and DMS for migration

Relational

Amazon Aurora, PostgreSQL, MariaDB, MySQL

Non-relational

MongoDB

Analytics

Amazon S3, AWS Snowball

Supported as a source in DMS to replicate data from

On-premises data warehouses

Oracle
Teradata
Vertica
Netezza
Greenplum
SQL Server

Supported in SCT for planning/conversion and migration

Schema Conversion Tool (SCT)

Database Migration Service (DMS)

Schema Conversion Tool (SCT)

Choice of data targets

Relational

Amazon Aurora, PostgreSQL, MariaDB, MySQL

Non-relational

Amazon DynamoDB

Supported by SCT and DMS

Relational

SQL Server, Oracle, SAP ASE

Non-relational

Amazon DocumentDB (with MongoDB compatibility)

Analytics

Amazon Elasticsearch Service, Amazon Kinesis Data Streams, Amazon S3

Supported as a target in DMS to replicate data to

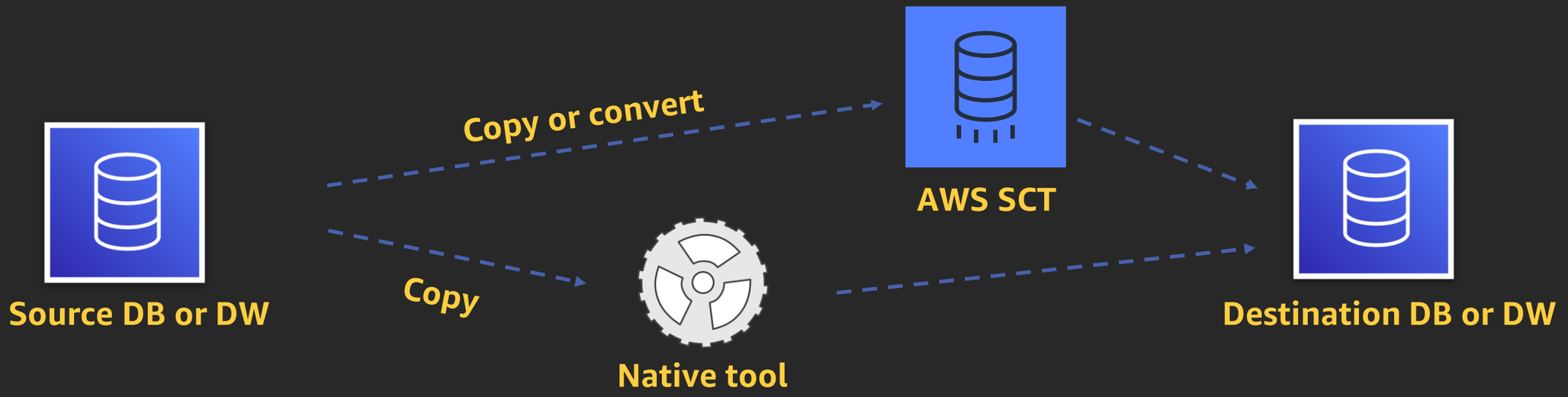
Cloud data warehouse

Amazon Redshift

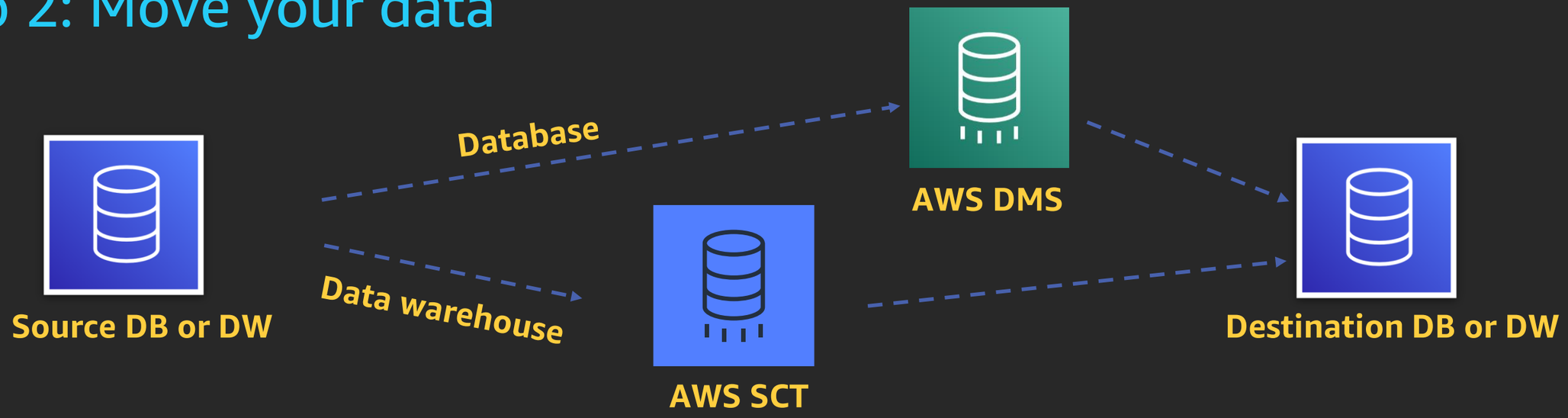
- **Supported as a target in DMS to replicate data to**
- **Supported as a target in SCT to migrate data from on-premises data warehouses**

