



INDIA | MAY 25, 2023

S D B 0 0 7

# Database modernization from legacy commercial to opensource database

Nethravathi Muddarajaiah  
Senior Database Specialist Solutions Architect  
AWS India



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Agenda

- Database migration patterns
- AWS purpose-built managed databases
- Migrations tools and services

**A strong vision, with the right preparation and training sets you up for success.**



**Swami Sivasubramanian**  
VP of Database, Analytics, and Machine Learning  
AWS

# Samsung

Samsung migrates 1.1 billion users to Amazon Aurora with AWS Database Migration Service (AWS DMS)

## Challenge

IT giant Samsung Electronics needed a more flexible, microservices-driven solution to replace its monolithic legacy internet data center, which was proving costly and unable to accommodate growing traffic.

## Solution

The company migrated its **1.1 billion** Samsung Account users to Amazon Aurora with minimal service disruption using AWS Database Migration Service, drastically reducing database costs and increasing scalability.

## Results

- Underwent a global migration of a mission-critical workload in 18 months
- Enabled 60 ms latency or less 90% of the time
- **Reduced monthly database costs by 44%**



# Experian

Experian sees 100% uptime with Amazon DynamoDB,  
Amazon Aurora

## Challenge

Poised to reinvent the way it presented credit and financial data to consumers, Experian realized its on-premises monolithic database architecture didn't provide the functionality the company needed to grow.

## Solution

The company migrated from a monolithic to a microservices database with Amazon DynamoDB and upgraded CSID's MySQL system to Amazon Aurora to expedite empowering financial products.

## Results

- Enables 100% availability for CSID
- Brings server configuration and deployment down from 60–90 days to a matter of hours
- Accommodates 50–75% data layer growth each year

Migrations →  
Modernize →



Amazon  
DynamoDB



Amazon  
Aurora



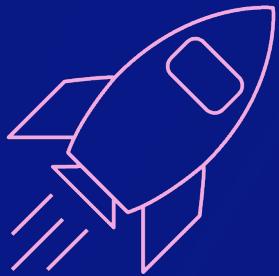
# Database migration patterns

# Key attributes of modern applications



Designed for:

**Innovation  
&  
agility**



Without limits on:

**Performance  
&  
scalability**



Is:

**Highly available,  
easily managed,  
&  
cost-effective**

# Can I migrate ?

**Why should I modernize?**

How do I get started with modernization?

**What tooling do you have to help with modernization?**

**How do I modernize my workloads?**

Which applications should I containerize and which should I move to serverless?

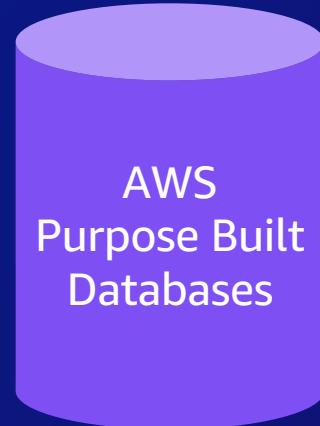
**How can I lower my application and database TCO?**

**How can I dynamically scale my application?**

How can I reduce the dependency on licensing?

**What workloads should I modernize?**

# Why customers are migrating?



Expensive



Restrictive  
licensing  
terms



Reduce  
cost



Increase  
agility



Innovate  
faster

# Database migration patterns

- **Rehost (Lift and Shift)**
  - Homogeneous database migration (like Oracle, SQL Server)
  - Quickest way to onboard workloads
- **Re-platform (Move to Managed)**
  - Auto provisioning, backups, recovery, patching
  - RDS Commercial Engines (Oracle, SQL Server)
- **Re-factor (Modernize)**
  - Break free from legacy databases
  - Reduced licensing costs
  - Enjoy the friendliness of open source solutions
  - Pick the right tool for the right job: choose from 15 purpose-built databases

# AWS Purpose-built managed databases

# Purpose-built databases use cases and AWS services

	Relational	Key-value	Document	In-memory	Graph	Time-series	Ledger	Wide column		
Data models	Referential integrity, ACID transactions, schema-on-write	High throughput, low-latency reads and writes, endless scale	Store documents and quickly access querying on any attribute	Query by key with microsecond latency	Quickly and easily create and navigate relationships between data	Collect, store, and process data sequenced by time	Complete, immutable, and verifiable history of all changes to application data	Scalable, highly available, and managed Apache Cassandra-compatible service		
Common use cases	Lift and shift, ERP, CRM, finance	Real-time bidding, shopping cart, social, product catalog, customer preferences	Content management, personalization, mobile	Leaderboards, real-time analytics, caching	Fraud detection, social networking, recommendation engine	IoT applications, event tracking	Systems of record, supply chain, healthcare, registrations, financial	Build low-latency applications, leverage open source, migrate Cassandra to the cloud		
	 Amazon Aurora	 Amazon RDS	 Amazon DynamoDB	 Amazon DocumentDB	 Amazon ElastiCache	 Amazon MemoryDB for Redis	 Amazon Neptune	 Amazon Timestream	 Amazon QLDB	 Amazon Keyspaces (for Apache Cassandra)

