Module 8: Databases

AWS Academy Cloud Foundations

Module overview

Topics

- Amazon Relational Database Service (Amazon RDS)
- Amazon DynamoDB
- Amazon Redshift
- Amazon Aurora

Demos

- Amazon RDS console
- Amazon DynamoDB console

Lab

- Lab 5: Build Your DB Server and Interact with Your DB Using an App

Activity

- Database case studies

Knowledge check

Module objectives

After completing this module, you should be able to:

- Explain Amazon Relational Database Service (Amazon RDS)
- Identify the functionality in Amazon RDS
- Explain Amazon DynamoDB
- Identify the functionality in Amazon DynamoDB
- Explain Amazon Redshift
- Explain Amazon Aurora
- Perform tasks in an RDS database, such as launching, configuring, and interacting

Section 1: Amazon Relational Database Service

Module 8: Databases

Amazon Relational Database Service

Amazon Relational Database Service (Amazon RDS)

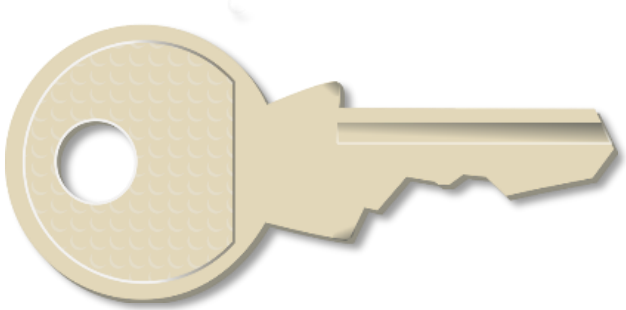
Unmanaged versus managed services

Unmanaged:

Scaling, fault tolerance, and availability are managed by you.

Managed:

Scaling, fault tolerance, and availability are typically built into the service.

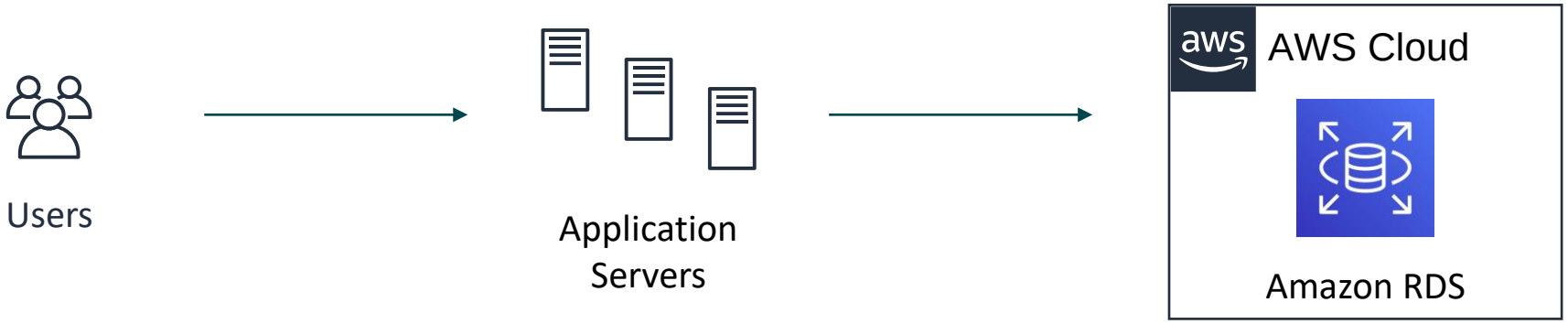


Challenges of relational databases

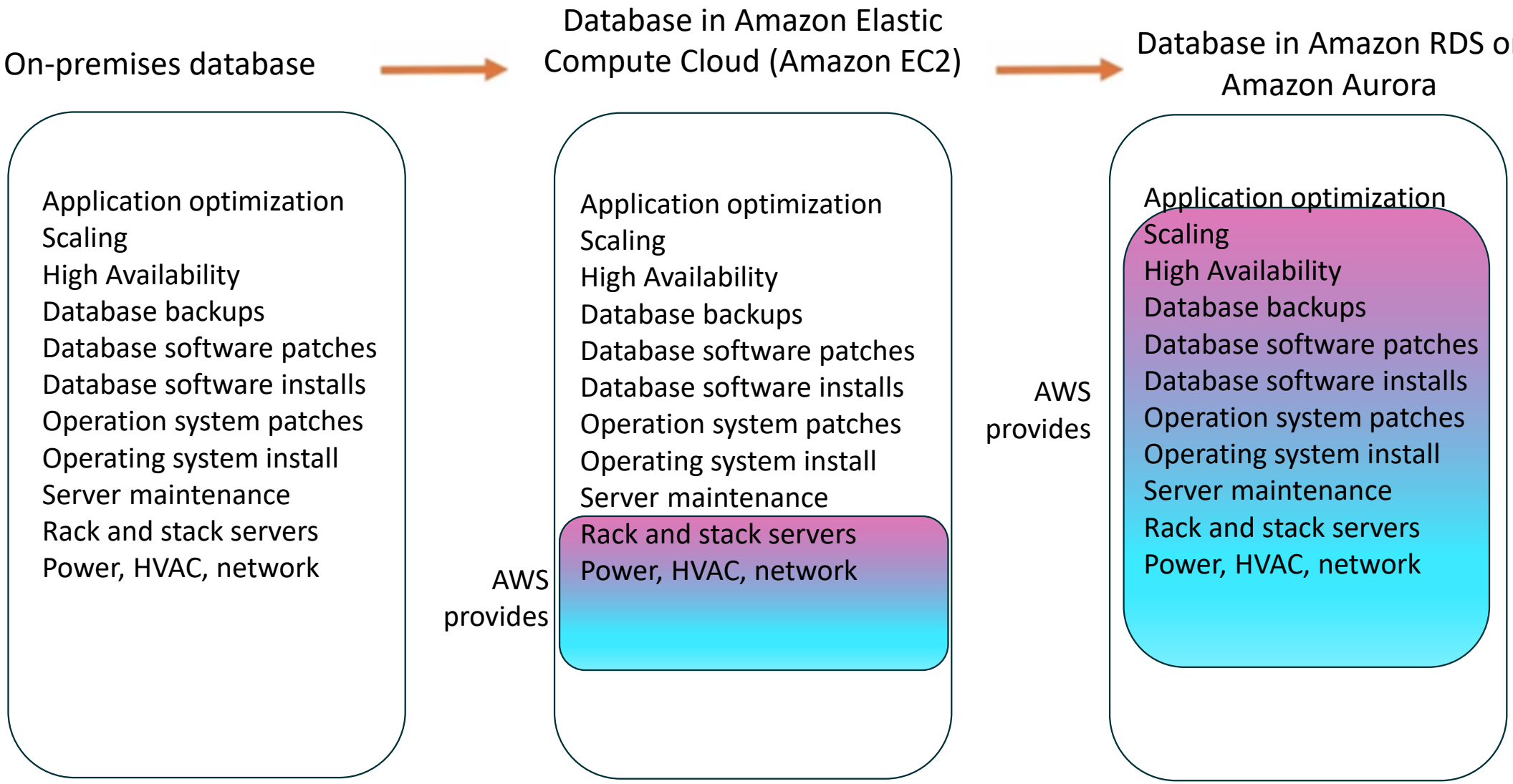
- Server maintenance and energy footprint
- Software installation and patches
- Database backups and high availability
- Limits on scalability
- Data security
- Operating system (OS) installation and patches

Amazon RDS

Managed service that sets up and operates a relational database in the cloud.



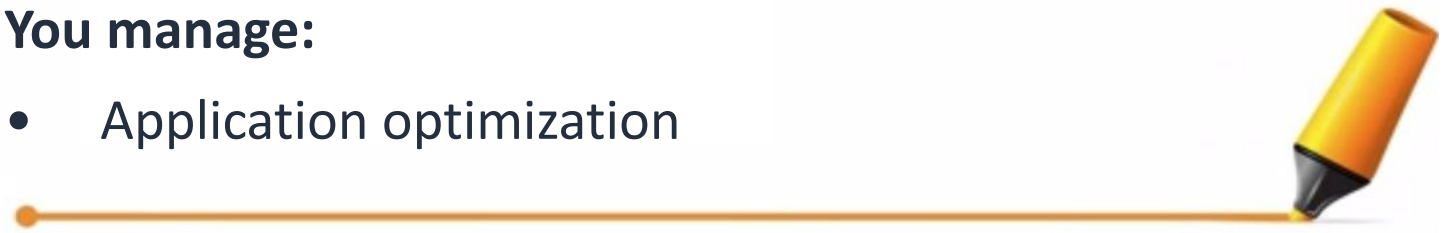
From on-premises databases to Amazon RDS



Managed services responsibilities

You manage:

- Application optimization



AWS manages:

- OS installation and patches
- Database software installation and patches
- Database backups
- High availability
- Scaling
- Power and racking and stacking servers
- Server maintenance

