Lecture 6: AWS SQL Database

Maharishi International University

Department of Computer Science

M.S. Thao Huy Vu

Maharishi International University - Fairfield, Iowa

Agenda

- **Database Services in AWS**
- **Amazon RDS (Relational Database Service)**
- **Amazon Aurora**
- **Best Practices and Considerations**

Database Services in AWS

- A managed service provides database solutions tailored to
  different needs – from relational databases to NoSQL, in-m
  and graph databases
- Flexibility
- Durability and Availability
- Cost-efficiency
- Security

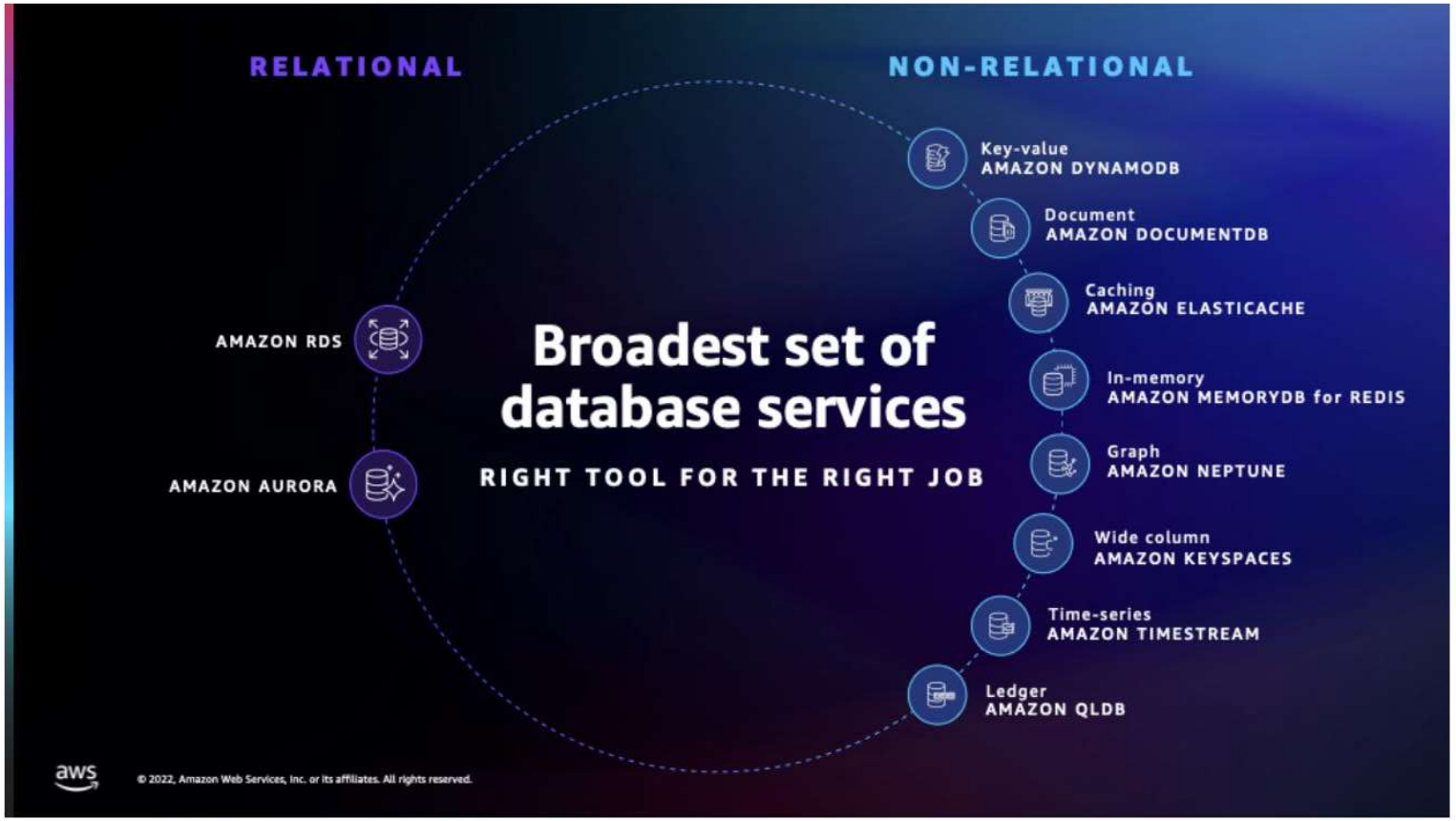
Database Solutions

Feature	AWS Database (Managed)	Self-Hosted Databas
Setup and Management	Fully managed by AWS, with automated setup, updates, and backups.	Requires manual insta configuration, and ma
Database Options	Offers managed SQL (e.g., RDS) and NoSQL (e.g., DynamoDB).	Supports SQL (e.g., M and NoSQL database
Scalability	Automatic scaling (vertical and horizontal) with minimal effort.	Scaling requires manu infrastructure manage
