accenture

DATABASE MIGRATION- AWS

Learning and Knowledge Management

LEARNING OBJECTIVES

At the end of this unit, you should be able to:

- Database Migration Challenges
- AWS Database Migration Service
- Replication Instances
- EndPoints
- AWS Schema Conversion Tool
- Homogeneous and Heterogenous database migrations
- Security

MIGRATING TO CLOUD

Understanding Migration

- Moving a meaningful portion - not necessarily all - of your organization’s existing IT assets to the cloud is considered a “migration”

- You can migrate a single data center, a collection of data centers or some other portfolio of systems

Migration to Cloud

- The process of transferring data, application code, and other technology-related business processes from an on-premise or legacy infrastructure to the cloud environment

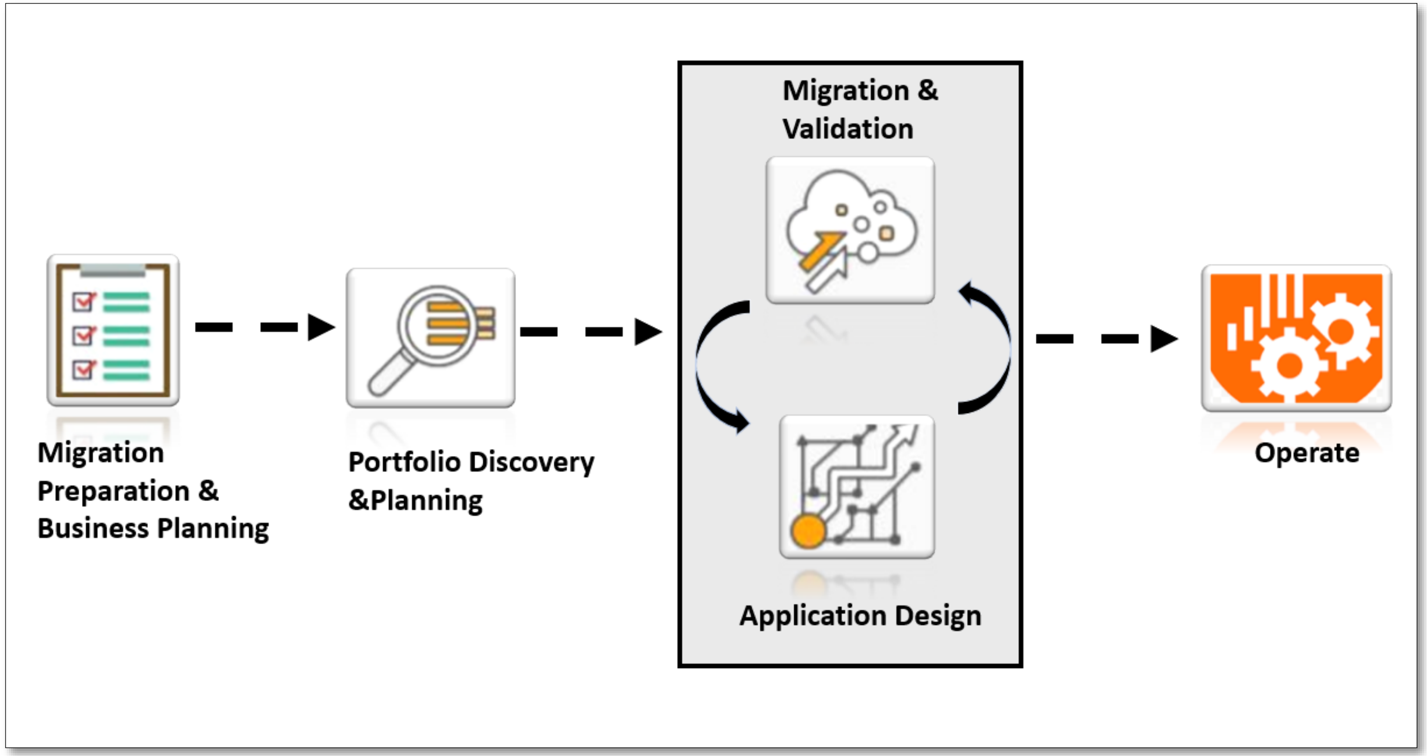
- Migrating to the cloud can be driven by several factors:
  - Data center lease expiration
  - Hardware upgrades
  - Software license renewals
  - Location requirements to meet regulatory compliance
  - Global market expansion
  - Increased developer productivity or the need for a standard architecture

> “There is no one-size-fits-all answer to determining the correct strategy for application migration.” - Stephen Orban

MIGRATING TO CLOUD

Migrating to AWS - THE FIVE-PHASE MIGRATION PROCESS

"With the five-phase migration process, you start with the least complex application to learn how to migrate while learning more about the target platform, and then build toward the more complex applications." -Stephen Orban


Migration Preparation & Business Planning
Portfolio Discovery &Planning
Migration & Validation
Application Design
Operate

Migrating to Cloud

Migrating to AWS - THE FIVE-PHASE MIGRATION PROCESS

Phase 1: MIGRATION PREPARATION AND BUSINESS PLANNING:

-If you don’t have a plan, you may be planning to fail

- Develop a sound business case
- It involves setting out the direction, goals and potential timelines for migrating to AWS
- An executive steering group - consisting of a team of wider business stakeholders which will be required to ensure the right combination of people, processes and technology are in place

PHASE 2: PORTFOLIO DISCOVERY AND PLANNING

-Crawl, Walk, Run

- In order to successfully plan the migration of workloads from private data centers to AWS, we must understand the complexity, interdependencies of the applications and the level of business dependency on these workloads
- It is important to collect data on resource utilization, configurations and their behavioral patterns.