Amazon RDS

Amazon RDS DB instances

Amazon RDS

Amazon RDS DB main instance

DB Instance Class

- CPU
- Memory
- Network performance

DB Instance Storage

- Magnetic
- General Purpose (solid state drive, or SSD)
- Provisioned IOPS

MySQL

Amazon Aurora

Microsoft SQL Server

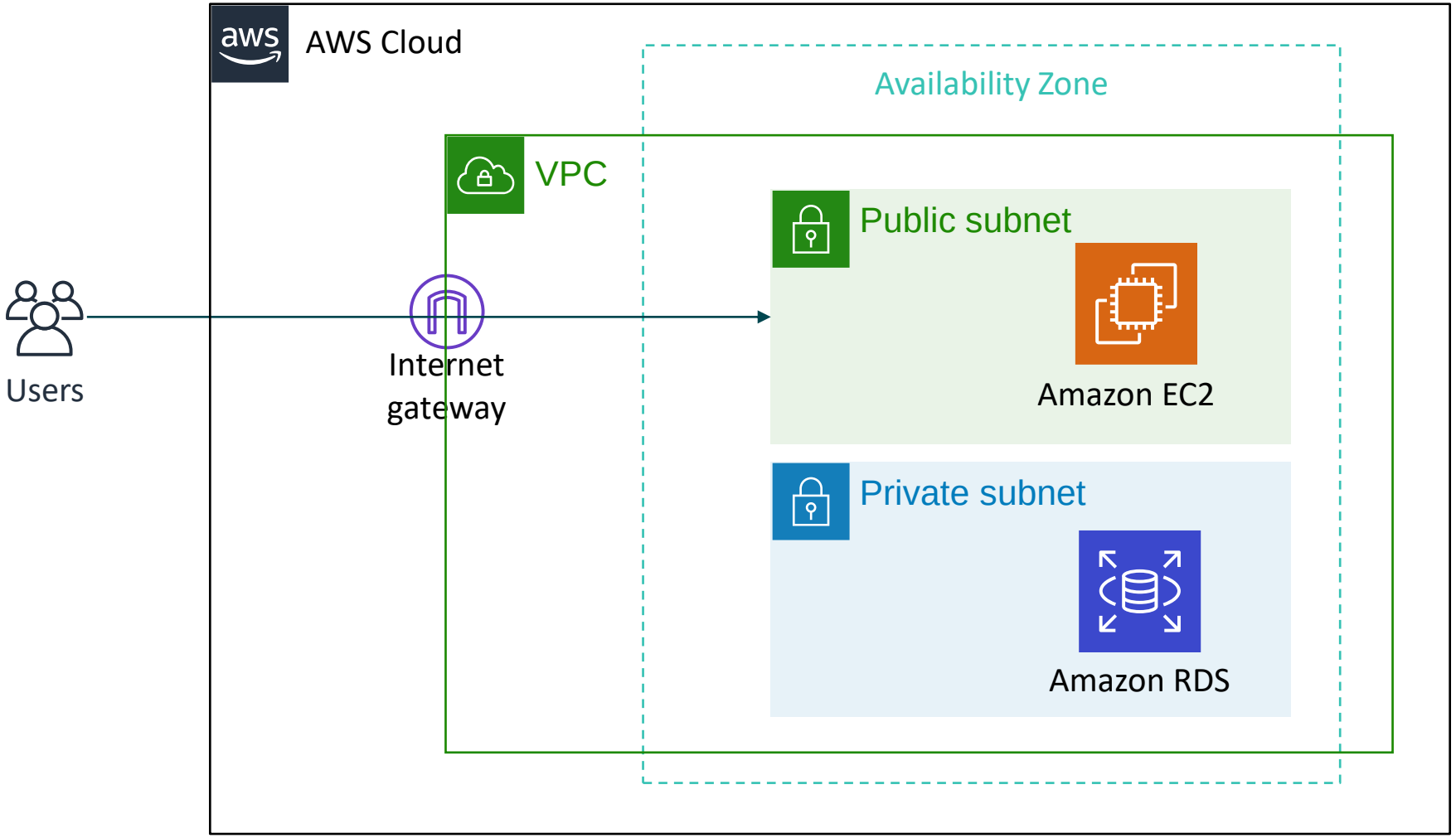
PostgreSQL

MariaDB

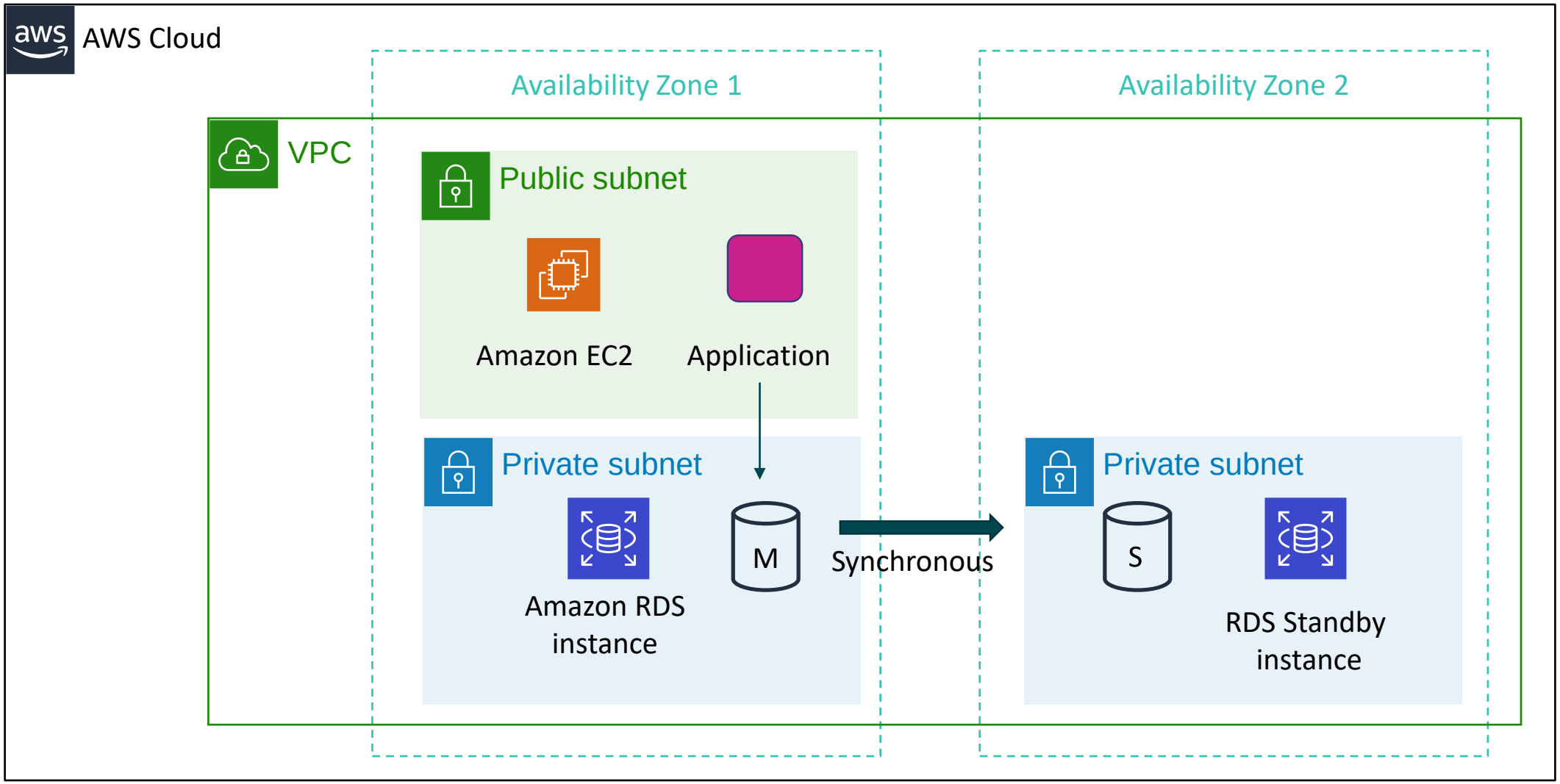
Oracle

DB engines

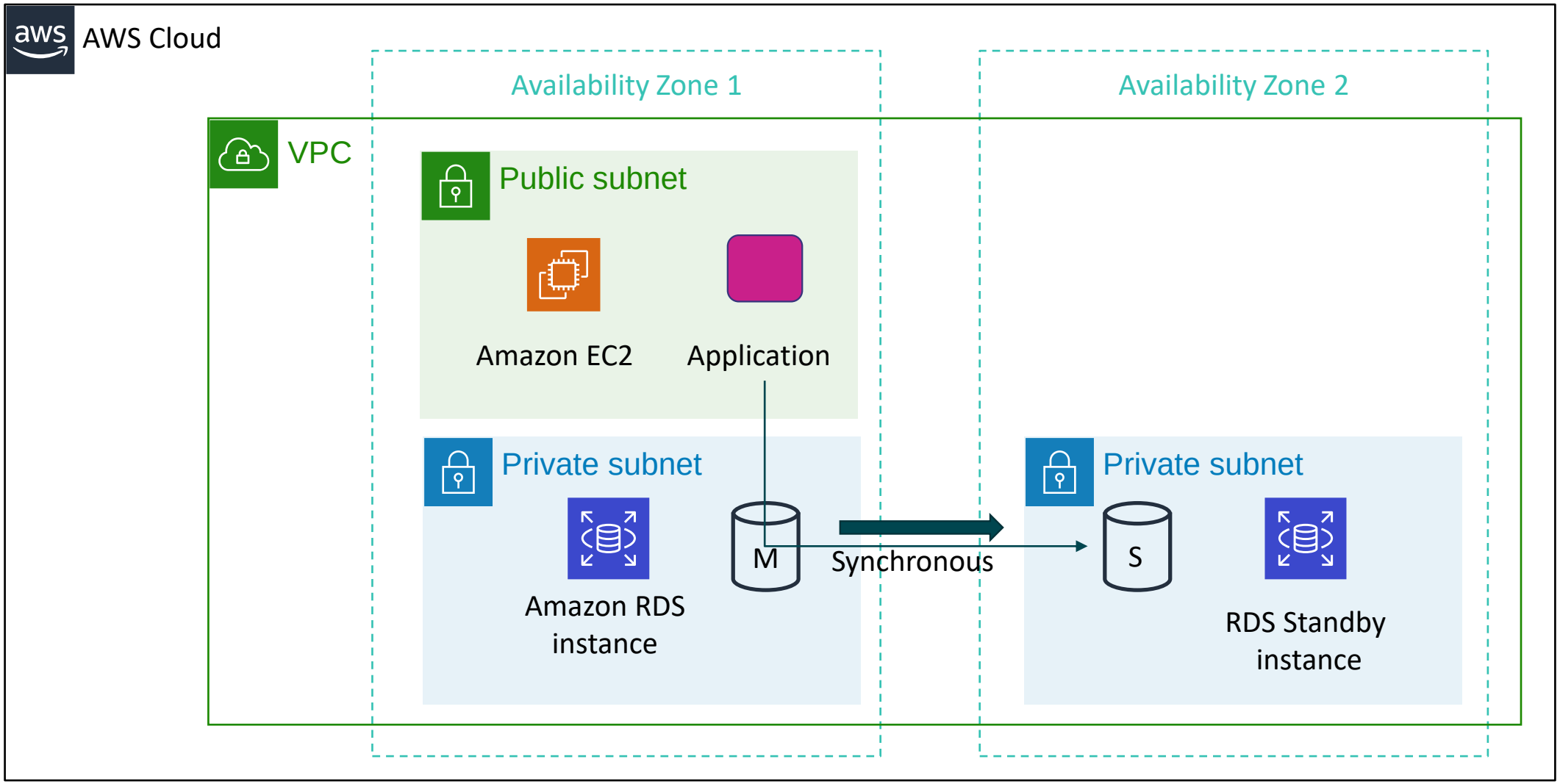
Amazon RDS in a virtual private cloud (VPC)



High availability with Multi-AZ deployment (1 of 2)



High availability with Multi-AZ deployment (2 of 2)