# Amazon RDS

MANAGED RELATIONAL DATABASE SERVICE WITH A CHOICE OF SIX POPULAR DATABASE ENGINES



## Easy to administer



Easily deploy and maintain hardware, OS and DB software; built-in monitoring

## Secure & compliant



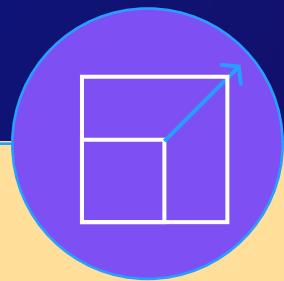
Data encryption at rest and in transit; industry compliance and assurance programs

## Available & durable



Automatic Multi-AZ data replication; automated backup, snapshots, failover

## Performant & scalable



Scale compute and storage with a few clicks; minimal downtime for your application

# Amazon Aurora decouples storage and query processing

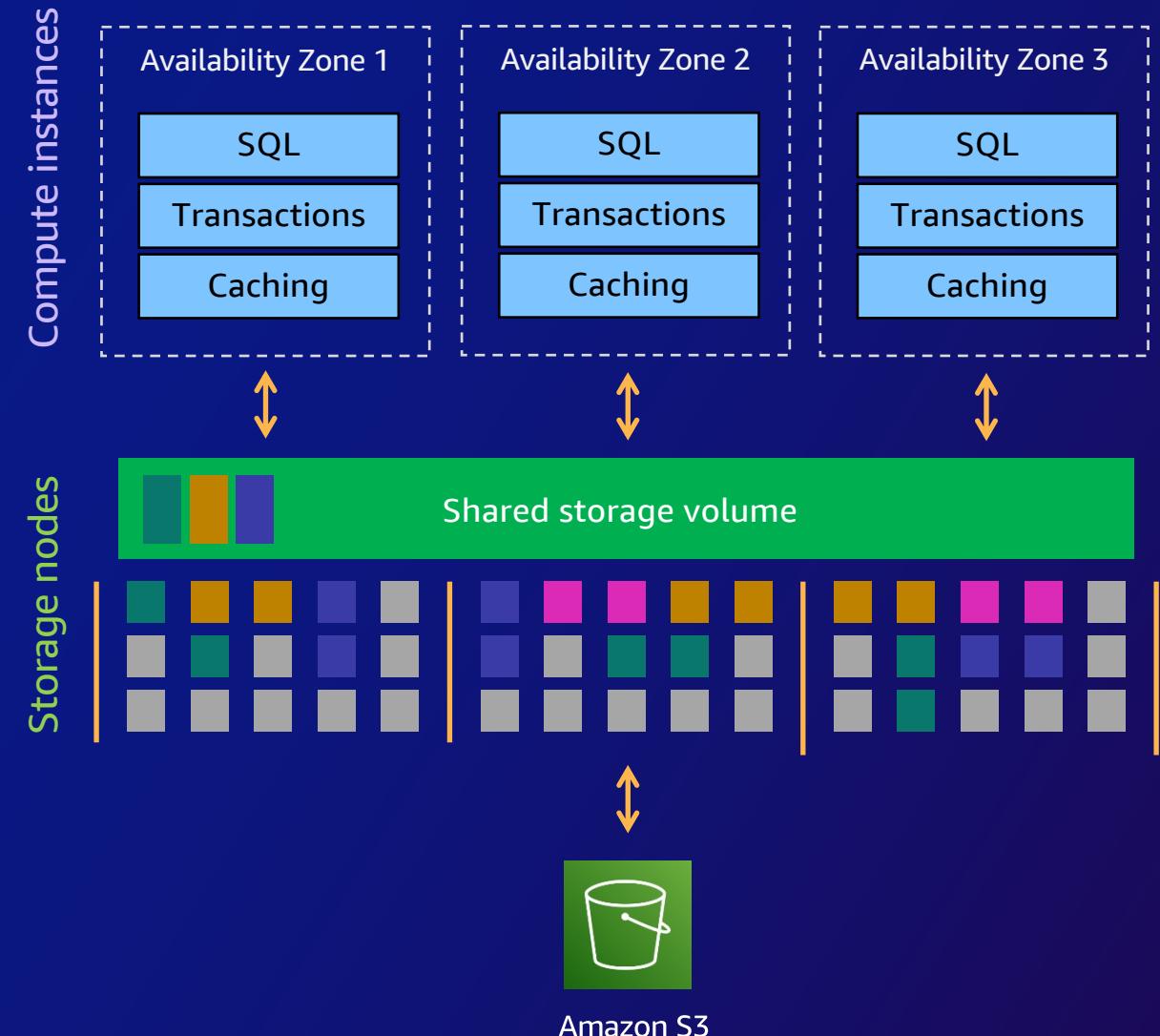
Wire compatible with PostgreSQL and MySQL

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

Highly available, durable, and fault-tolerant custom SSD storage layer: six-way replicated across three Availability Zones