Migrating to Cloud

Migrating to AWS - THE FIVE-PHASE MIGRATION PROCESS

- Beginning the migration process with less critical and complex applications in your portfolio creates a sound learning opportunity for your team to exit their initial round of migration with:
  - Confidence that they are not practicing with mission critical applications in the early learning stages
  - Foundational learnings they can apply to future iterations
  - Ability to fill skills and process gaps and positively reinforce best practices now based on experience

PHASE 3 / PHASE 4: DESIGNING, MIGRATING, AND VALIDATING APPLICATIONS

- Agile, Flexible, Iterative

- The execution phase of the migration plan where the focus is on the actual migration of applications to AWS
- Each application is designed, migrated, and validated according to one of the six common migration strategies (“The 6 R’s”)
- Integrating a continuous improvement methodology and iteratively reviewing the design, migration and validation phases is also recommended for successful migration to Cloud

Migrating to Cloud

Migrating to AWS - THE FIVE-PHASE MIGRATION PROCESS

PHASE 5: MODERN OPERATING MODEL

- Shift to cloud

- As applications are migrated, a new foundation is optimized and old systems is removed, and constantly iterate toward a modern operating model

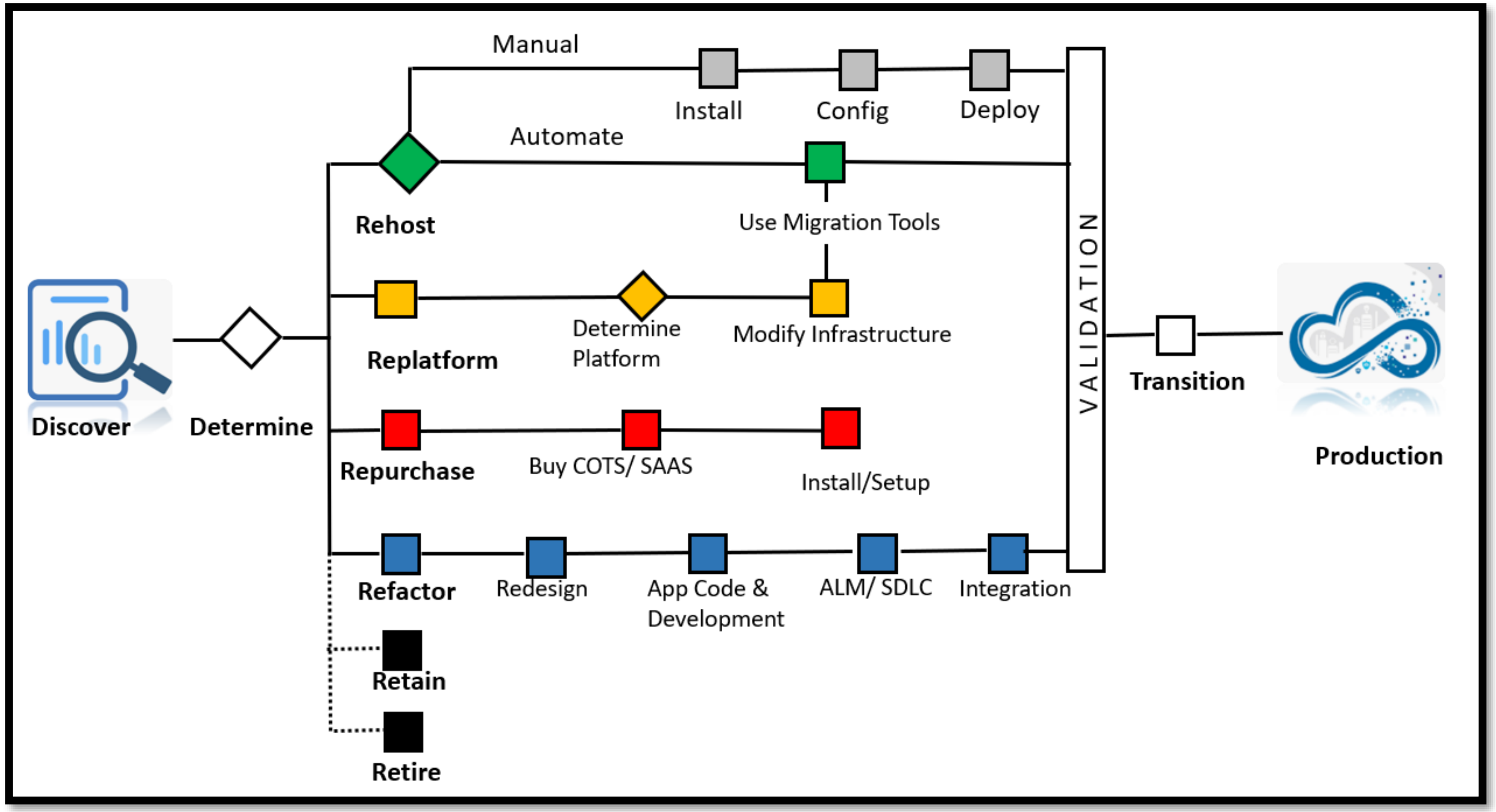
- The combination of people, processes and technologies will be required for managing the IT estate post migration, continually improving as the migration accelerates

- First few application migrations can be used to develop the foundation, and the operating model will continually improve and become more sophisticated as migration accelerates

Migrating to Cloud can be a daunting task for an organization that has primarily operated in an on-premise environment. However, taking the phased approach with due diligence on preparation, discovery and planning before embarking on the actual migration will ensure certainty in realising the business objectives.

Migrating to Cloud

Migrating to AWS - Six Common Strategies for Migration: “The 6 R’s”



- Enterprises typically begin to contemplate how to migrate an application during the second phase of the “**Migration Process**”—Portfolio Discovery and Planning.

- Using **6 R’s**, organizations can outline a plan (which should be considered subject to change as they progress through their migration) on how they’ll approach migrating each of the applications in their portfolio and in what order.

Migrating to Cloud

Migrating to AWS - Six Common Strategies for Migration: “The 6 R’s”

6 different migration strategies are used to migrate applications to the cloud.

1.REHOST – Also known as “lift-and-shift.”

The rehosting involves the migration of one or multiple applications to the cloud without making any changes before the transfer.

- The “lift and shift” strategy is often used by companies in need of a quick and simple transition to the cloud to meet a specific business objective
- This approach has many benefits like simplicity, quickness, and the option for almost complete automation
- Rehosting often doesn’t allow to take advantage of all the opportunities the cloud has to offer straight away
- Rehosting can be performed manually, or it can be automated with the help of a tool like AWS Server Migration Service.

Migrating to Cloud

Migrating to AWS - Six Common Strategies for Migration: “The 6 R’s”

2. Replatform – Known as “Lift, Tinker and Shift”

The Replatforming includes some customization or optimizations of the digital assets before the move to the cloud.

- the “lift, tinker and shift” phase does not include changes to the core architecture of applications, unlike the refactor phase
- changes provide more flexibility in comparison to the rehosting method and assure more of the cloud’s capabilities can be leveraged after the migration
- As customization is required beforehand, this approach is bit more complicated
- For example, there is a need to reduce the amount of time spend in managing database instances, so move to a database-as-a-service offering like Amazon Relational Database Service (Amazon RDS)

Migrating to Cloud

Migrating to AWS - Six Common Strategies for Migration: “The 6 R’s”

3. Repurchase– Known as “Drop and Shop”

The repurchasing includes the retirement of old solutions in favor of newer upgraded ones or entirely scrapping some applications and directly using different cloud-based alternatives

- The “drop and shop” process requires a detailed audit of your current portfolio of digital assets (which you’re going to do before the migration begins regardless of your strategy)

- During this time, opportunities for improvement can be identified in specific areas through the implementation of upgraded applications or entirely new ones

Migrating to Cloud

Migrating to AWS - Six Common Strategies for Migration: “The 6 R’s”

4. Refactor –known as “ Re-architect “

Refactor includes fundamental changes to your code base in order to turn non-cloud applications into cloud applications.