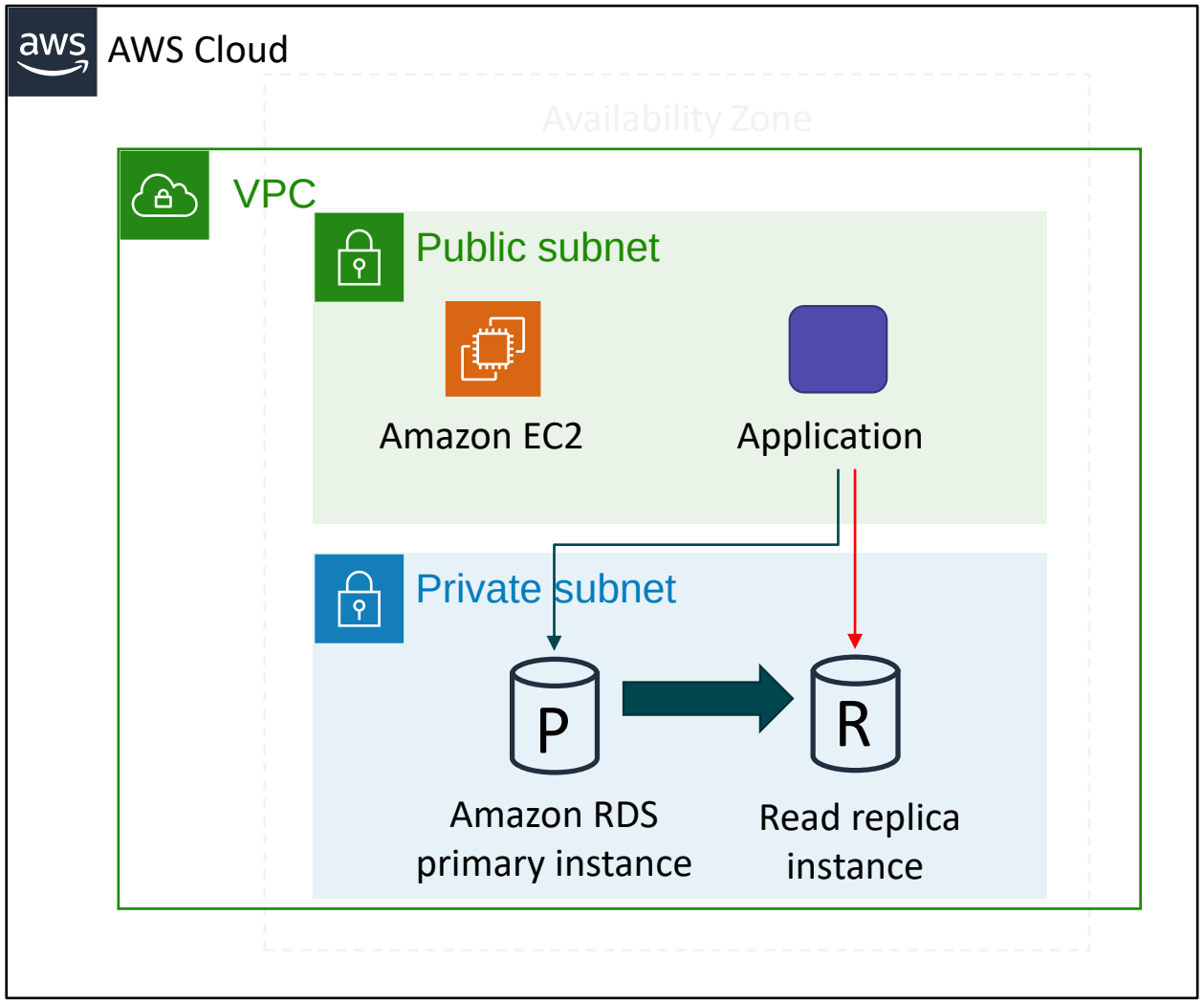
Amazon RDS read replicas

Features

- Offers asynchronous replication
- Can be promoted to primary if needed

Functionality

- Use for read-heavy database workloads
- Offload read queries



Use cases

Web and mobile applications	✓ High throughput ✓ Massive storage scalability ✓ High availability
Ecommerce applications	✓ Low-cost database ✓ Data security ✓ Fully managed solution
Mobile and online games	✓ Rapidly grow capacity ✓ Automatic scaling ✓ Database monitoring

When to Use Amazon RDS

Use Amazon RDS when your application requires:

- Complex transactions or complex queries
- A medium to high query or write rate – Up to 30,000 IOPS (15,000 reads + 15,000 writes)
- No more than a single worker node or shard
- High durability

- **Do not use Amazon RDS when your application requires:**
  - Massive read/write rates (for example, 150,000 write/second)
  - Sharding due to high data size or throughput demands
  - Simple GET or PUT requests and queries that a NoSQL database can handle
  - Relational database management system (RDBMS) customization

Amazon RDS: Clock-hour billing and database characteristics

Clock-hour billing –

- Resources incur charges when running

Database characteristics –

- Physical capacity of database:
  - Engine
  - Size
  - Memory class

Amazon RDS: DB purchase type and multiple DB instances

DB purchase type –

- On-Demand Instances
  - Compute capacity by the hour
- Reserved Instances
  - Low, one-time, upfront payment for database instances that are reserved with a 1-year or 3-year term

Number of DB instances –

- Provision multiple DB instances to handle peak loads

Amazon RDS: Storage

Provisioned storage –

- No charge
  - Backup storage of up to 100 percent of database storage for an active database
- Charge (*GB/month*)
  - Backup storage for terminated DB instances

Additional storage –

- Charge (*GB/month*)
  - Backup storage in addition to provisioned storage

Amazon RDS: Deployment type and data transfer

Requests –

- The number of input and output requests that are made to the database

Deployment type—Storage and I/0 charges vary, depending on whether you deploy to –

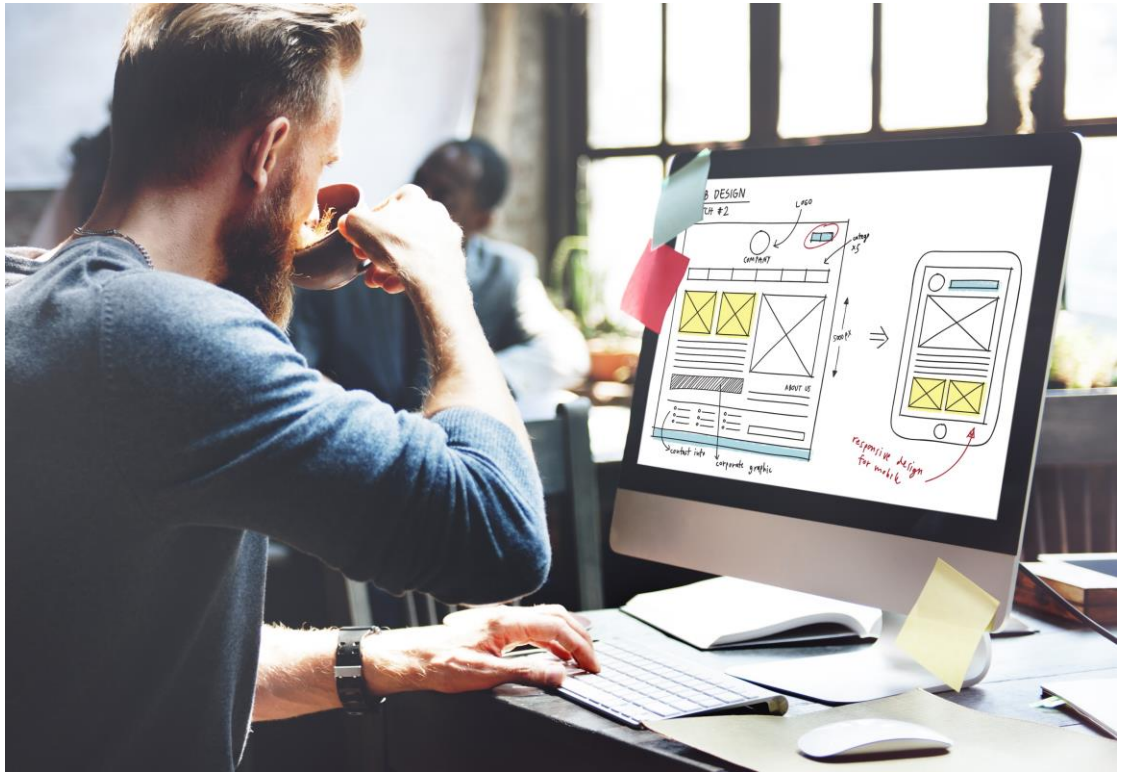
- Single Availability Zone
- Multiple Availability Zones

Data transfer –

- No charge for inbound data transfer
- Tiered charges for outbound data transfer