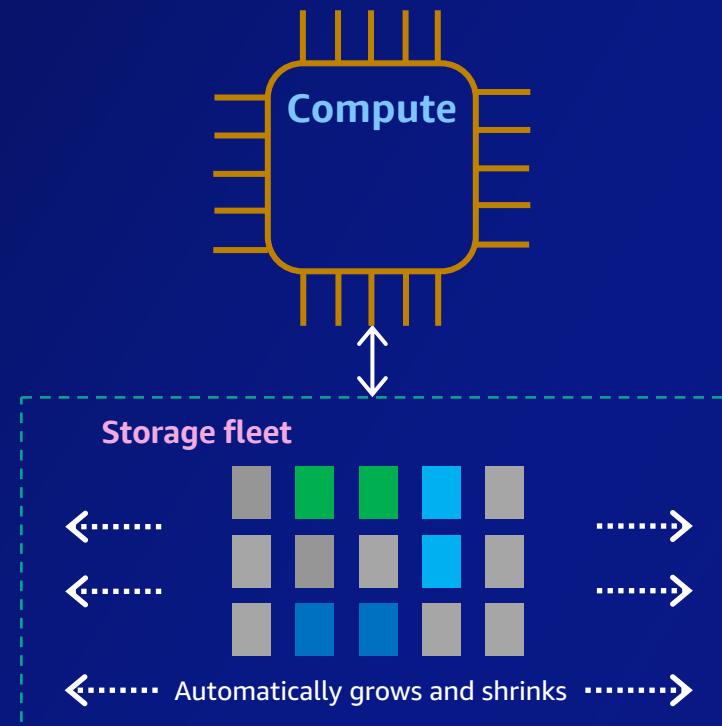
Scalable with up to 128 TB in single database  
Data is written in 10 GB “protection groups,” growing automatically when needed