- Refactoring is often the most complex and expensive phase while it provides the most significant benefits in terms of taking full advantage of the cloud

- Refactoring allows for the implementation of features and scale to match current business needs

Migrating to Cloud

Migrating to AWS - Six Common Strategies for Migration: “The 6 R’s”

5.Retire – Decommission or archive unneeded portions of IT portfolio

- Identifying IT assets that are no longer useful and can be turned off will help boost your business case, and direct your team’s attention toward maintaining the resources that are widely used

6. Retain – Do nothing, for now—revisit.

- Organizations retain portions of their IT portfolio because there are some that they are not ready to migrate and feel more comfortable keeping them on-premises, or they are not ready to migrate an application that was recently upgraded and then make changes to it again

- As the IT portfolio moves to the cloud, there will be fewer reasons to retain

Migrating to Cloud

Migrating to AWS - Scenarios

Scenario -1:
The server for Tom’s meme generator is seeing a lot more traffic these days. He must address the performance. He has looked at the CPU and memory usage. It has two allocated CPUs, and Tom is seeing spikes of up to 100% usage multiple times throughout the day. The memory usage is within limits; it must be the rendering process for the graphics that is taking too much of the CPU. Tom has looked at the VMware cluster, and he has plenty of capacity. He considers adding just two more CPUs to compensate

.

- In AWS, when working with instance sizing, you cannot adjust CPU and memory separately
- Increasing CPU’s in AWS, will result in increasing memory as well
- Vertical scaling adds more capacity to a single server and horizontal scaling adds more servers to meet the load of application
- With AWS pay-as-you-go model, you can gain significant cost reduction by using horizontal scaling
- For Horizontal scaling, AWS offers, ***Elastic Load Balancing*** and ***AWS Auto Scaling***

Migrating to Cloud

Migrating to AWS - Scenarios

- In this scenario, the perfect solution is to switch from a vertical scaling solution used on-premises to horizontal scaling in the cloud

Tip: Always start with horizontal scaling and work back from there to find technical reasons why it will not work. Only then revert to vertical scaling

.

Migrating to Cloud

Migrating to AWS - Scenarios

Scenario -2:
Sam’s company is located in Washington, DC, and all of its corporate users are located in that office except for a few remote sales users. The largest customer using its online application—representing 90% of the consumption—is in Seattle, and the rest of the users are spread across the United States

- First, select an AWS region with proximity to staff to reduce latency (us-east-1 region in Virginia)

.

- Second, Sam’s company has largest customer is in Seattle, AWS has regions all around the US; Select us-west-2 region in Oregon as a second deployment location for online application

- Deploying in two regions like this on-premises would be costly

Migrating to Cloud

Migrating to AWS - Scenarios

Easy Access to Newer Technologies:

Scenario 3:
Amy works for a hospital, and someone from purchasing is asking the IT department to create an AI program to estimate supply need. Some supplies are expiring before they use them. However, sometimes if they order less, they run out. She is looking for a program to estimate when and how many to order based on historical patient data

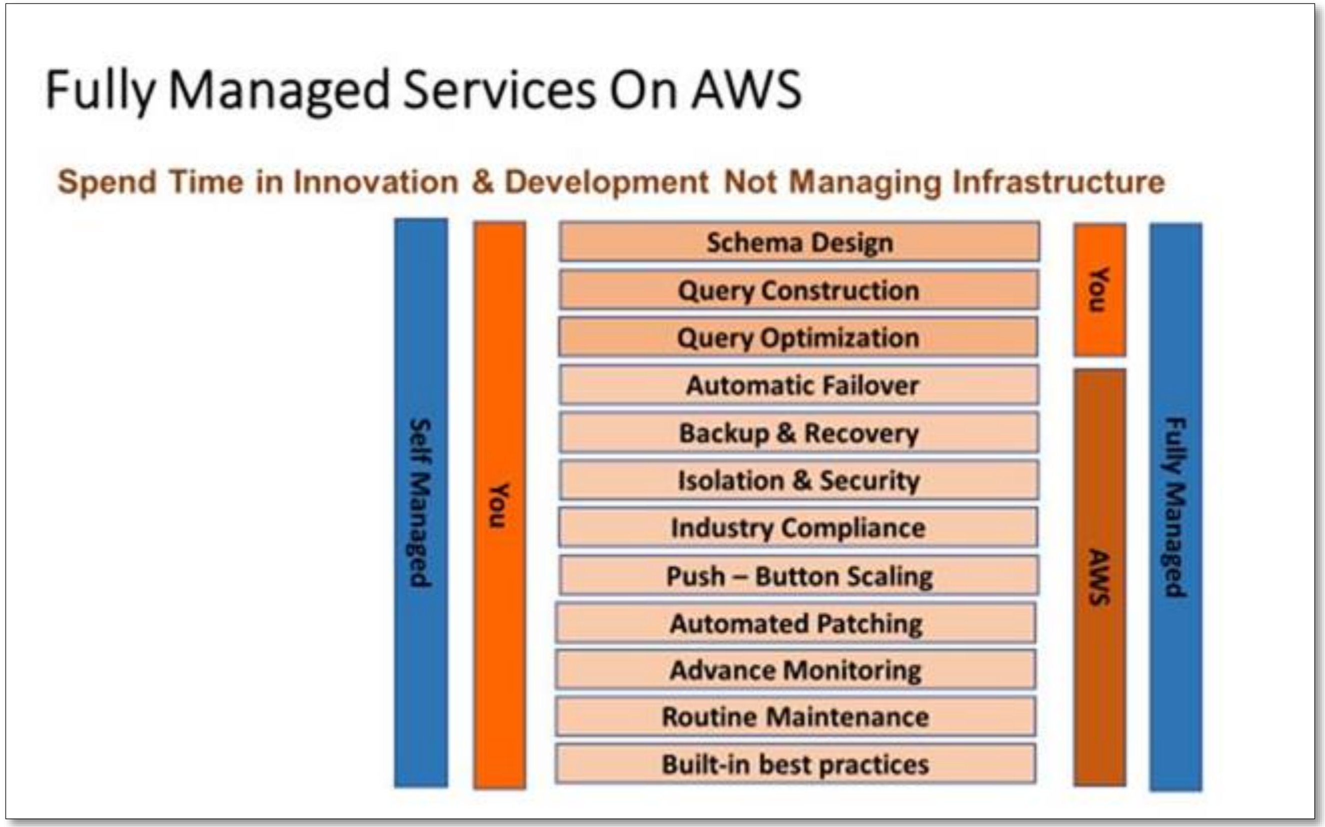
- AWS service Amazon Forecast can be used for AI computation
- Amy, no need to worry about servers, hardware, or a specialized AI programming skill set to test new technology
- Amy can use current IT staff and programmers to implement a proof of concept (POC) and need to pay only for the training time in hours, data storage in gigabytes (GB), and generated forecasts

Migrating to Cloud

Migrating to AWS – Use Case - Move to manged Databases

Why to move to managed databases?