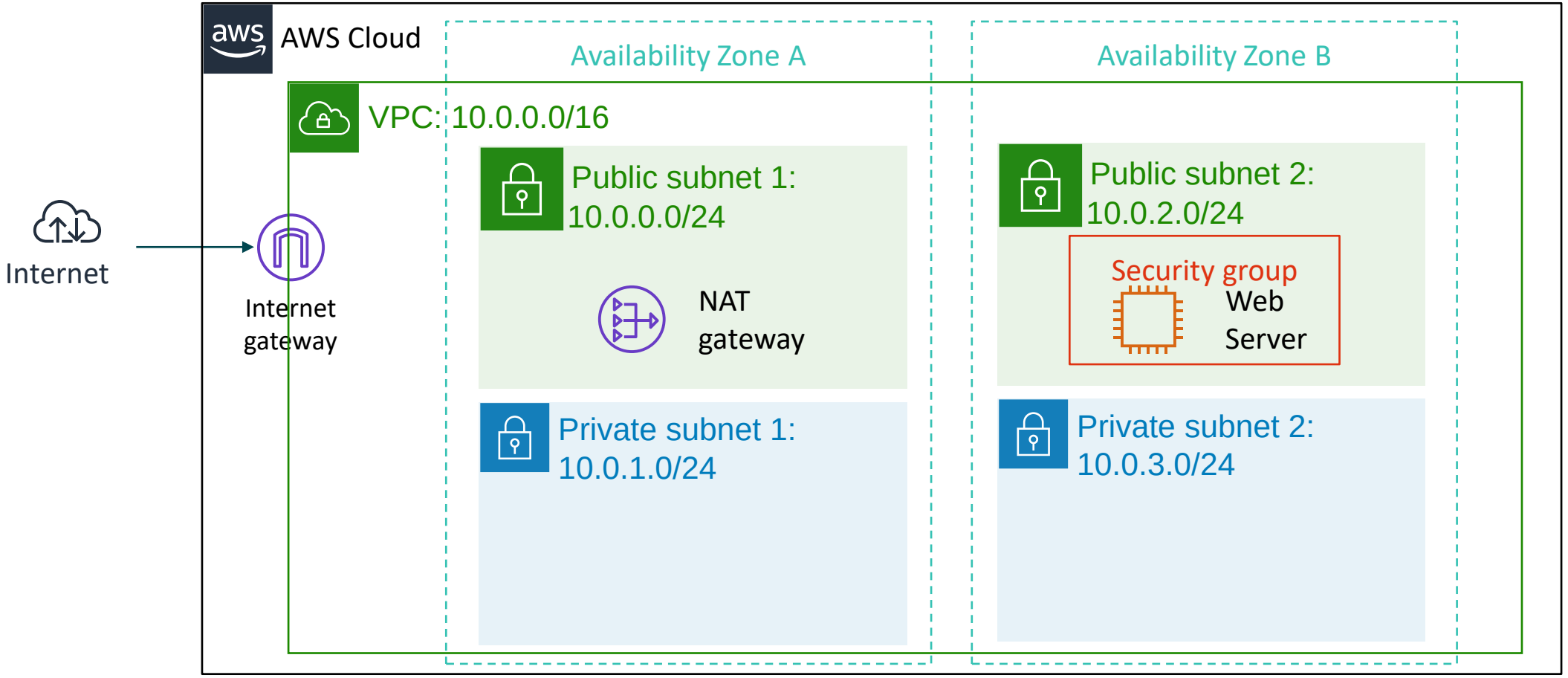
Recorded demo: Amazon RDS console

Build Your DB Server and Interact with Your DB Using an App

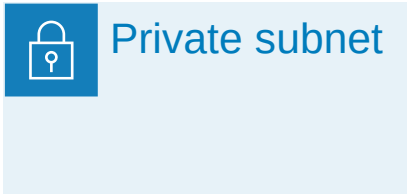


Lab 5: Scenario

This lab is designed to show you how to use an AWS managed database instance to solve a need for a relational database.



Lab 5: Tasks

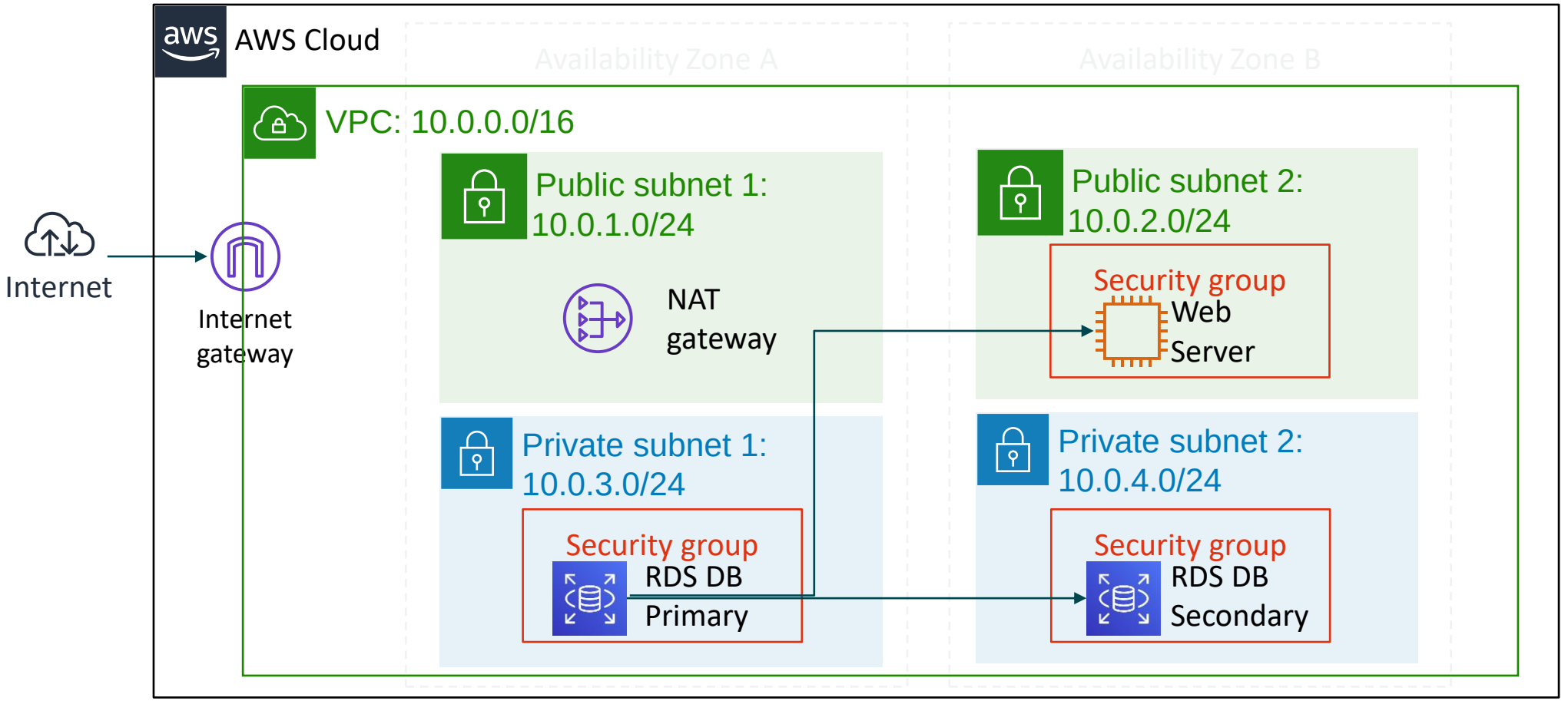


Create a **VPC security group.**

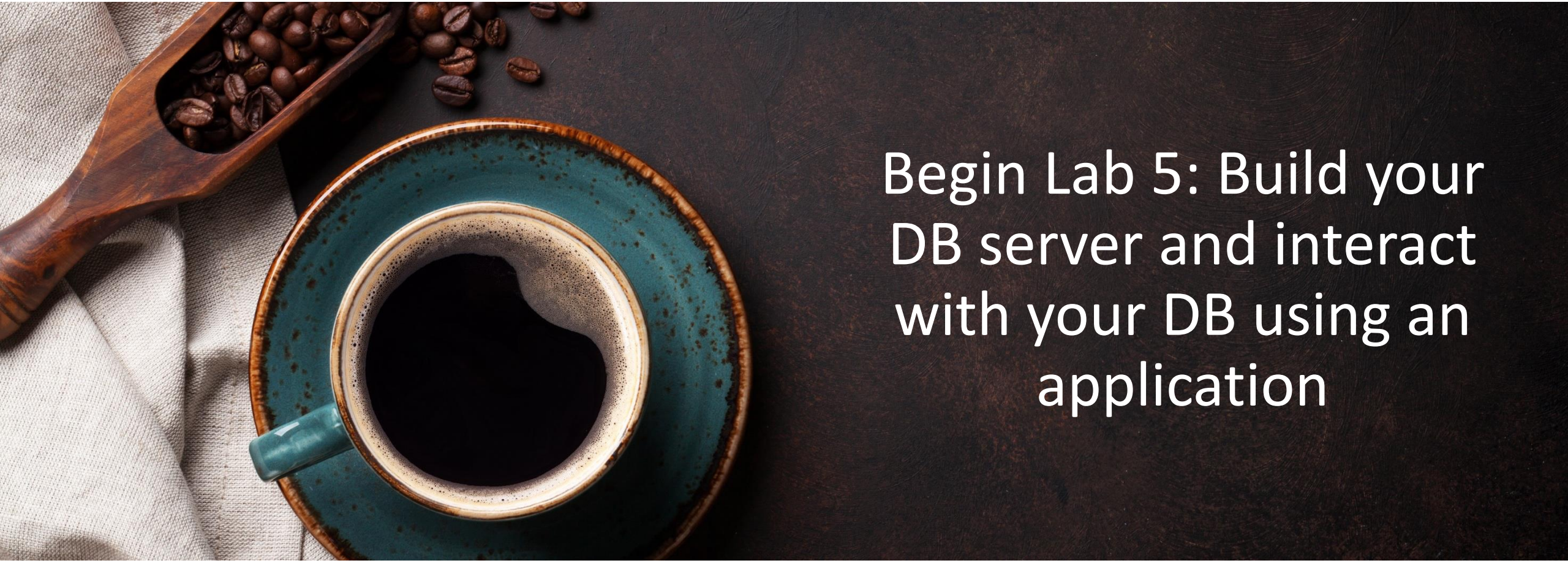
Create a **DB subnet group.**

Create an **Amazon RDS DB** instance and interact with your database.

Lab 5: Final product



~ 30 minutes



Begin Lab 5: Build your DB server and interact with your DB using an application

Lab debrief: key takeaways

Section 1 key takeaways

- With Amazon RDS, you can set up, operate, and scale relational databases in the cloud.
- Features –
  - Managed service
  - Accessible via the console, AWS Command Line Interface (AWS CLI), or application programming interface (API) calls
  - Scalable (compute and storage)
  - Automated redundancy and backup are available
  - Supported database engines:
    - Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle, Microsoft SQL Server

Section 2: Amazon DynamoDB

Module 8: Databases

Relational versus non-relational databases

<table>
<tr><th></th><th>Relational (SQL)</th><th>Non-Relational</th></tr>
<tr><td>Data Storage</td><td>Rows and columns</td><td>Key-value, document, graph</td></tr>
<tr><td>Schemas</td><td>Fixed</td><td>Dynamic</td></tr>
<tr><td>Querying</td><td>Uses SQL</td><td>Focuses on collection of documents</td></tr>
<tr><td>Scalability</td><td>Vertical</td><td>Horizontal</td></tr>
<tr><td>Example</td><td>
<table>
<tr><th>ISBN</th><th>Title</th><th>Author</th><th>Format</th></tr>
<tr><td>3111111223439</td><td>Withering Depths</td><td>Jackson, Mateo</td><td>Paperback</td></tr>
<tr><td>3122222223439</td><td>Wily Willy</td><td>Wang, Xiulan</td><td>Ebook</td></tr>
</table>
</td><td>{
ISBN: 3111111223439,
Title: “Withering Depths”,
Author: ”Jackson, Mateo”,
Format: “Paperback”
}</td></tr>
</table>

What is Amazon DynamoDB?