Database migration process

Step 1: Convert or copy your schema



Step 2: Move your data



>200K databases migrated with AWS DMS

AWS Schema Conversion Tool

AWS Schema Conversion Tool helps automate database schema and code conversion tasks when migrating from source to target database engines

Features

Create assessment reports for homogeneous/heterogeneous migrations

Convert database schema

Convert data warehouse schema

Convert embedded application code

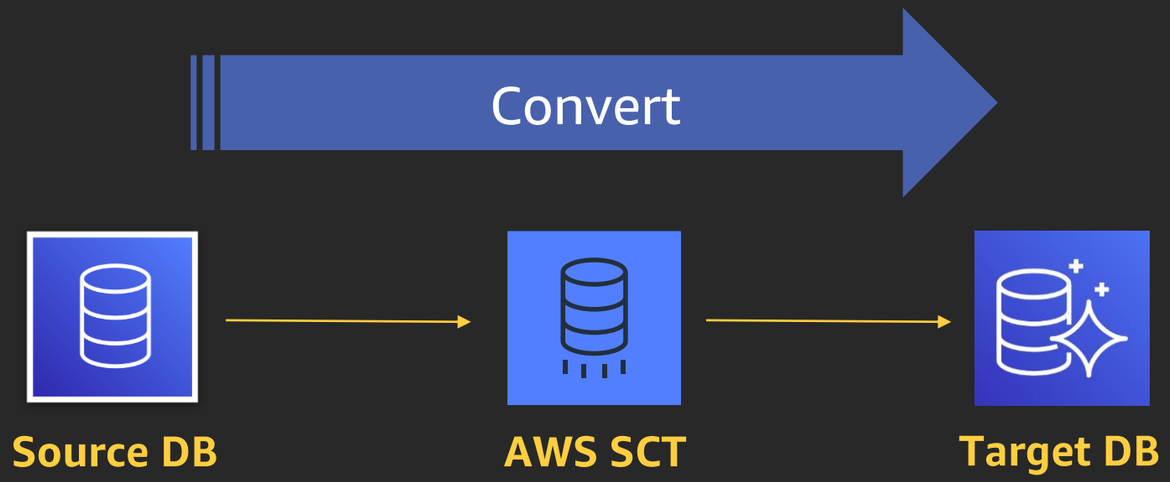
Code browser that highlights places where manual edits are required

Secure connections to your databases with SSL

Service substitutions/ETL modernization to AWS Glue

Migrate data to data warehouses using AWS SCT data extractors

Optimize schemas in Amazon Redshift



Supported source and targets

Relational

NoSQL

Data lake

Data warehouse*

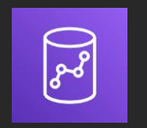
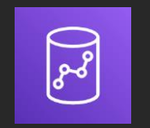
Sources


Amazon Aurora

MongoDB

Cassandra

Amazon S3

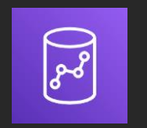
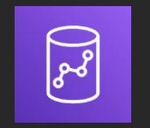
AWS Snowball