- Get higher level services and lower costs
  - To get full advantages of Cloud
- Database management is undifferentiated heavy lifting
  - No need to worry about
    - Automated Patching
    - Automated backup and failover
    - High Availability and durability



Fully Managed Services On AWS

Spend Time in Innovation & Development Not Managing Infrastructure

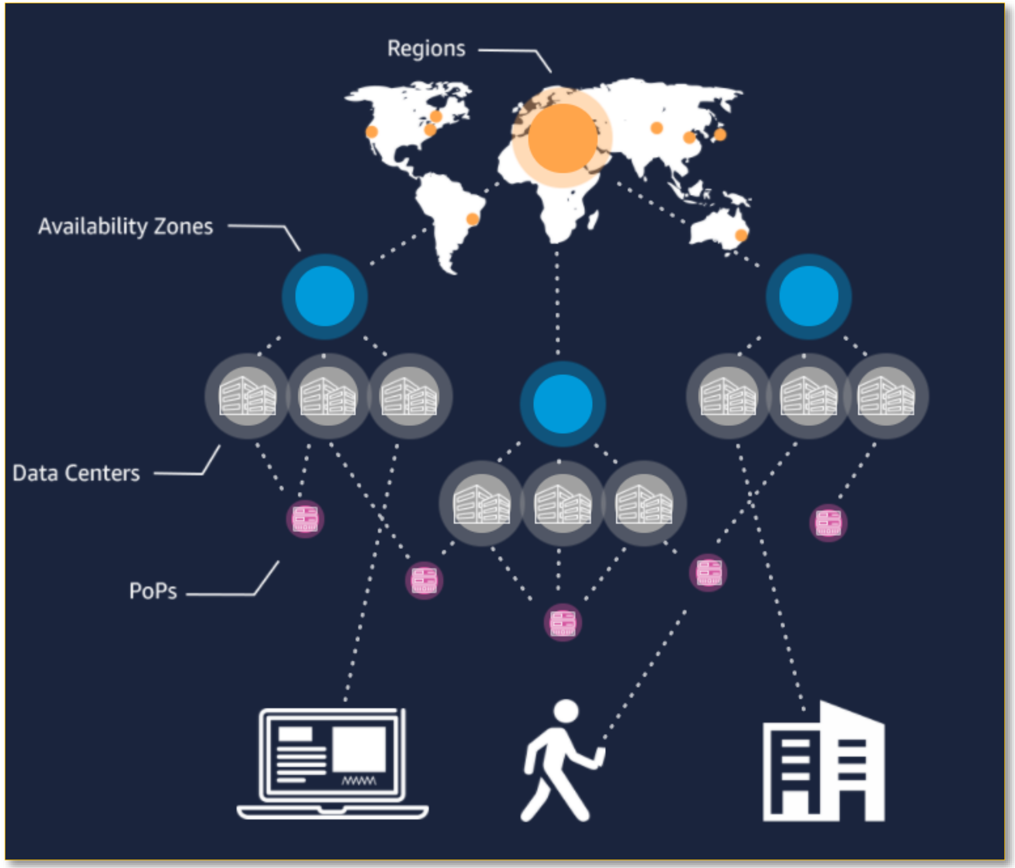
Migrating to Cloud

Migrating to AWS – Use Case - Move to manged Databases

Why to move to managed databases?

- Database failover and recovery are hard to get right
  - The AWS Cloud is split into different AWS Regions
  - Each Region is split into multiple Availability Zones (separate data centers within the Region)
  - Each Availability Zone isolated from the failure of other Availability Zones in the same Region
  - AWS managed databases uses this isolation by running instances of your database in multiple Availability Zones within a Region to avoid failure and downtime

.



Migrating to Cloud

Migrating to AWS – Use Case - Move to manged Databases

Why to move to managed databases?

- Flexible capacity planning and scaling
  - With manged databases, you can increase and decrease your database as needed
  - Database instance size or cluster size can be increased as needed quickly in the console

.

  - Choose a pay-as-you-go model that is based only on the instance size and storage used rather than purchasing annual licenses up front