Fast and flexible NoSQL database service for any scale

Amazon DynamoDB

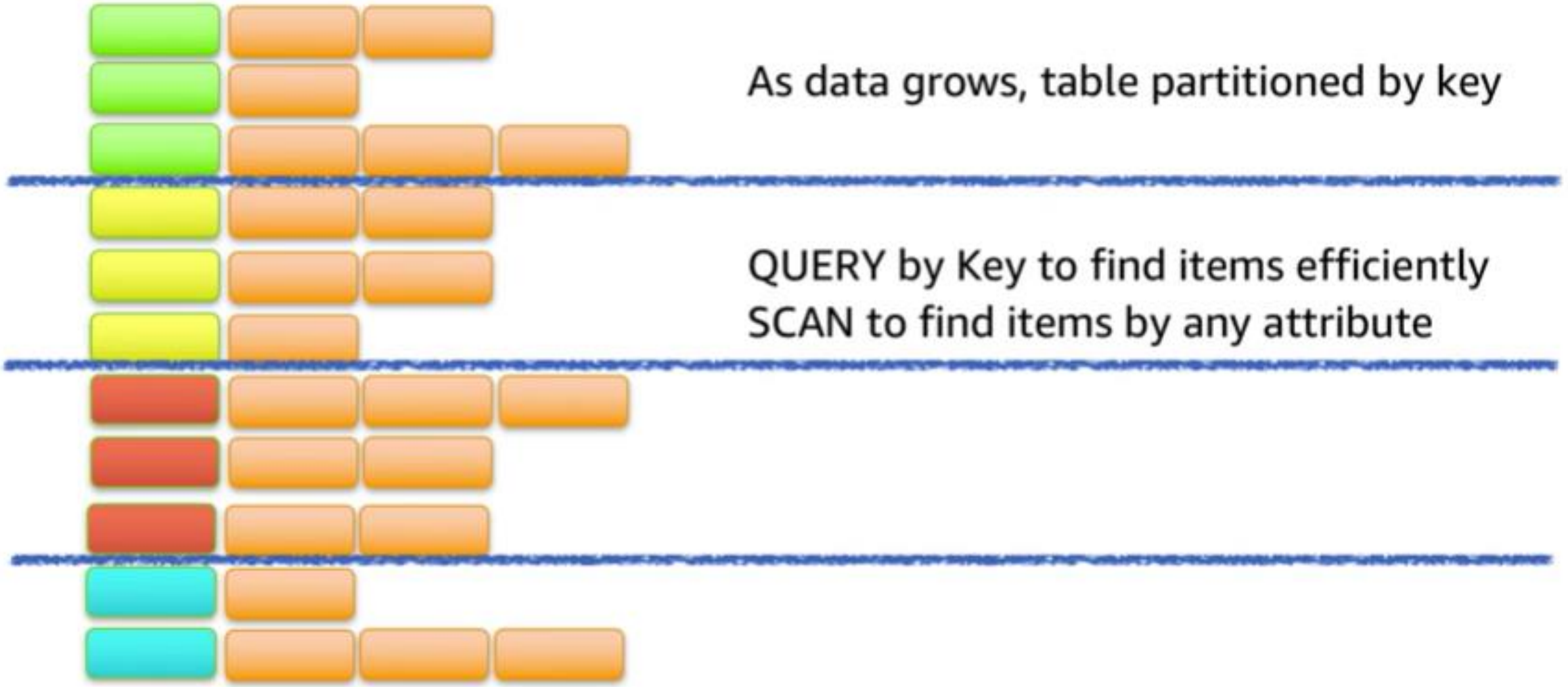
- NoSQL database tables
- Virtually unlimited storage
- Items can have differing attributes
- Low-latency queries
- Scalable read/write throughput

Amazon DynamoDB core components

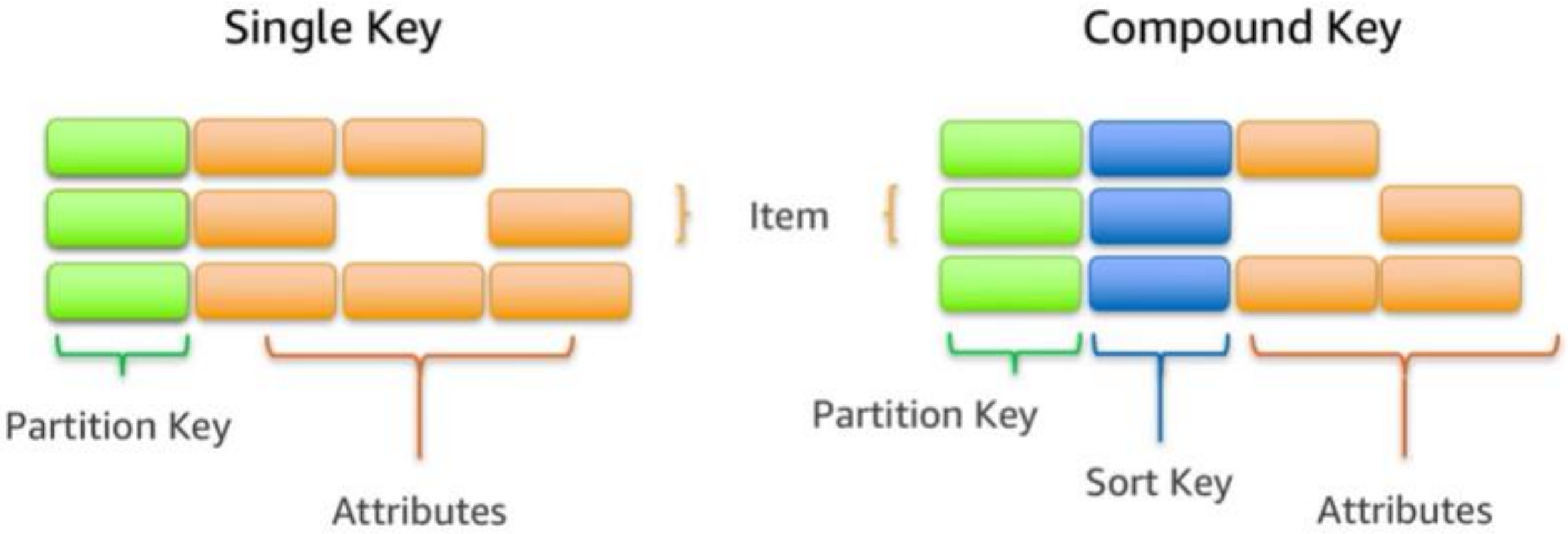
- Tables, items, and attributes are the core DynamoDB components

- DynamoDB supports two different kinds of primary keys: Partition key and partition and sort key

Partitioning



Items in a table must have a key



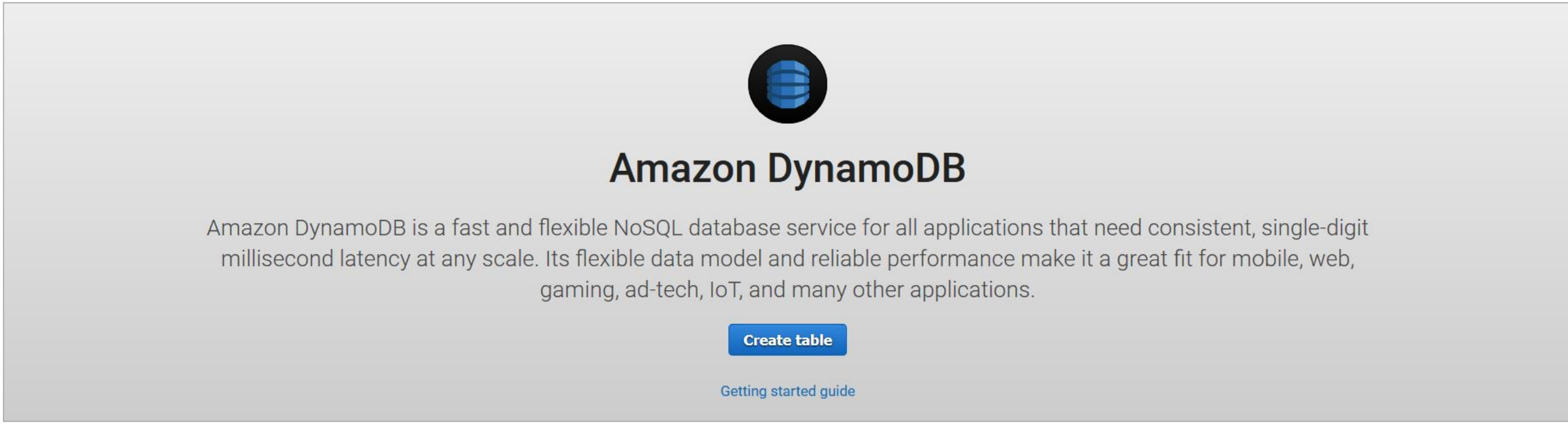
Section 2 key takeaways

Amazon DynamoDB:

- Runs exclusively on SSDs.
- Supports document and key-value store models.
- Replicates your tables automatically across your choice of AWS Regions.
- Works well for mobile, web, gaming, adtech, and Internet of Things (IoT) applications.
- Is accessible via the console, the AWS CLI, and API calls.
- Provides consistent, single-digit millisecond latency at any scale.
- Has no limits on table size or throughput.