Oracle

SQL Server

Netezza



SAP ASE

Db2 LUW

SQL Azure

Greenplum

Teradata

Vertica

Targets


Amazon Aurora

Amazon DynamoDB

Amazon Elasticsearch Service

Amazon Kinesis Data Streams

Amazon Redshift



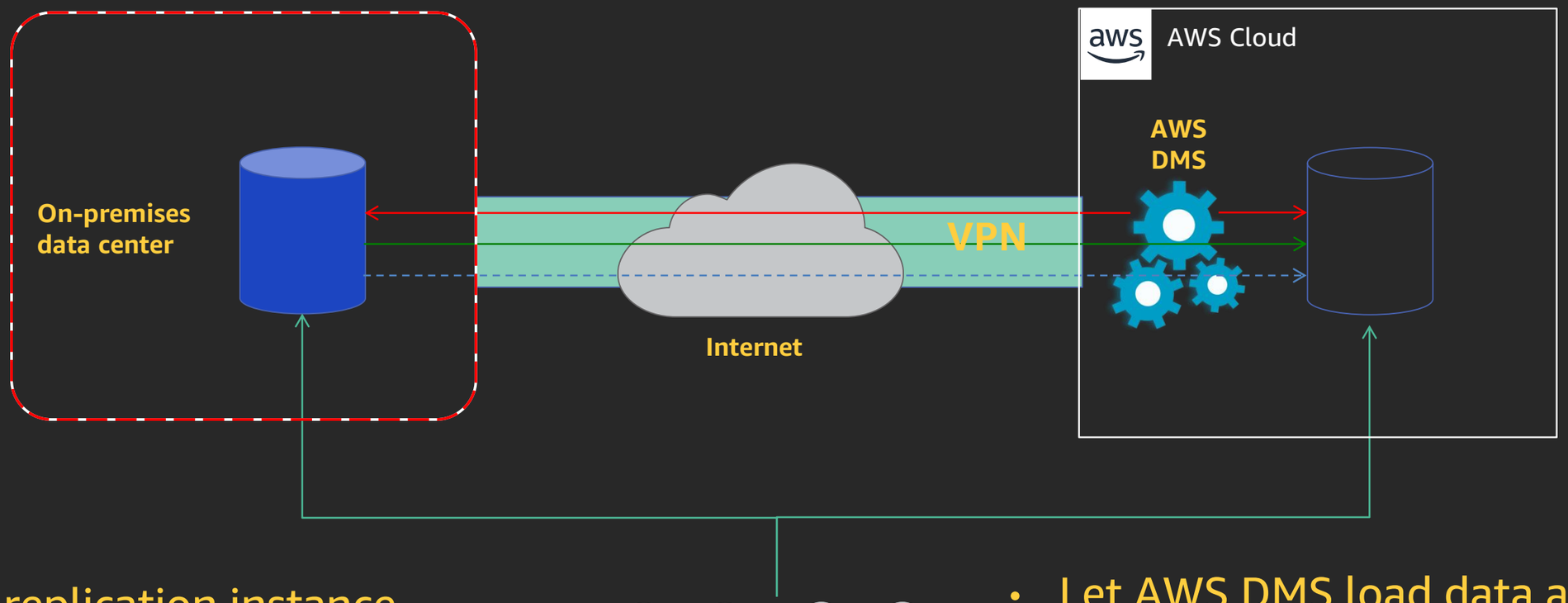
SAP ASE

Amazon DocumentDB (with MongoDB compatibility)

Amazon S3

*** Supported via AWS SCT data extractors**

The data migration process



- Start a replication instance
- Connect to source and target databases
- Select tables, schemas, or databases

- Let AWS DMS load data and keep them in sync
- Switch applications over to the target once in sync at your convenience

Amazon Aurora PostgreSQL

Amazon Aurora

A relational database reimagined for the cloud

- ☑ **Speed** and **availability** of high-end commercial databases
- ☑ **Simplicity** and **cost effectiveness** of open-source databases
- ☑ Drop-in **compatibility** with MySQL and PostgreSQL
- ☑ Simple **pay-as-you-go** pricing

Delivered as a **managed** service

Amazon Aurora 101

MySQL- and PostgreSQL-compatible relational database built for the cloud

Performance and availability of commercial-grade databases at 1/10th the cost

---

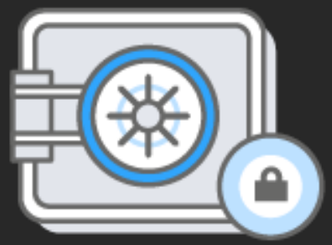
Performance & scalability

5x throughput of standard MySQL and 3x of standard PostgreSQL; scale-out up to 15 read replicas

Availability & durability

Fault-tolerant, self-healing storage; six copies of data across three AZs; continuous backup to Amazon S3