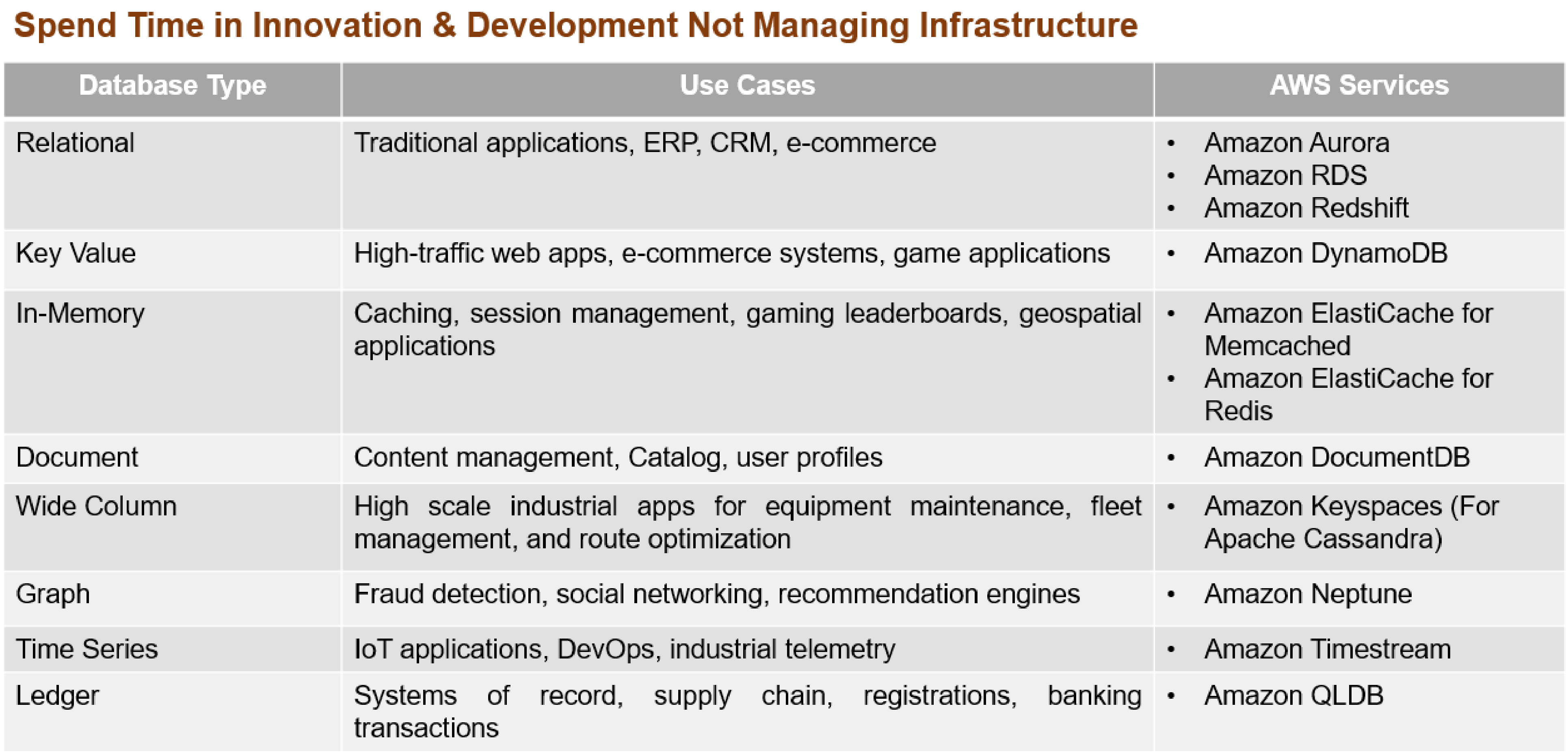
Migrating to Cloud

Migrating to AWS – Use Case - Move to manged Databases

Factors to Consider

- Type of managed database

.



Spend Time in Innovation & Development Not Managing Infrastructure

Database Type	Use Cases	AWS Services
Relational	Traditional applications, ERP, CRM, e-commerce	• Amazon Aurora • Amazon RDS • Amazon Redshift
Key Value	High-traffic web apps, e-commerce systems, game applications	• Amazon DynamoDB
In-Memory	Caching, session management, gaming leaderboards, geospatial applications	• Amazon ElastiCache for Memcached • Amazon ElastiCache for Redis
Document	Content management, Catalog, user profiles	• Amazon DocumentDB
Wide Column	High scale industrial apps for equipment maintenance, fleet management, and route optimization	• Amazon Keyspaces (For Apache Cassandra)
Graph	Fraud detection, social networking, recommendation engines	• Amazon Neptune
Time Series	IoT applications, DevOps, industrial telemetry	• Amazon Timestream
Ledger	Systems of record, supply chain, registrations, banking transactions	• Amazon QLDB

Migrating to Cloud

Migrating to AWS - Move to manged Databases

Factors to Consider

- The process to migrate data
  - AWS Data Migration Service
  - AWS Schema Conversion Tool

.

  - AWS Database Migration Service Partners

Migrating to Cloud

Migrating to AWS - Move to manged Databases

- **AWS Database Migration Service** helps you migrate databases to AWS quickly and securely

- The source database remains fully operational during the migration, minimizing downtime to applications that rely on the database.

- The AWS Database Migration Service can migrate your data to and from the most widely used commercial and open-source databases.

- It supports homogeneous migrations such as Oracle to Oracle, as well as heterogeneous migrations between different database platforms, such as Oracle to Amazon Aurora.

- Migrations can be from on-premises databases to Amazon RDS or Amazon EC2, databases running on EC2 to RDS, or vice versa, as well as from one RDS database to another RDS database.

- It can also move data between SQL, NoSQL, and text-based targets.

    - AWS Schema Conversion Tool
    - AWS Database Migration Service Partners

Migrating to Cloud

Migrating to AWS - Move to manged Databases

AWS Schema Conversion Tool

- The AWS Schema Conversion Tool makes heterogeneous database migrations predictable by automatically converting the source database schema and a majority of the database code objects, including views, stored procedures, and functions, to a format compatible with the target database.

- Any objects that cannot be automatically converted are clearly marked so that they can be manually converted to complete the migration.

- SCT can also scan your application source code for embedded SQL statements and convert them as part of a database schema conversion project.

- During this process, SCT performs cloud native code optimization by converting legacy Oracle and SQL Server functions to their equivalent AWS service thus helping you modernize the applications at the same time of database migration.

- Once schema conversion is complete, SCT can help migrate data from a range of data warehouses to Amazon Redshift using built-in data migration agents.