Multi-AZ Aurora clusters supported by 99.99% uptime SLA

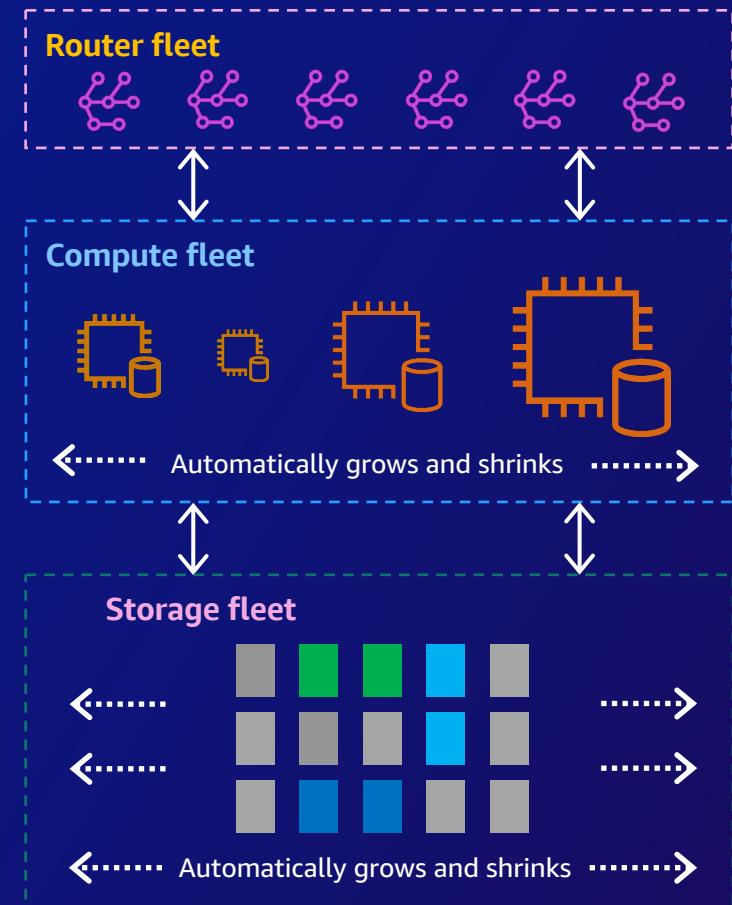


# Amazon Aurora provisioned and serverless V2

## Aurora-provisioned DB cluster

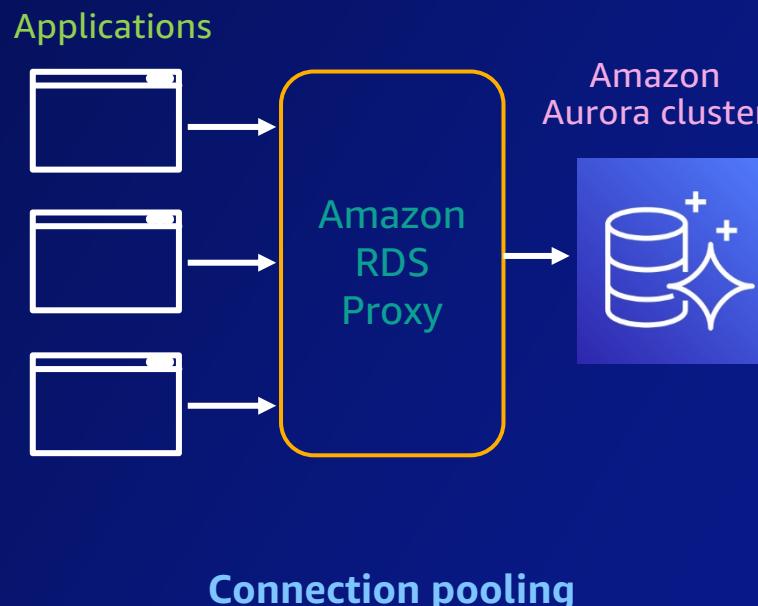


## Aurora serverless V2



# Improve application scalability, availability, and security with Amazon RDS Proxy

Supports a large number of application connections



Deployed across multiple AZs and supports failover without losing a connection

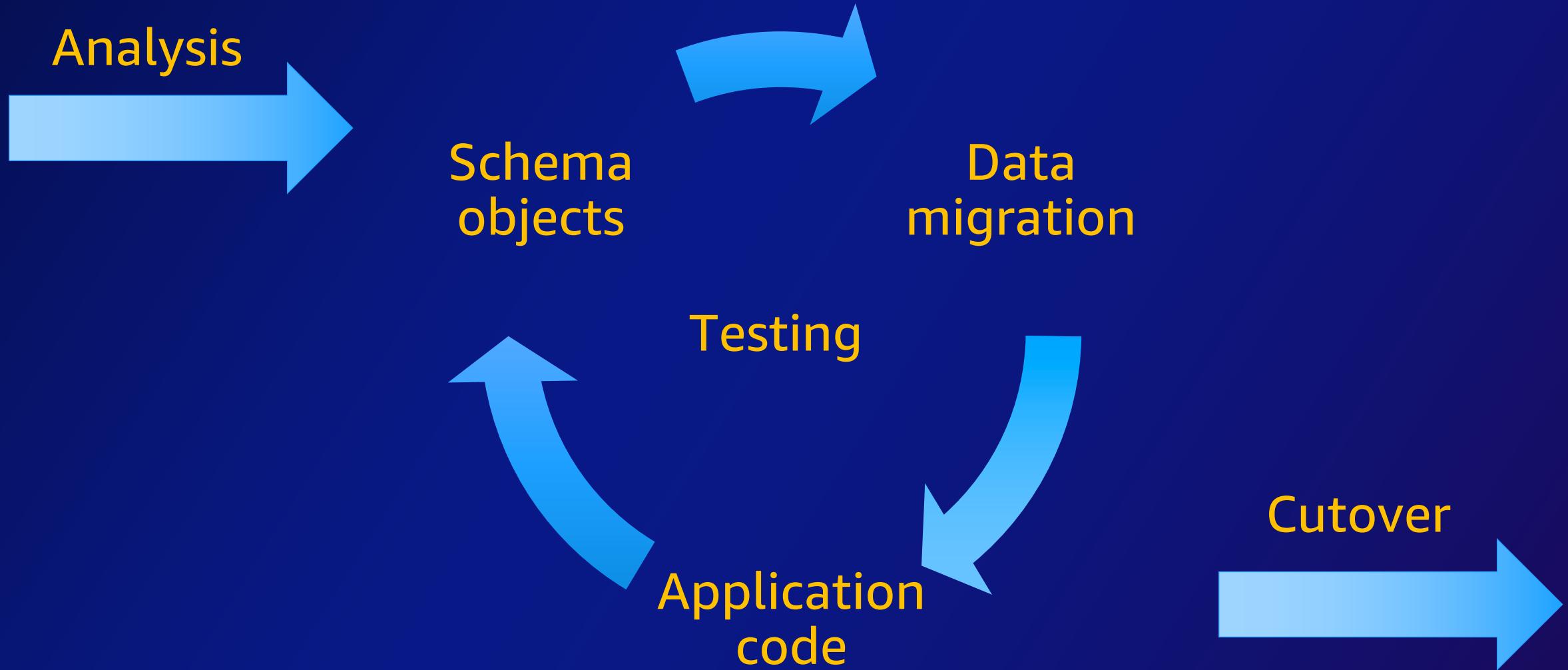
Integrates with AWS Secrets Manager and AWS Identity and Access Management (IAM)

Get started with a few clicks in the console

More: <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/rds-proxy.html>

# Migrations tools and services

# Database migration overview



# Example migration plan

Analysis  Schema conversion  Application conversion  Data migration  Cutover	Assessment	2%
	Planning	7%
	Database conversion	14%
	Procedure, function, and db script conversion	25%
	Application conversion/remediation	5%
	Integration with 3 <sup>rd</sup> party applications	2%
	Data migration	3%
	Functional end to end testing	28%
	Performance testing	2%
	Integration and deployment	5%
	Training and knowledge transfer	2%
	Documentation and version control	2%
	Post production support	3%

# AWS migration tooling

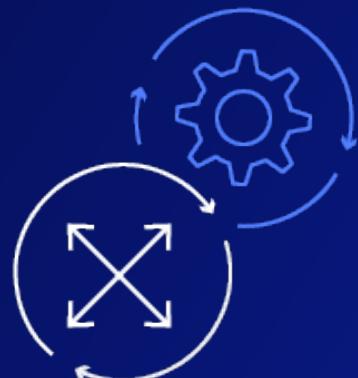


**AWS Schema Conversion Tool (AWS SCT)** converts your commercial database and data warehouse schemas to open-source engines or AWS-native services, such as Amazon Aurora and Amazon Redshift

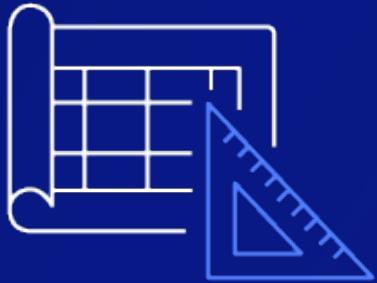
**AWS Database Migration Service (AWS DMS)** easily and securely migrates and/or replicates your databases **and** data warehouses to AWS