Recorded demo: Amazon DynamoDB console

Amazon DynamoDB demonstration



Create tables

Add and query items

Monitor and manage tables

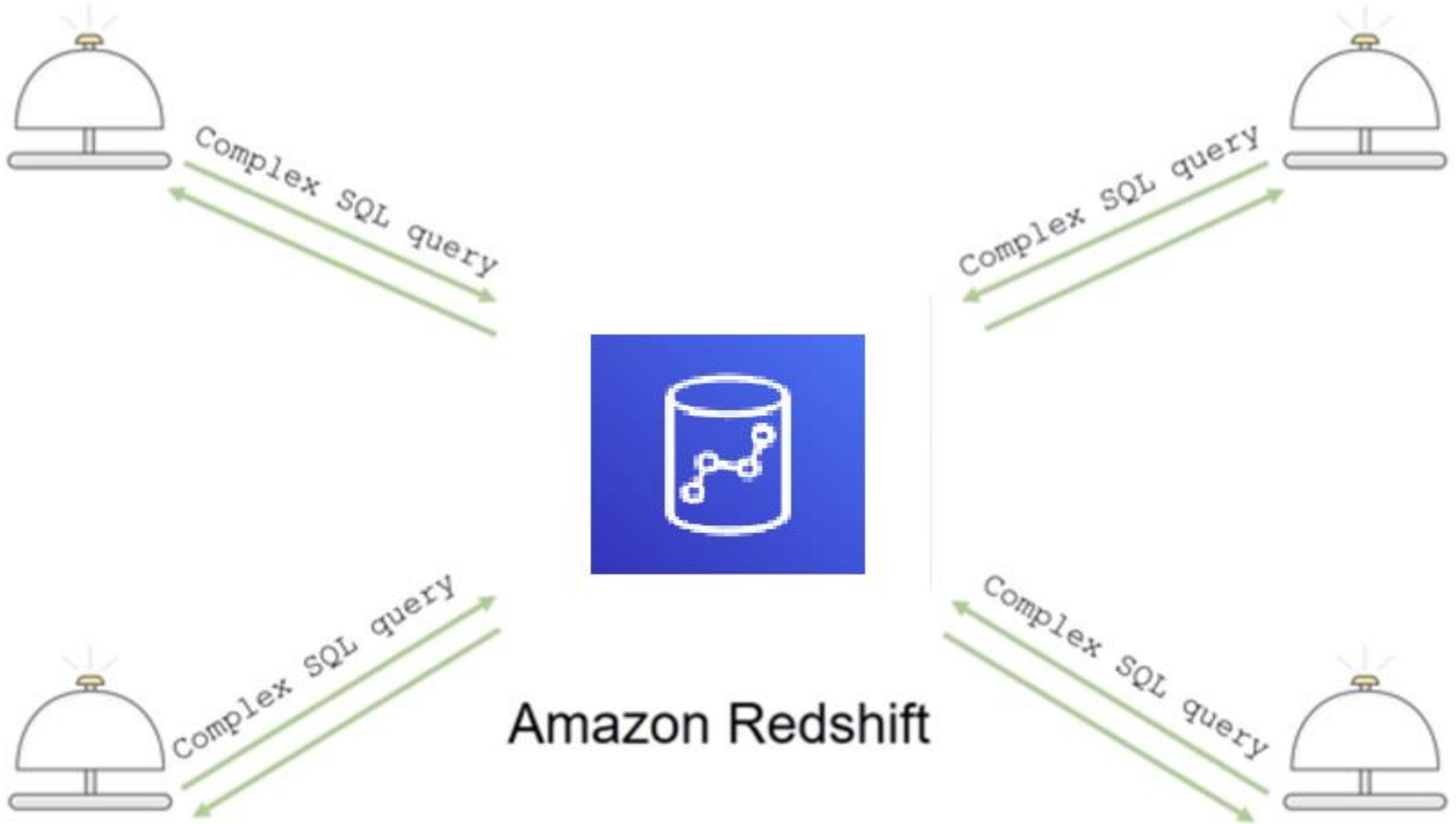
Section 3: Amazon Redshift

Module 8: Databases

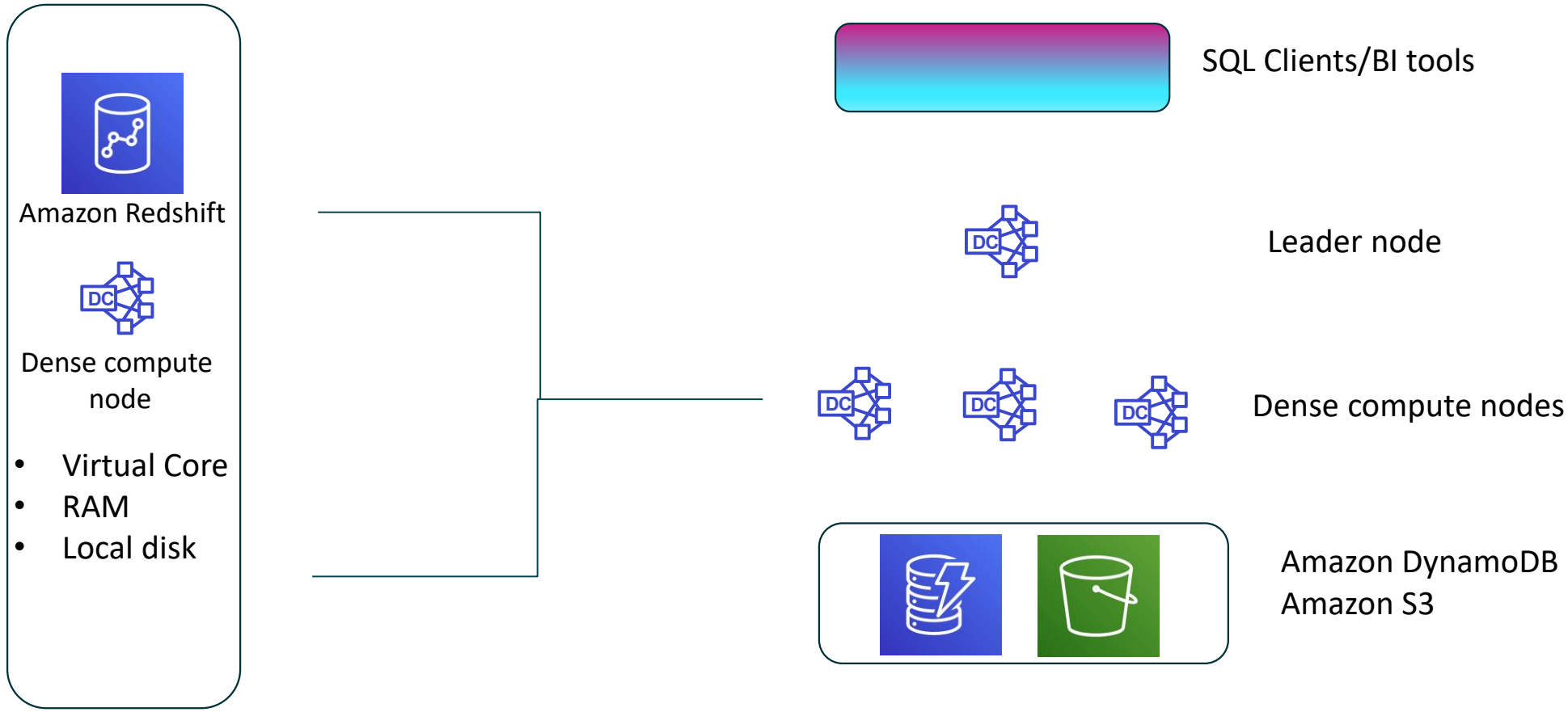
Amazon Redshift

Amazon Redshift

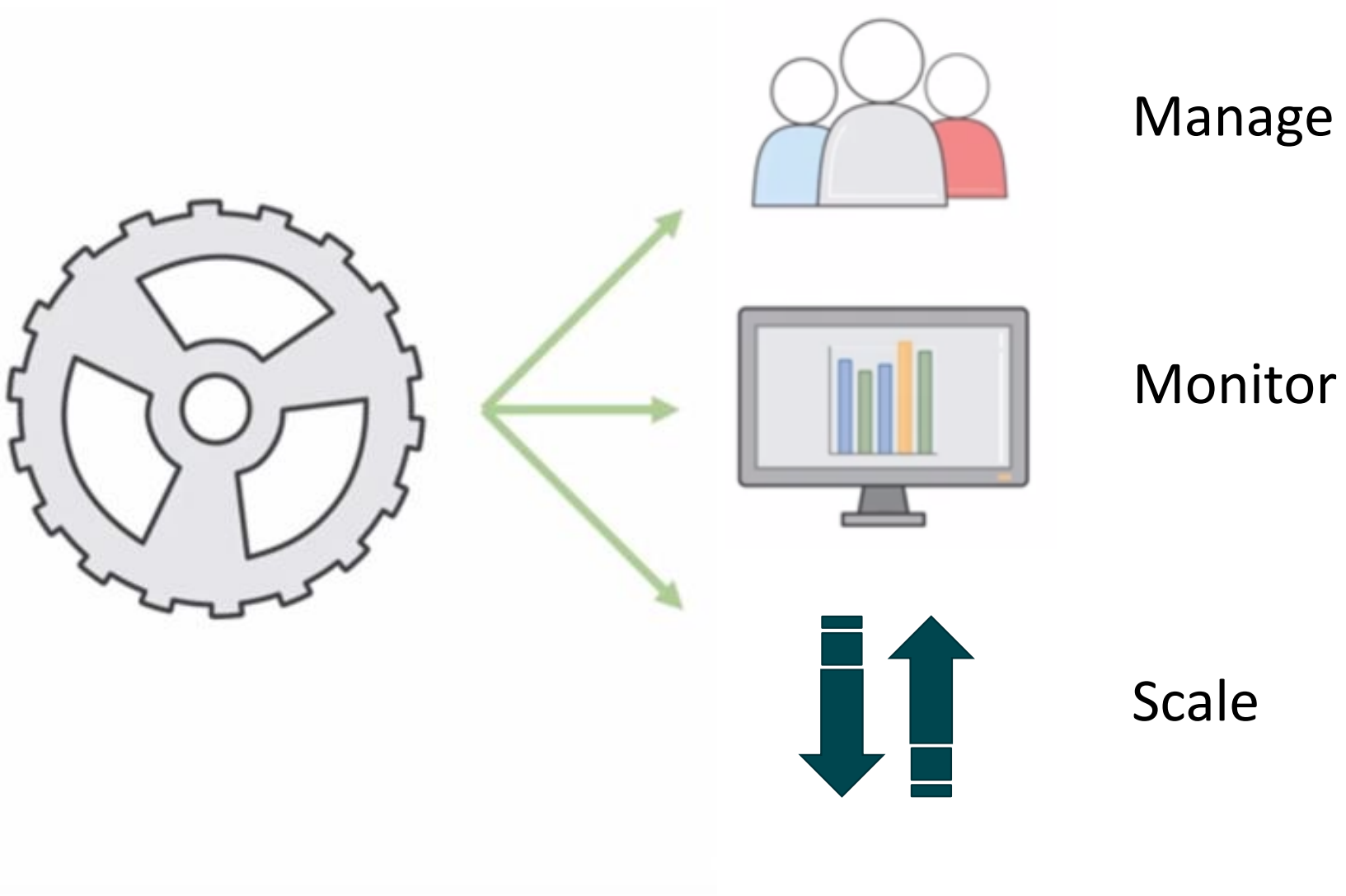
Introduction to Amazon Redshift



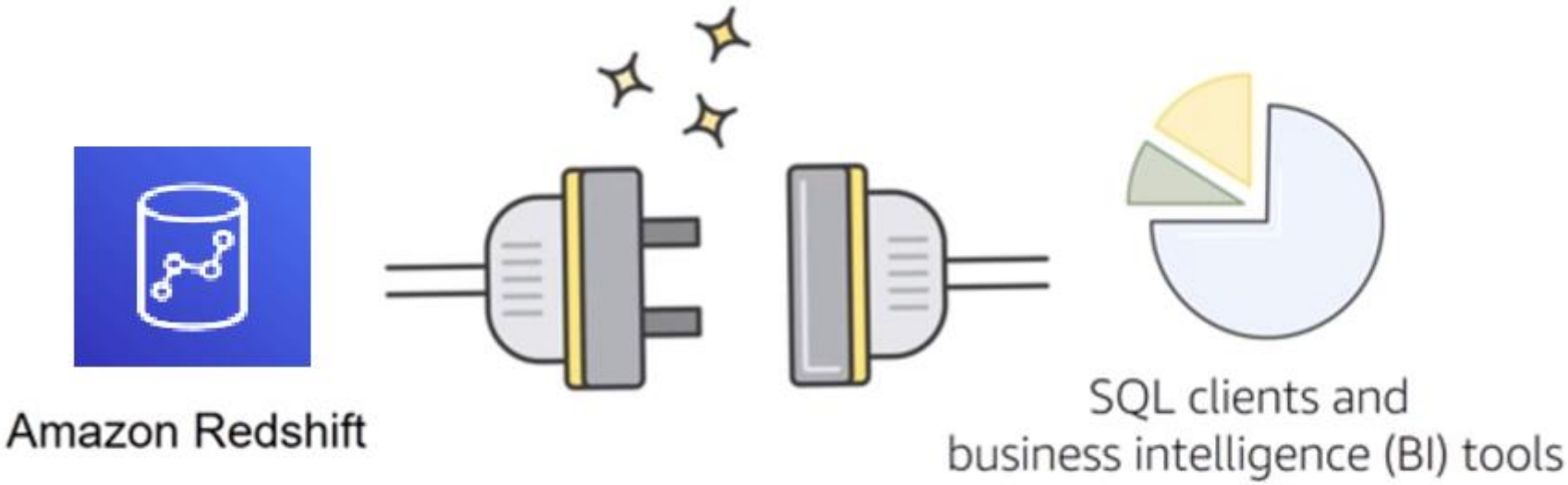
Parallel processing architecture



Automation and scaling

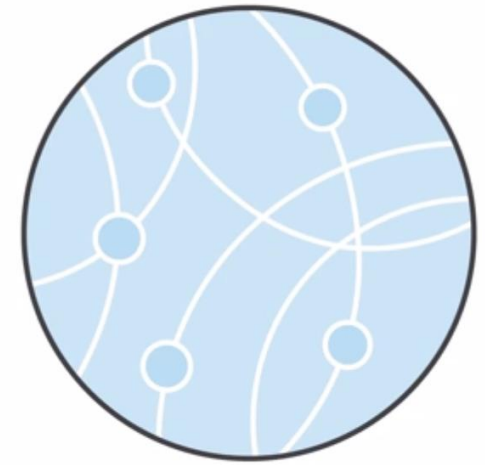


Compatibility



Amazon Redshift use cases (1 of 2)

- Enterprise data warehouse (EDW)
  - Migrate at a pace that customers are comfortable with
  - Experiment without large upfront cost or commitment
  - Respond faster to business needs
- Big data
  - Low price point for small customers
  - Managed service for ease of deployment and maintenance