Migrating to Cloud

Migrating to AWS - Move to manged Databases

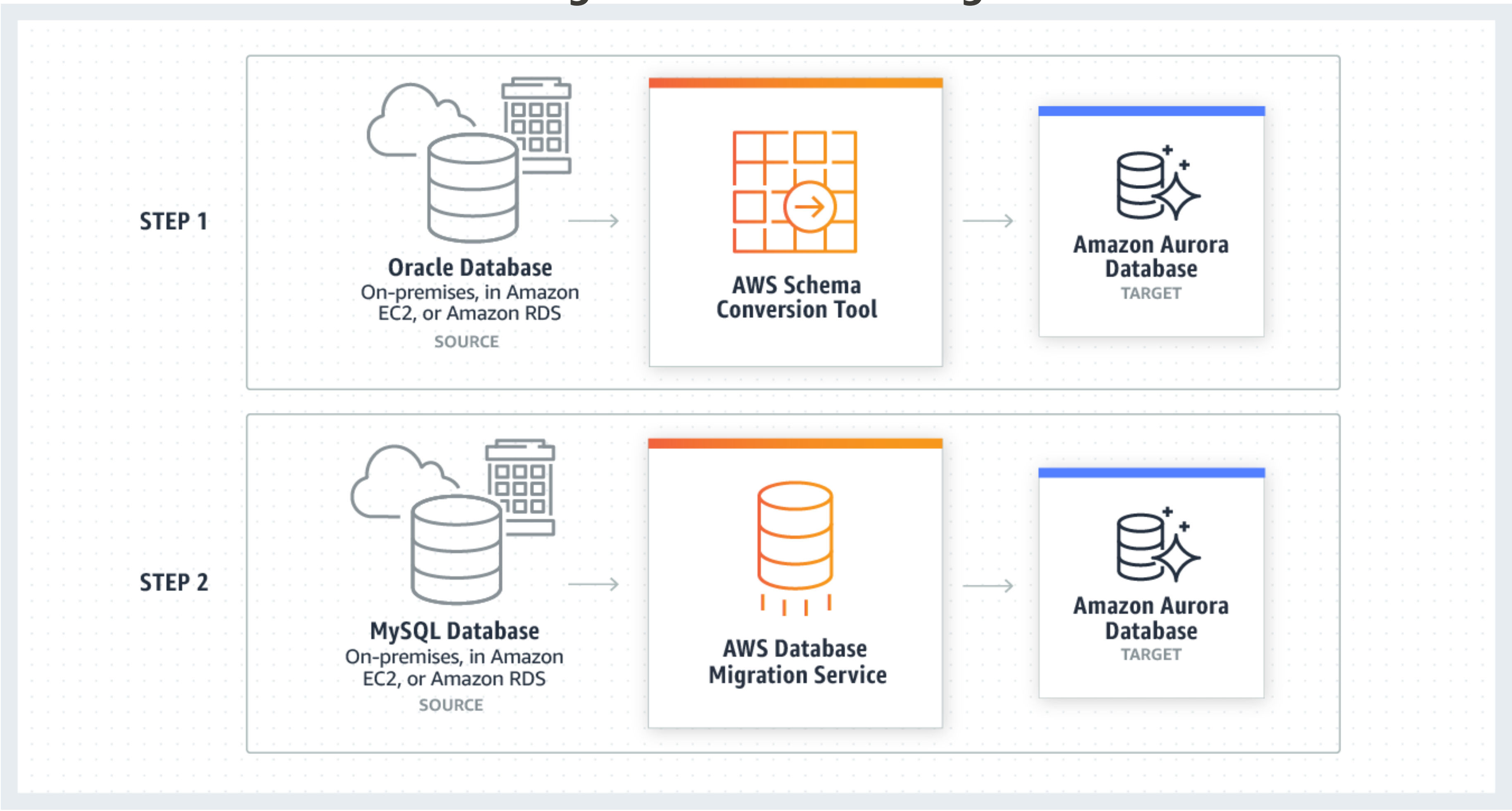
AWS Schema Conversion Tool supports the following conversions -

Source Database	Target Database on Amazon RDS
Oracle Database	Amazon Aurora, MySQL, PostgreSQL, Oracle
Oracle Data Warehouse	Amazon Redshift
Azure SQL	Amazon Aurora, MySQL, PostgreSQL
Microsoft SQL Server	Amazon Aurora, Amazon Redshift, MySQL, PostgreSQL
Teradata	Amazon Redshift
IBM Netezza	Amazon Redshift
Greenplum	Amazon Redshift
HPE Vertica	Amazon Redshift
MySQL and MariaDB	PostgreSQL
PostgreSQL	Amazon Aurora, MySQL
Amazon Aurora	PostgreSQL
IBM DB2 LUW	Amazon Aurora, MySQL, PostgreSQL
Apache Cassandra	Amazon DynamoDB
SAP ASE	RDS for MySQL, Aurora MySQL, RDS for PostgreSQL, and Aurora PostgreSQL

Migrating to Cloud

Migrating to AWS - Move to manged Databases

Heterogeneous database migration



STEP 1

Oracle Database
On-premises, in Amazon EC2, or Amazon RDS
SOURCE

AWS Schema Conversion Tool

Amazon Aurora Database
TARGET

STEP 2

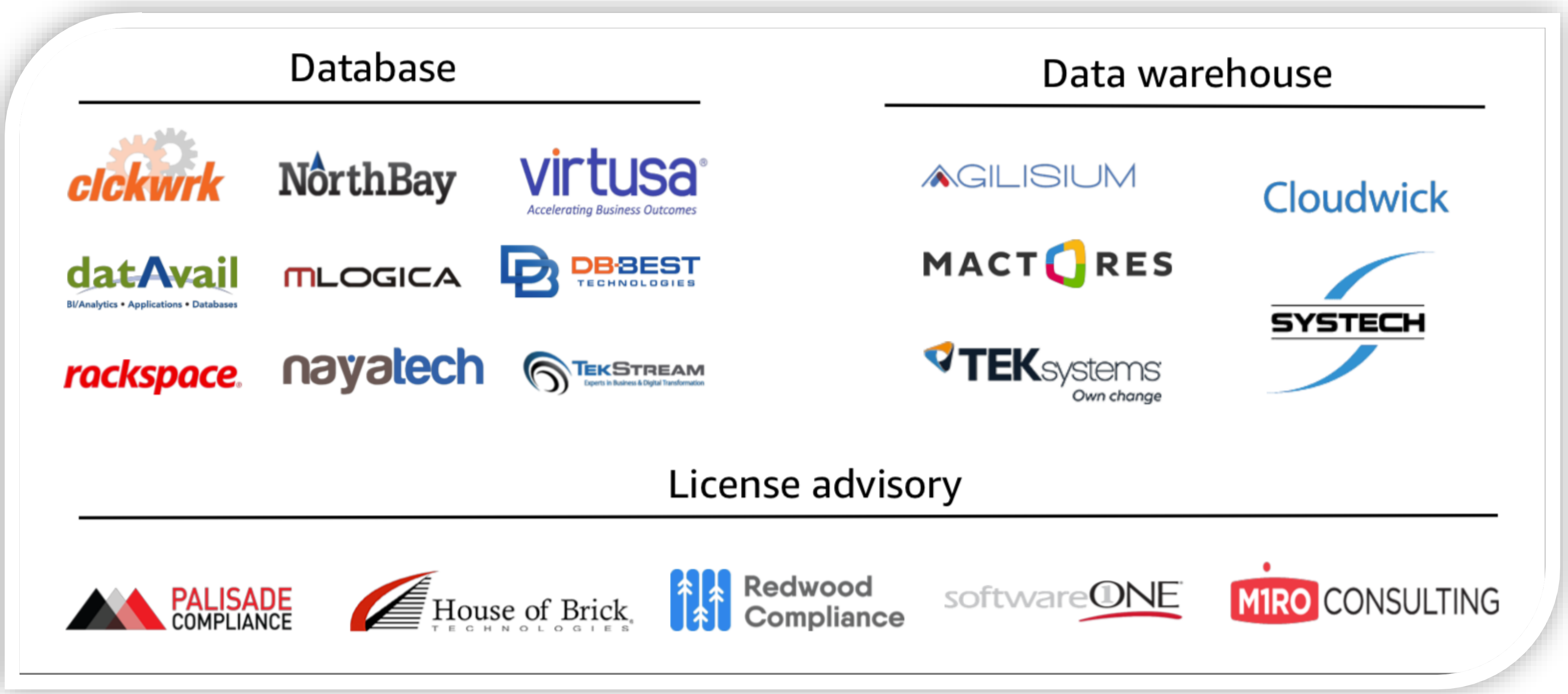
MySQL Database
On-premises, in Amazon EC2, or Amazon RDS
SOURCE