# AWS SCT product highlights



**Assess**



**Plan**



**Convert schema  
and code**



**Optimize**



**Migrate  
data warehouses**

# AWS Schema conversion tool

Modernize



Modernize your database tier



ORACLE



Amazon Aurora



Modernize and Migrate your Data

Warehouse to Amazon Redshift

VERTICA



TERADATA



ORACLE



Amazon Redshift

# Database schema conversion using AWS SCT

## SCT Assessment Report

Database migration assessment report

Source database: Oracle Database 12c Enterprise Edition 12.1.0.2.0 (64bit Production), Enterprise edition

### Executive summary

We completed the analysis of your Oracle source database and estimate that 89% of the database storage objects and 98% of database code objects can be converted automatically or with minimal changes. If you select Amazon RDS for PostgreSQL as your migration target, Database storage objects include schemas, tables, table constraints, indexes, types, sequences, synonyms, view-constraints, clusters and database links. Database code objects include triggers, views, materialized views, materialized view logs, procedures, functions, packages, package constants, package cursors, package exceptions, package variables, package functions, package procedures, package types, package collection types, scheduler-jobs, scheduler-programs and scheduler-schedules. Based on the source code syntax analysis, we estimate 99.8% (based on # lines of code) of your code can be converted to Amazon RDS for PostgreSQL automatically. To complete the migration, we recommend 6 conversion action(s) ranging from simple tasks to medium-complexity actions to significant conversion actions.

Migration guidance for database objects that could not be converted automatically can be found [here](#).

### Database objects with conversion actions for Amazon RDS for PostgreSQL

Of the total 64 database storage object(s) and 23 database code object(s) in the source database, we identified 63 (98%) database storage object(s) and 22 (96%) database code object(s) that can be converted to Amazon RDS for PostgreSQL automatically or with minimal changes.

The target database version is less than PostgreSQL 11.1 (11.1). The converted code might not work properly. 1 (2%) database storage object(s) require 1 significant user action(s) to complete the conversion. 1 (4%) database code object(s) require 1 medium user action(s) to complete the conversion.

Figure: Conversion statistics for database storage objects

Object Type	Count	Percentage	Action Type
Table	17	94%	Objects Automatically Converted
Constraint	33	100%	Objects with simple actions
Index	9	100%	Objects with simple actions
Sequence	5	100%	Objects with simple actions

Legend: Green bar = Objects Automatically Converted, Yellow bar = Objects with simple actions, Orange bar = Objects with medium-complexity actions.

Database migration assessment report

Source database: Oracle Database 12c Enterprise Edition 12.1.0.2.0 (64bit Production), Enterprise edition

### Package Procedure Changes

Not all package procedures can be converted automatically. You'll need to address these issues manually.

Issue 5584: Converted functions depends on the time zone settings  
Recommended action: Review the transformed code, and set time zone manually if necessary.  
Issue code: 5584 | Number of occurrences: 2 | Estimated complexity: Simple  
Documentation references: <http://www.postgresql.org/docs/9.6/static/functions-datetime.html>  
Schemas.DMS\_SAMPLE.Packages.TICKETMANAGEMENT.Private procedures.TRANSFERTICKET: 1463:1469  
Schemas.DMS\_SAMPLE.Packages.TICKETMANAGEMENT.Public procedures.SELLTICKETS: 1772:1778

### Package Function Changes

Not all package functions can be converted automatically. You'll need to address these issues manually.

Issue 5644: Unable automatically convert assign operation of array or global nested table, because of nested record  
Recommended action: Perform a manual conversion.  
Issue code: 5644 | Number of occurrences: 1 | Estimated complexity: Medium  
Schemas.DMS\_SAMPLE.Packages.TICKETMANAGEMENT.Private functions.GET\_OPEN\_EVENTS: 270:297

### Procedure Changes

Not all procedures can be converted automatically. You'll need to address these issues manually.

Issue 5103: Unable to convert hints  
Recommended action: Use PostgreSQL methods for performance tuning.  
Issue code: 5103 | Number of occurrences: 2 | Estimated complexity: Simple  
Documentation references: <http://www.postgresql.org/docs/9.6/static/geno.html>  
Schemas.DMS\_SAMPLE.Procedures.GENERATESEATS: 2041:2089  
Schemas.DMS\_SAMPLE.Procedures.GENERATE\_TICKETS: 290:1184

## SCT Schema Conversion

AWS profile: None selected

Oracle

- Issue 5103: Unable to convert hints  
Recommended action: Use PostgreSQL methods for performance tuning.  
Number of occurrences: 2 | Documentation reference(s): <http://www.postgresql.org/docs/9.6/static/geno.html>
- Issue 5584: Converted functions depends on the time zone settings  
Recommended action: Review the transformed code, and set time zone manually if necessary.  
Number of occurrences: 2 | Documentation reference(s): <http://www.postgresql.org/docs/9.6/static/functions-datetime.html>
- Issue 5644: Unable automatically convert assign operation of array or global nested table, because of nested record  
Recommended action: Perform a manual conversion.  
Number of occurrences: 1

Function: GET\_OPEN\_EVENTS (Number of occurrences: 1)