  - Focus more on data and less on database management



Amazon Redshift use cases (2 of 2)

- Software as a service (SaaS)
  - Scale the data warehouse capacity as demand grows
  - Add analytic functionality to applications
  - Reduce hardware and software costs

Section 3 key takeaways

Amazon Redshift features:

- Fast, fully managed data warehouse service
- Easily scale with no downtime
- Columnar storage and parallel processing architectures
- Automatically and continuously monitors cluster
- Encryption is built in

Section 4: Amazon Aurora

Module 8: Databases

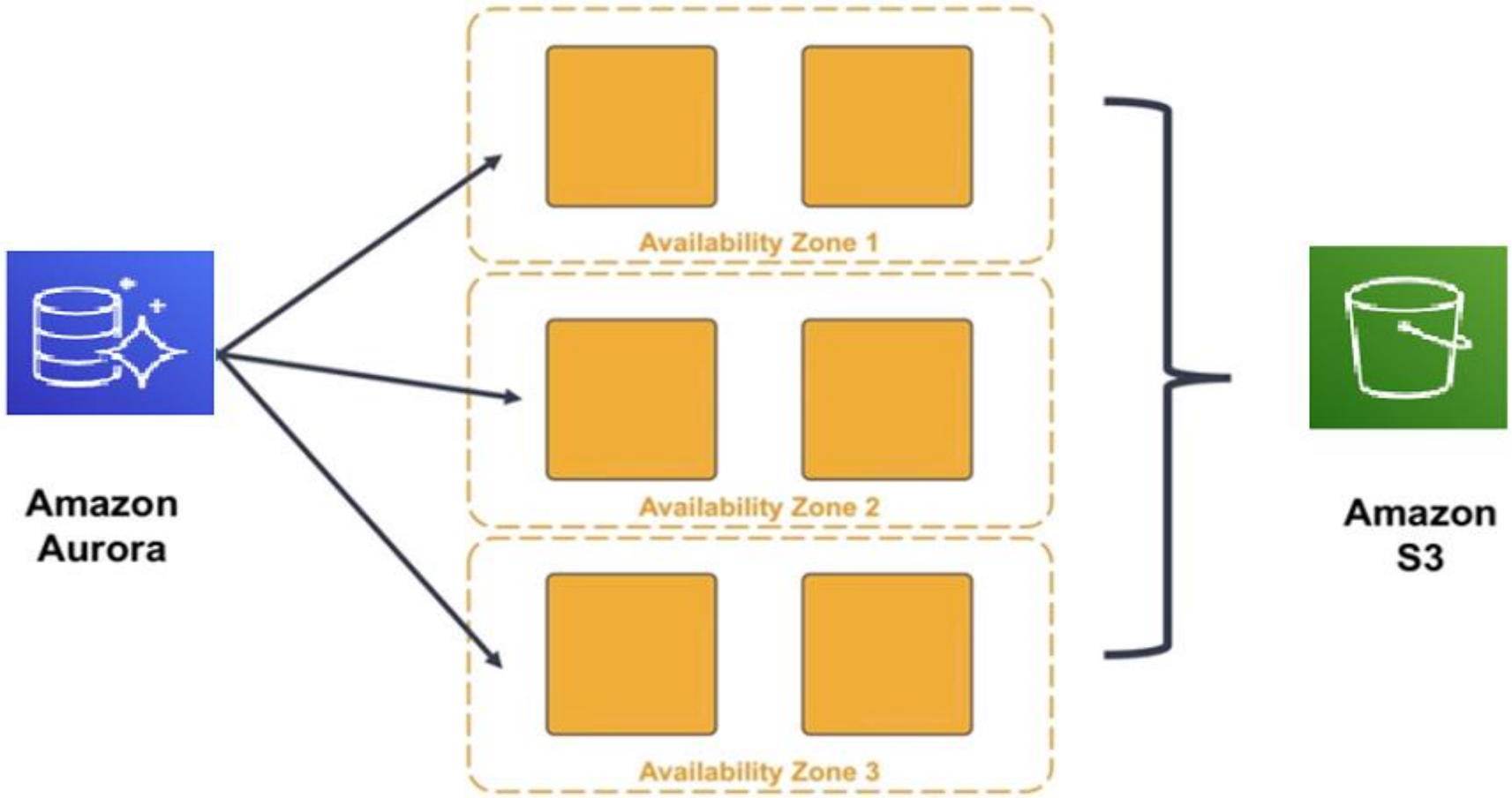
Amazon Aurora

Amazon Aurora

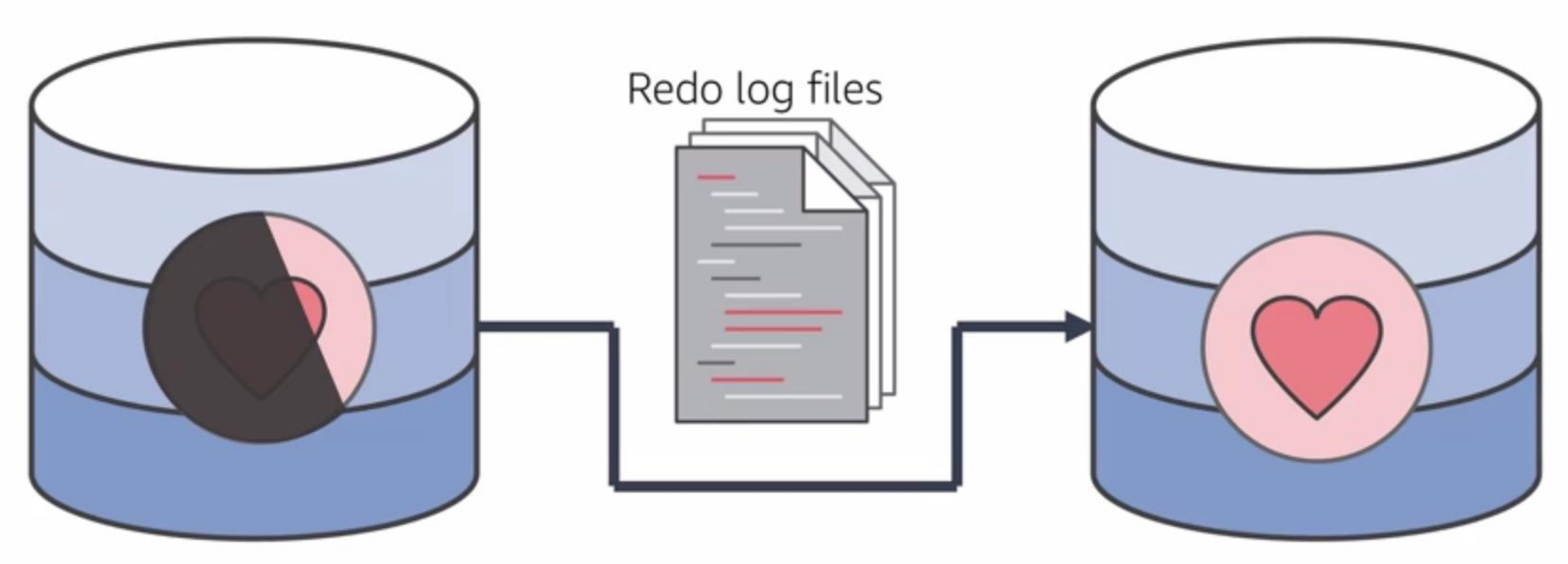
- Enterprise-class relational database
- Compatible with MySQL or PostgreSQL
- Automate time-consuming tasks (such as provisioning, patching, backup, recovery, failure detection, and repair).