AWS Database Migration Service

Amazon Aurora Database
TARGET

Migrating to Cloud

AWS Database Migration Partners



Database

clckwrk

NorthBay

virtusa
Accelerating Business Outcomes

datAvail
BI/Analytics • Applications • Databases

mLOGICA

DB BEST TECHNOLOGIES

rackspace

nayatech

TekStream
Experts in Business & Digital Transformation

Data warehouse

AGILISIUM

Cloudwick

MACTORES

SYSTECH

TEKsystems
Own change

License advisory

PALISADE COMPLIANCE

House of Brick TECHNOLOGIES

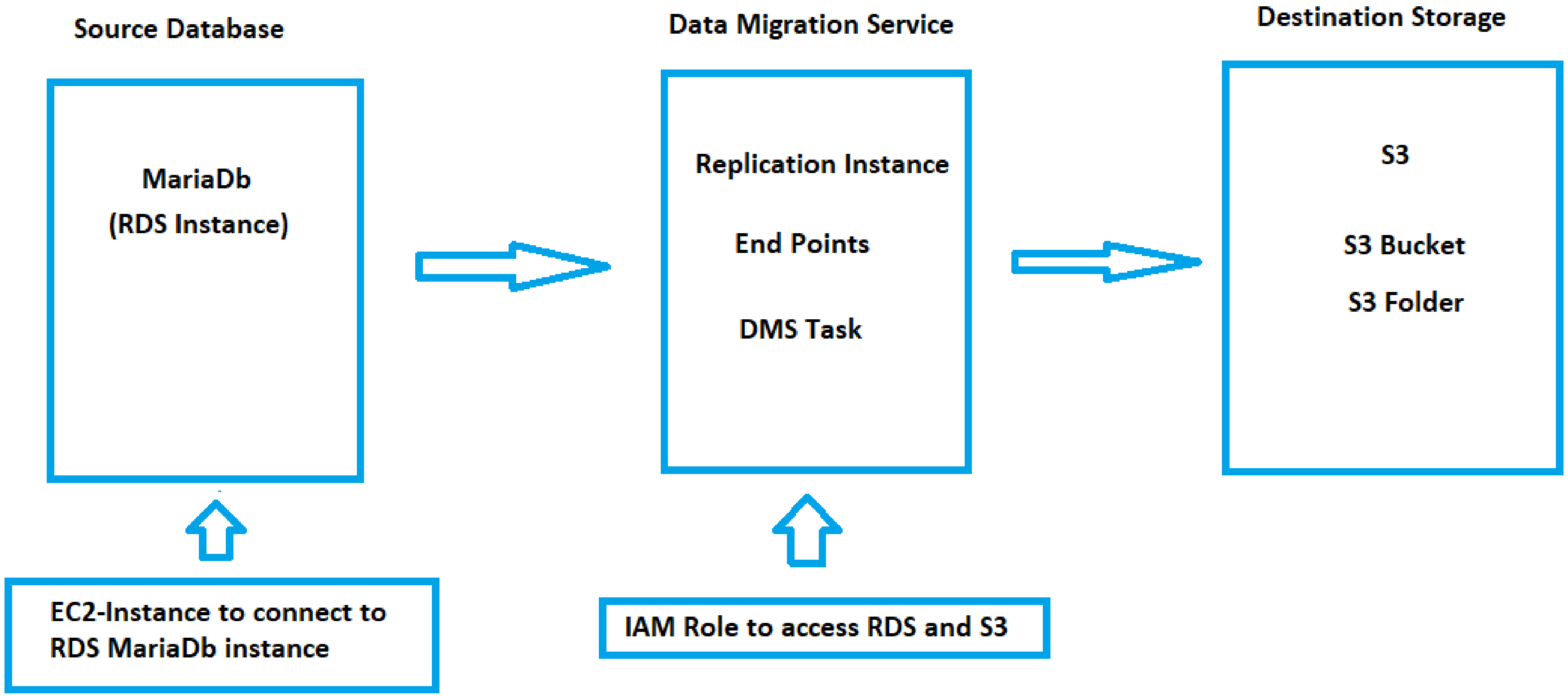
Redwood Compliance

softwareONE

M1RO CONSULTING

Migrating to Cloud

Demo: AWS Database Migration Service MySql → S3



Source Database

MariaDb
(RDS Instance)