Highly secure



Network isolation, encryption at rest/transit

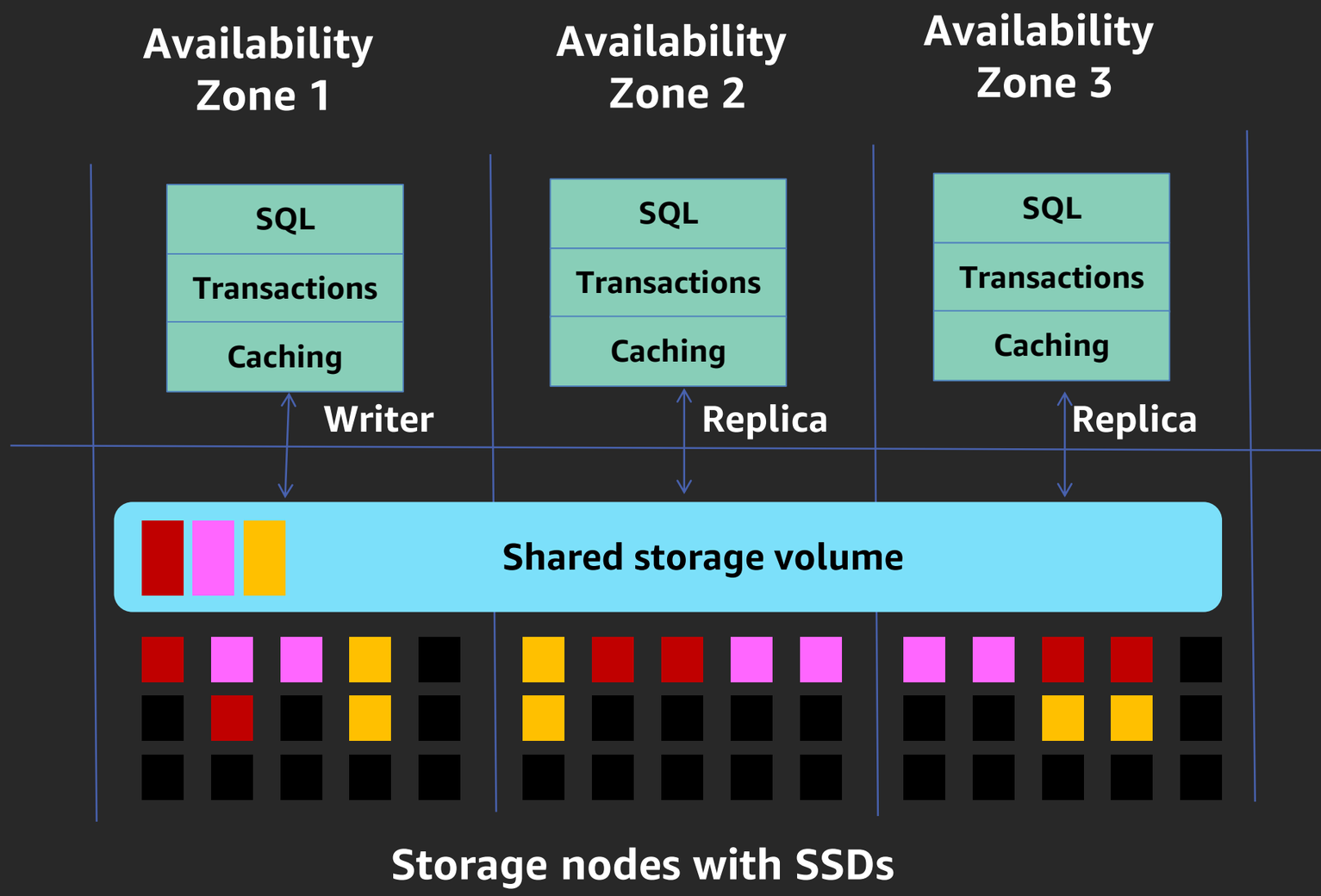
Fully managed

Managed by RDS: No hardware provisioning, software patching, setup, configuration or backups

What is Amazon Aurora?

Leverages a scale-out, distributed architecture

- Purpose-built, log-structured, distributed storage system designed for databases
- Storage volume is striped across hundreds of storage nodes distributed over three different Availability Zones
- Six copies of data, two copies in each Availability Zone to protect against AZ+1 failures
- Plan to apply same principles to other layers of the stack

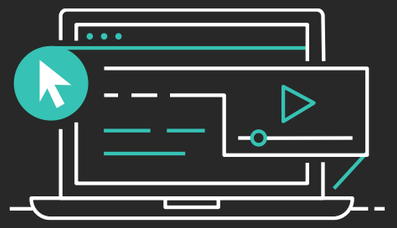


Some Key Aurora PostgreSQL features

- Serverless
- Query plan management
- Cluster cache management
- Global physical replication (in preview)
- Logical cross-region replication
- Integration with AWS Identity and Access Management (IAM), Amazon Simple Storage Service (Amazon S3), AWS Lambda, and Amazon CloudWatch
- Fast database cloning

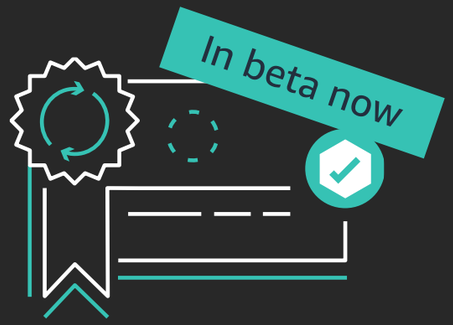
Learn databases with AWS Training and Certification

Resources created by the experts at AWS to help you build and validate database skills



25+ free digital training courses cover topics and services related to databases, including:

- Amazon Aurora
- Amazon Neptune
- Amazon DocumentDB
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Redshift
- Amazon RDS



Validate expertise with the new **AWS Certified Database - Specialty** beta exam

Visit aws.training

aws training and certification

Thank you!

Richard Waymire

waymire@amazon.com

Mahesh Kansara

kansara@amazon.com

Please complete the session survey in the mobile app.