Properties SQL Parameters Related converted objects Mapping

```
01 function get_open_events return eventTab IS
02   event_tab eventTab;
03
04   CURSOR open_events IS
05   SELECT *
06   FROM sporting_event
07   WHERE sold_out = 0
08   ORDER BY start_date_time;
09   row_num BINARY_INTEGER := 1;
10
11  BEGIN
12    FOR oe_rec IN open_events LOOP
13      event_tab(row_num) := oe_rec;
14      row_num := row_num + 1;
15    END LOOP;
16    return event_tab;
17  END;
```

Amazon Aurora (PostgreSQL compatible)

Properties SQL Parameters

```
01 CREATE OR REPLACE FUNCTION dms_sample.ticketmanagement$get_open_events RETURNS VARCHAR AS
02 $BODY$
03 DECLARE
04   event_tab VARCHAR(100) := 'event_tab';
05   open_events CURSOR FOR
06   SELECT *
07   FROM dms_sample.sporting_event
08   WHERE sold_out = 0
09   ORDER BY start_date_time;
10   row_num INTEGER := 1;
11
12  BEGIN
13    FOR oe_rec IN open_events LOOP
14      event_tab(row_num) := oe_rec;
15      row_num := row_num + 1;
16    END LOOP;
17    RETURN event_tab;
18  END;
```

ticketmanagement\$generate  
ticketmanagement\$generate  
ticketmanagement\$sellticket  
ticketmanagement\$sellandc  
ticketmanagement\$transfert  
ticketmanagement\$get\_ope  
ticketmanagement\$init  
generatesets  
loadmlplayers  
loadmlteams  
loadnlplayers  
loadnlteams  
Trigger functions [3]  
User defined types [2]

- Connect AWS SCT to source and target databases
- Address warnings and issues



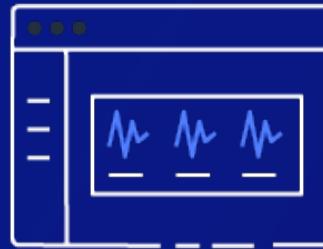
# AWS DMS product highlights



**Secure**



**Assess**



**Validate**



**AWS Snowball  
integration**



**Monitor**



**Stream data**



**Low cost**



**Multiple options**

# AWS DMS supported sources and targets

	Relational	NoSQL	Analytics	Data warehouse
Sources	Oracle SQL Server PostgreSQL MySQL MariaDB Amazon Aurora SAP ASE IBM Db2 LUW IBM Db2 z/OS Azure SQL GCP MySQL	MongoDB Cassandra Amazon DocumentDB	Amazon S3 AWS Snowball	Oracle SQL Server Netezza Greenplum Teradata Vertica Azure Synapse
Targets	Oracle SQL Server PostgreSQL MySQL MariaDB Amazon Aurora SAP ASE	Amazon DynamoDB Amazon DocumentDB Amazon Neptune Amazon ElastiCache Redis, Memcached	Amazon OpenSearch Service Amazon Kinesis Data Streams Amazon S3 Amazon Managed Streaming for Apache Kafka	Amazon Redshift

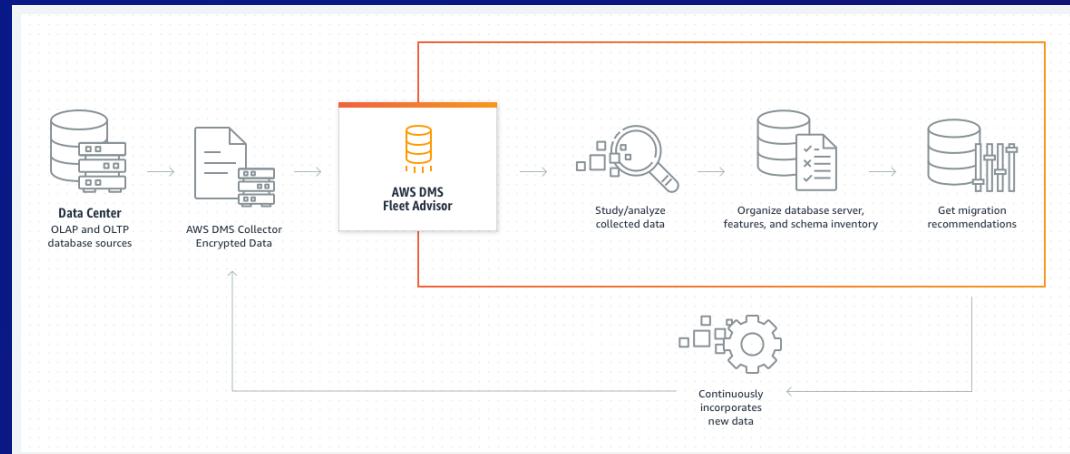
# AWS Database Migration Service (AWS DMS) fleet advisor



AWS DMS Fleet Advisor

[https://aws.amazon.com/  
dms/fleet-advisor/](https://aws.amazon.com/dms/fleet-advisor/)

- Discover and inventory
- Collect metadata
- Gather usage data
- Analyze infrastructure