Amazon Aurora service benefits

High availability



Resilient design



Section 4 key takeaways

Amazon Aurora features:

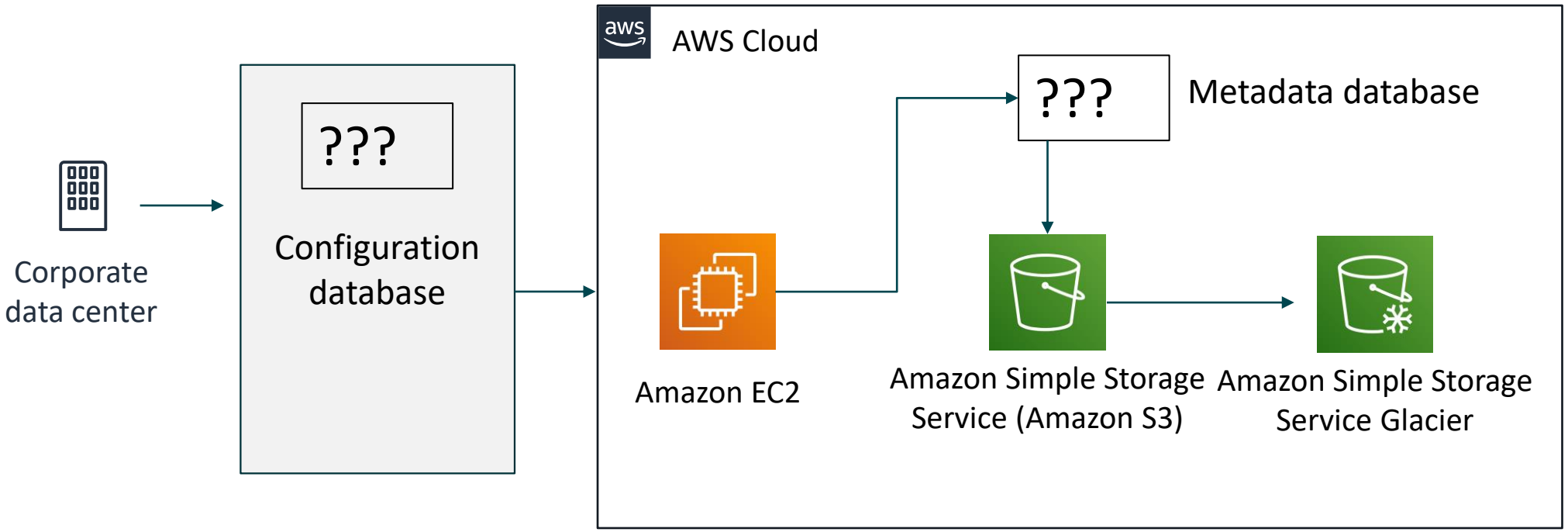
- High performance and scalability
- High availability and durability
- Multiple levels of security
- Compatible with MySQL and PostgreSQL
- Fully managed

The right tool for the right job

What are my requirements?	
Enterprise-class relational database	Amazon RDS
Fast and flexible NoSQL database service for any scale	Amazon DynamoDB
Operating system access or application features that are not supported by AWS database services	Databases on Amazon EC2
Specific case-driven requirements (machine learning, data warehouse, graphs)	AWS purpose-built database services

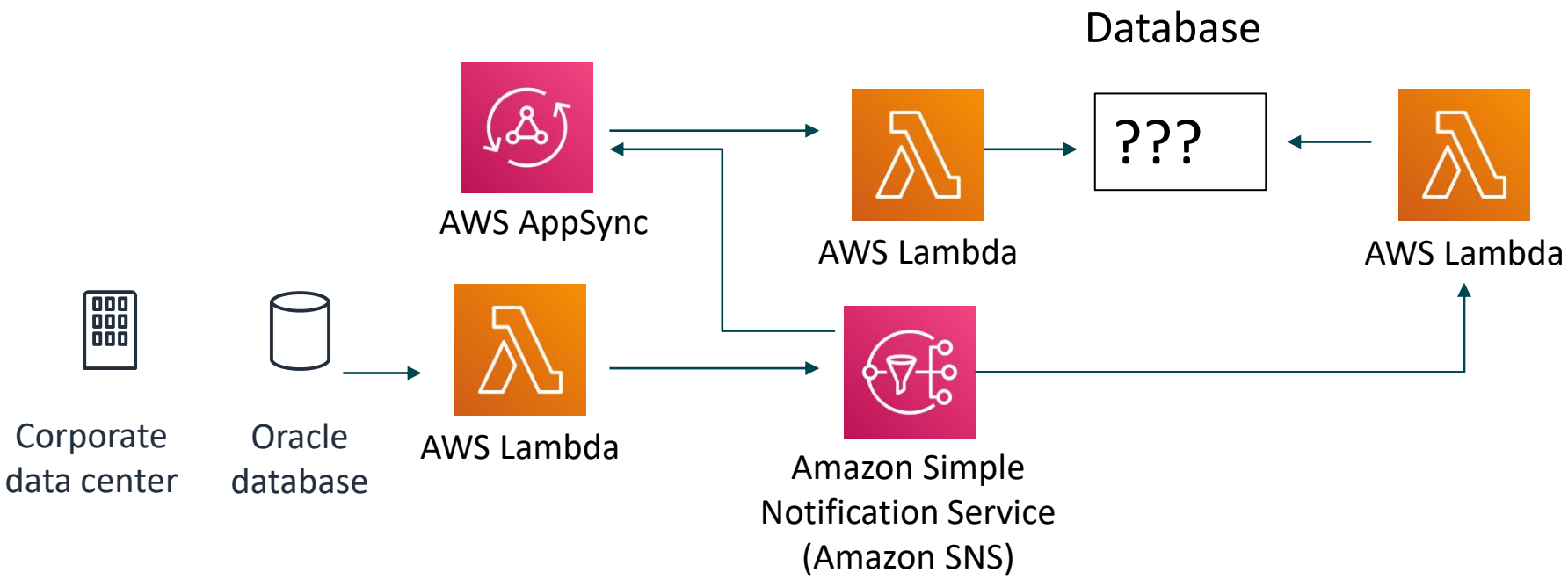
Database case study activity (1 of 3)

Case 1: A data protection and management company that provides services to enterprises. They must provide database services for over 55 petabytes of data. They have two types of data that require a database solution. First, they need a relational database store for configuration data. Second, they need a store for unstructured metadata to support a de-duplication service. After the data is de-duplicated, it is stored in Amazon S3 for quick retrieval, and eventually moved to Amazon S3 Glacier for long-term storage. The following diagram illustrates their architecture.



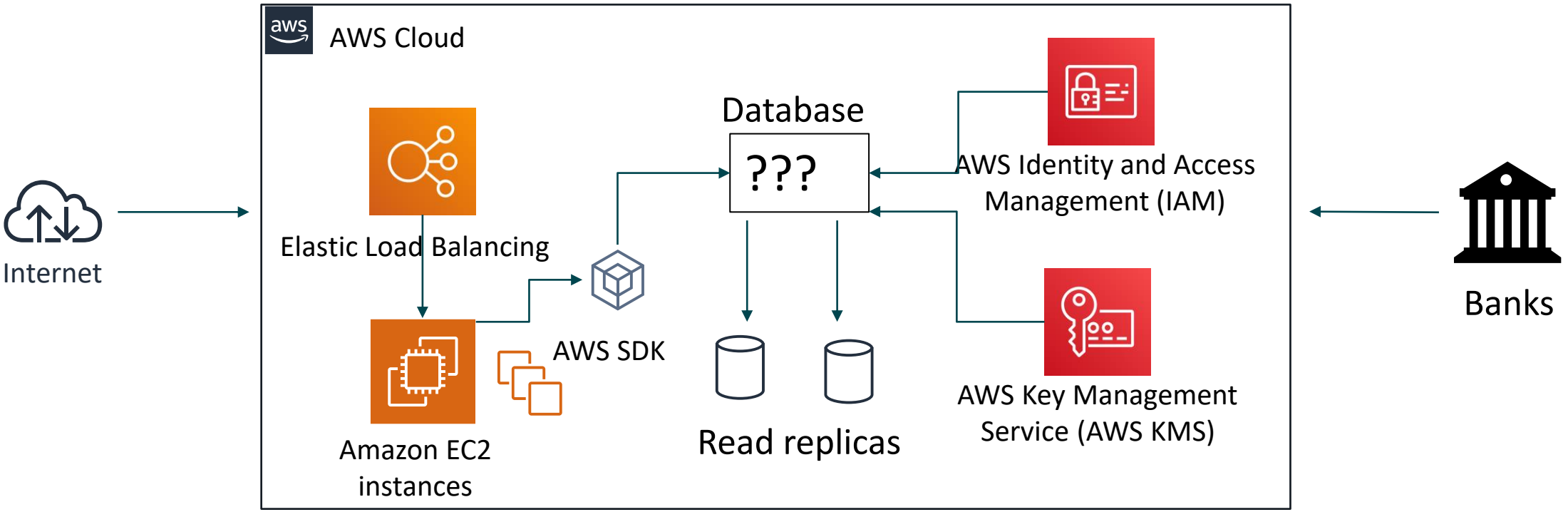
Database case study activity (2 of 3)

Case 2: A commercial shipping company that uses an on-premises legacy data management system. They must migrate to a serverless ecosystem while they continue to use their existing database system, which is based on Oracle. They are also in the process of decomposing their highly structured relational data into semistructured data. The following diagram illustrates their architecture.



Database case study activity 3

Case 3: An online payment processing company that processes over1 million transactions per day. They must provide services to ecommerce customers who offer flash sales (sales that offer greatly reduced prices for a limited time), where demand can increase by 30 times in a short time period. They use IAM and AWS KMS to authenticate transactions with financial institutions. They need high throughput for these peak loads. The following diagram illustrates their architecture.



Module wrap-up

Module 8: Databases

Module summary

In summary, in this module, you learned how to:

- Explain Amazon Relational Database Service (Amazon RDS)
- Identify the functionality in Amazon RDS
- Explain Amazon DynamoDB
- Identify the functionality in Amazon DynamoDB
- Explain Amazon Redshift
- Explain Amazon Aurora
- Perform tasks in an RDS database, such as launching, configuring, and interacting

Complete the knowledge check

Sample exam question

Which of the following is a fully-managed NoSQL database service?

Choice	Response
A	Amazon Relational Database Service (Amazon RDS)
B	Amazon DynamoDB
C	Amazon Aurora
D	Amazon Redshift

Sample exam question answer

Which of the following is a fully-managed NoSQL database service?

The correct answer is B.

The keywords in the question are “NoSQL database service”.

Additional resources

- AWS Database page: https://aws.amazon.com/products/databases/
- Amazon RDS page: https://aws.amazon.com/rds/
- Overview of Amazon database services: https://docs.aws.amazon.com/whitepapers/latest/aws-overview/database.html
- Getting started with AWS databases: https://aws.amazon.com/products/databases/learn/

Thank you

All trademarks are the property of their owners.