EC2-Instance to connect to RDS MariaDb instance

Data Migration Service

Replication Instance

End Points

DMS Task

IAM Role to access RDS and S3

Destination Storage

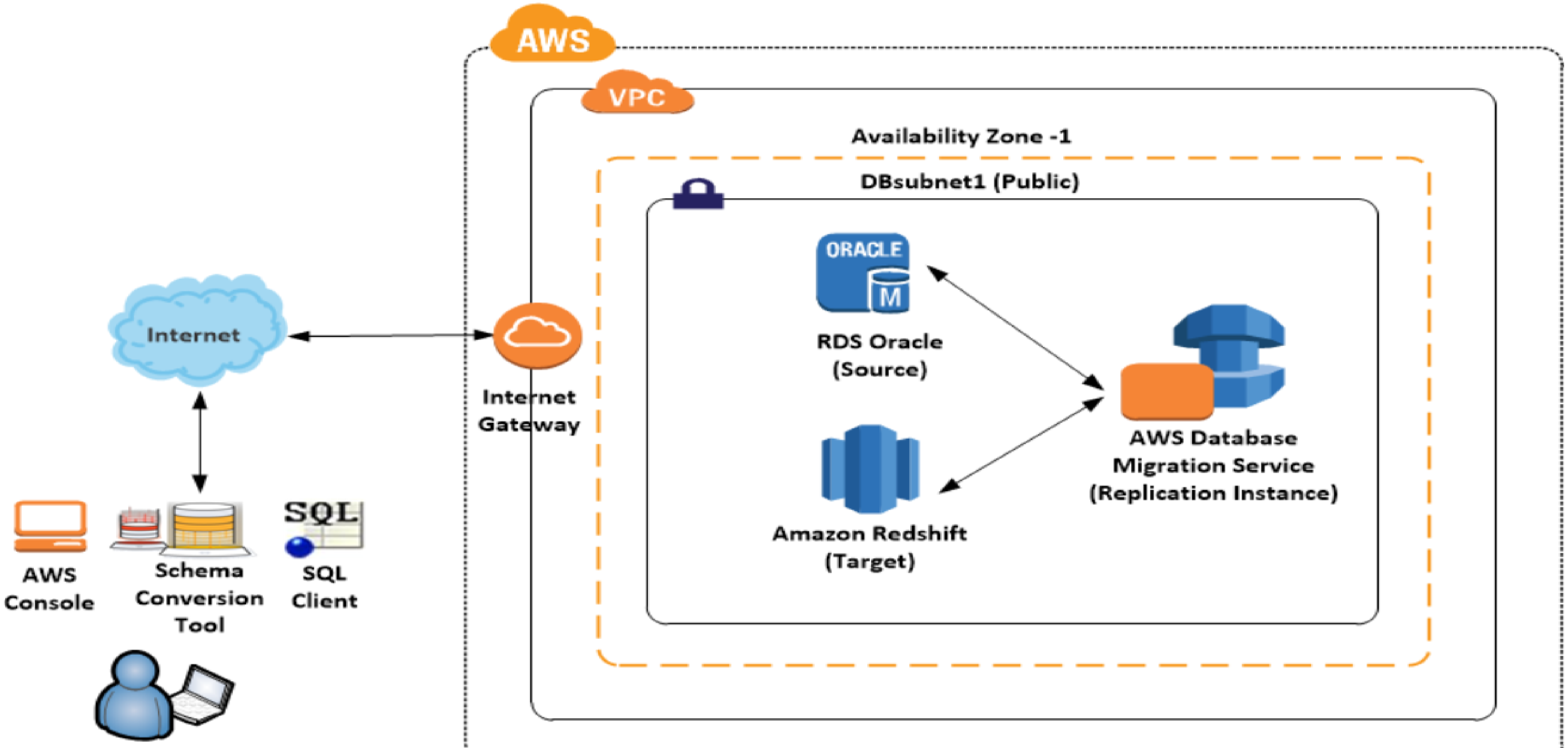
S3

S3 Bucket

S3 Folder

Migrating to Cloud

Demo: AWS Migration from Oracle to Amazon Redshift



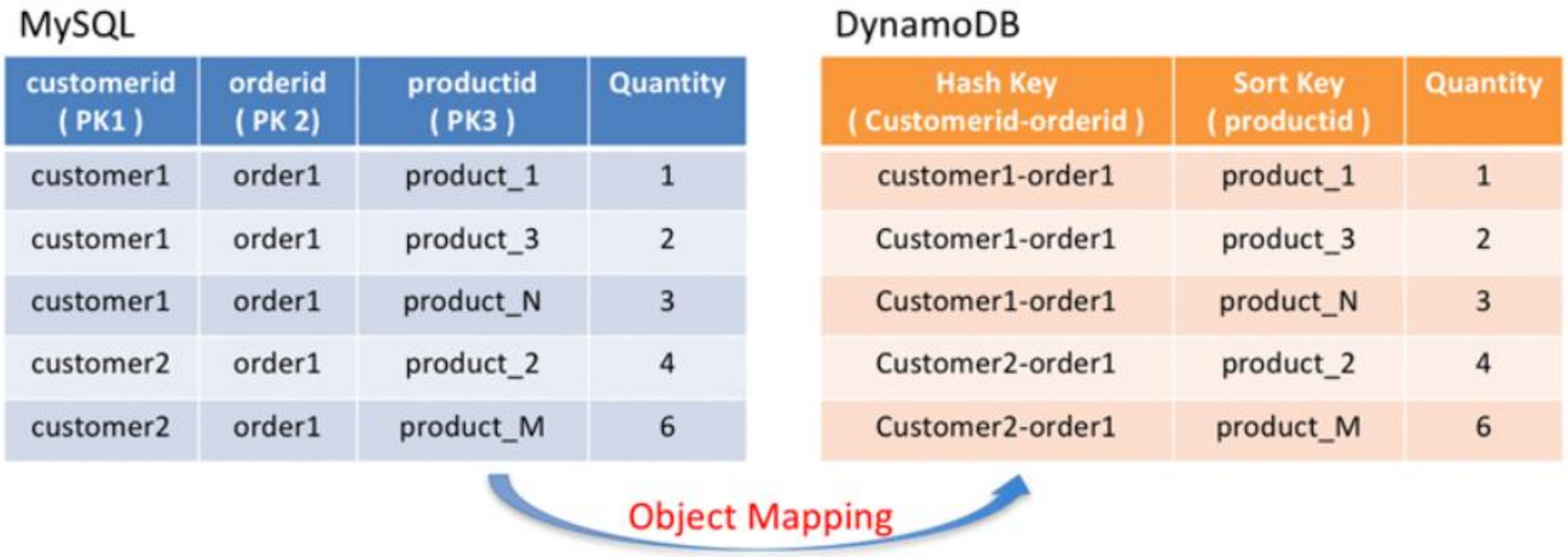
Migrating to Cloud

Demo: AWS Migration from Oracle to Amazon Redshift

Step 1: Launch the RDS Instances in a VPC by Using the CloudFormation Template
Step 2: Install the SQL Tools and AWS Schema Conversion Tool on Your Local Computer
Step 3: Test Connectivity to the Oracle DB Instance and Create the Sample Schema
Step 4: Test the Connectivity to the Amazon Redshift Database
Step 5: Use AWS SCT to Convert the Oracle Schema to Amazon Redshift
Step 6: Validate the Schema Conversion
Step 7: Create an AWS DMS Replication Instance
Step 8: Create AWS DMS Source and Target Endpoints
Step 9: Create and Run Your AWS DMS Migration Task
Step 10: Verify That Your Data Migration Completed Successfully
Step 11: Delete Walkthrough Resources

Migrating to Cloud

Demo: AWS Migration from RDS to Amazon DynamoDB



MySQL

customerid (PK1)	orderid (PK 2)	productid (PK3)	Quantity
customer1	order1	product_1	1
customer1	order1	product_3	2
customer1	order1	product_N	3
customer2	order1	product_2	4
customer2	order1	product_M	6

DynamoDB

Hash Key (Customerid-orderid)	Sort Key (productid)	Quantity
customer1-order1	product_1	1
Customer1-order1	product_3	2
Customer1-order1	product_N	3
Customer2-order1	product_2	4
Customer2-order1	product_M	6