# 775,000+ databases migrated using AWS DMS



U.S. Department  
of Veterans Affairs



Nielsen

Johnson & Johnson



Expedia



THOMSON REUTERS



Shaadi.com



APPRiSS



halodoc



hotelbeds

verizon<sup>✓</sup>

oddity



yapstone

entradas.com

Sotheby's



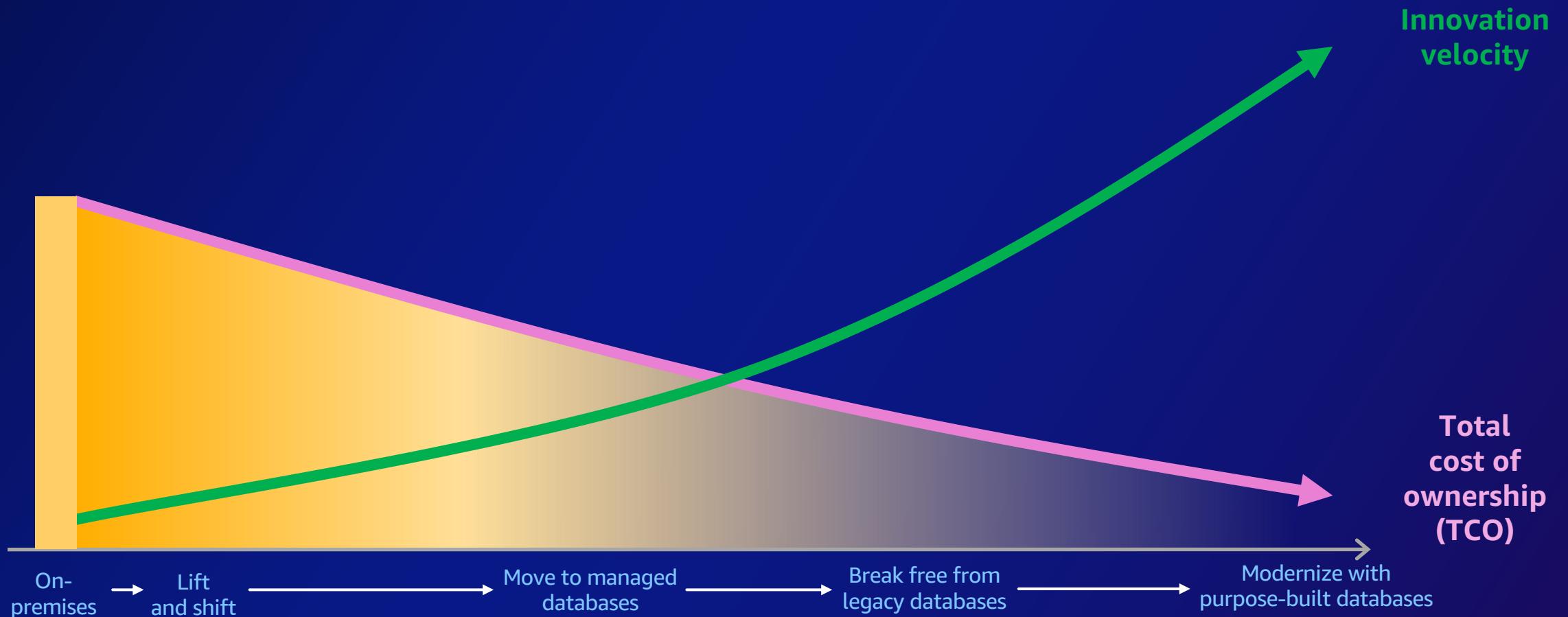
CUPONATION



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Modern data architecture

# Both migration and modernization are important



# Modern Data Architecture (MDA) on AWS



**SCALABLE DATA LAKES**

**PURPOSE-BUILT  
DATA SERVICES**

**UNIFIED DATA ACCESS**

**UNIFIED GOVERNANCE**

**PERFORMANT AND  
COST-EFFECTIVE**

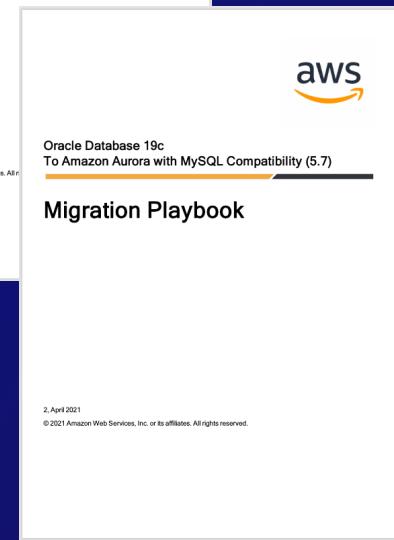
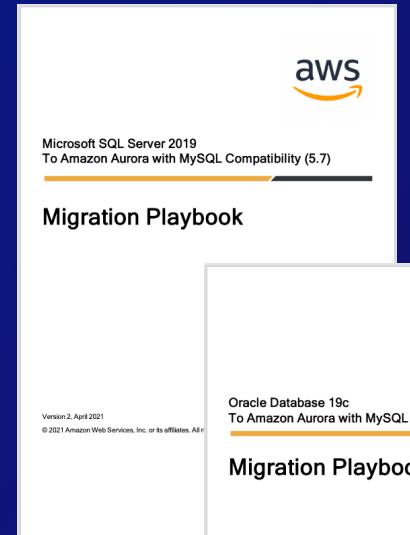
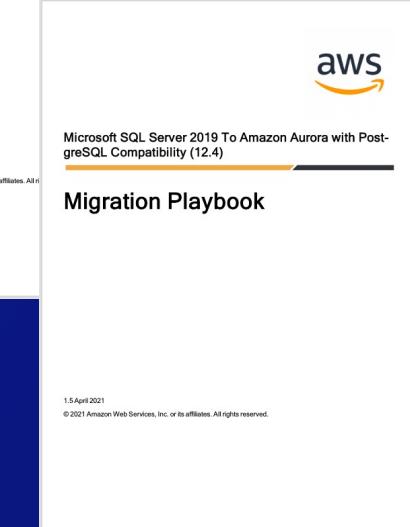
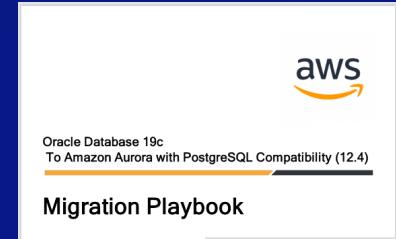
# Database migration playbooks



Migration  
playbooks

[aws.amazon.com/  
dms/resources](https://aws.amazon.com/dms/resources)

Detailed guides  
describing feature by  
feature differences  
between databases and  
techniques on how to  
address them



# Amazon Database Migration Accelerator (Amazon DMA) – Advisors



## Amazon DMA – Advisors

[aws.amazon.com/solutions/databasemigrations/database-migration-accelerator](https://aws.amazon.com/solutions/databasemigrations/database-migration-accelerator)

- **Amazon DMA Advisors** – Database migration specialists with over 100,000 hours of cumulative experience with technologies like PostgreSQL, MySQL, and other AWS databases
- **Amazon DMA Connect** – Provides self-service resources like questionnaires, guides, webinars, and live office hours
- **Amazon DMA Fixed-Price Migrations** – AWS converts the database and application, leveraging AWS tooling and AWS database experts

# AWS Migration Acceleration Program (MAP) & AWS Optimization and Licensing Assessment (AWS OLA)



MAP

[aws.amazon.com/migration-acceleration-program](https://aws.amazon.com/migration-acceleration-program)

- Accelerate cloud migration with outcome-driven methodology
- Tools to automate and accelerate execution and reduce costs
- Programs for AWS investment, including:

AWS OLA for database to help customers plan migration paths through complex database licensing scenarios

# Database freedom



## Database Freedom

[aws.amazon.com/solutions  
/databasemigrations/datab  
ase-freedom](https://aws.amazon.com/solutions/databasemigrations/database-freedom)

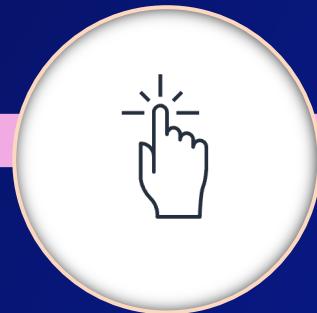
- Help customers to retire technical debt by migrating away from commercial systems to modern, flexible, cloud-native systems
- Advise customers on application architecture, migration strategies, and prioritization
- Support in building a strong migration business case for executive buy-in

# Migrations tools and services

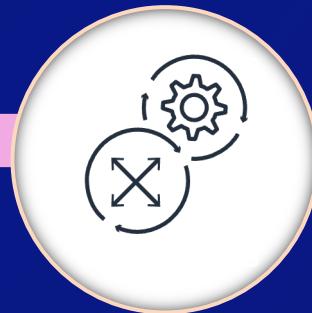
Create a case for change      Mobilize through experiences      Accelerate migration at scale



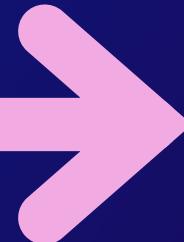
Assess



Mobilize



Migrate and modernize



Tools

AWS DMS Fleet Advisor

AWS Schema Conversion Tool  
(AWS SCT)

AWS Database Migration Service  
(AWS DMS)

People

AWS Solutions Architects

AWS Partners and  
Professional Services

Amazon Database Migration Accelerator  
advisors

Programs

AWS MAP and OLA

Database Freedom

Database migration  
best practices, playbooks, and guides

Amazon Database Migration Accelerator –  
fixed price

# Next steps



Learn more about AWS databases: [aws.amazon.com/products/databases](https://aws.amazon.com/products/databases)



Check out our Amazon DMA Connect Guide, webinars, office hours:  
<https://go.aws/3vY33P4>



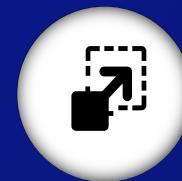
Speak with your AWS account team about how AWS can help your application and database migration/modernization: [aws.amazon.com/products/databases/migrations](https://aws.amazon.com/products/databases/migrations)



AWS Optimization and  
Licensing Assessment  
(AWS OLA)



AWS migration  
tools



AWS Database  
Freedom  
Program



Amazon Database  
Migration Accelerator  
(Amazon DMA)



AWS Partners and  
AWS Professional  
Services

skillbuilder.aws 

# Your time is now

## Build in-demand cloud skills your way



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Thank you!



Please complete the  
session survey

Nethravathi Muddarajaiah  
Senior Database Specialist Solutions Architect  
AWS India