Availability	Built-in high availability (e.g., Multi-AZ for RDS).	Must configure replica manually.
Cost	Pay-as-you-go pricing; no upfront hardware costs.	Potentially lower hard operational overhead.
Control	Limited access to underlying infrastructure and database engine.	Full control over the O and configurations.

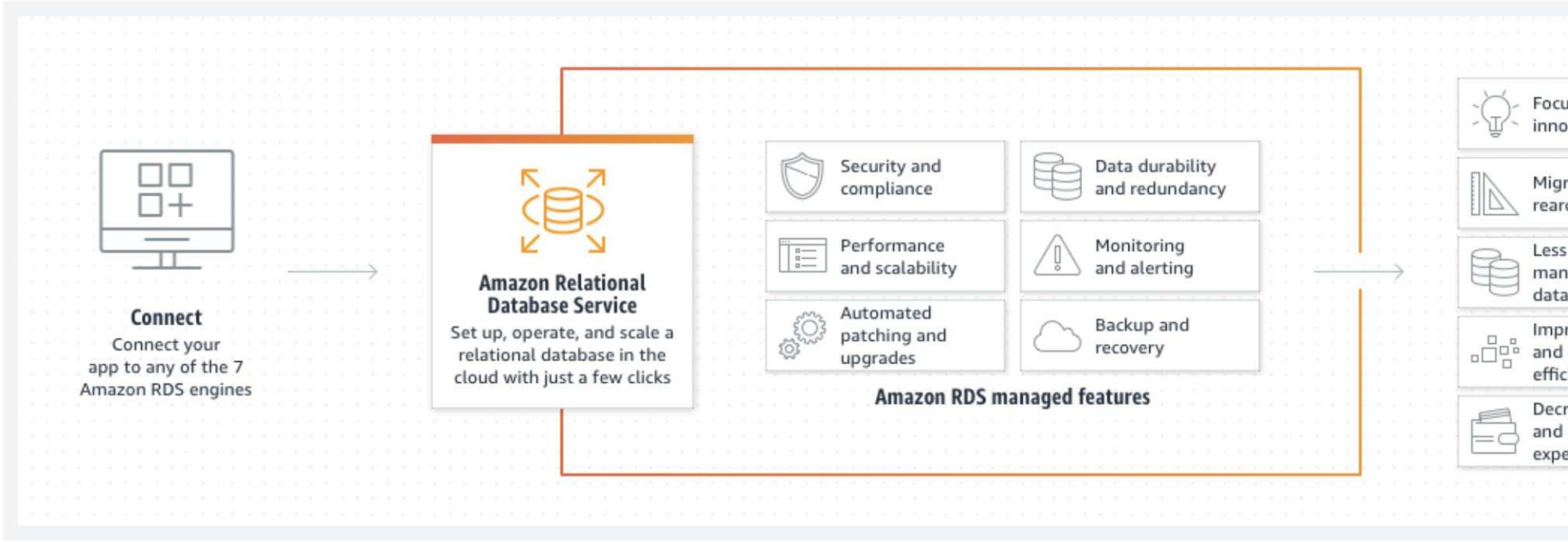
Database Services in AWS

- Rational Databases: Amazon RDS, Amazon Aurora
- NoSQL Databases: DynamoDB, DocumentDB
- In-memory Databases: Amazon ElasticCache, Redis, Mem
- Graph Databases: Amazon Neptune

Database Services in AWS



Amazon RDS


Connect
Connect your app to any of the 7 Amazon RDS engines
Amazon Relational Database Service
Set up, operate, and scale a relational database in the cloud with just a few clicks
Security and compliance
Data durability and redundancy
Performance and scalability
Monitoring and alerting
Automated patching and upgrades
Backup and recovery
Amazon RDS managed features

Amazon RDS

- **Database Engines**: Supports popular engines like **MySQL**, **PostgreSQL**, **MariaDB**, **SQL Server**, and **Aurora**.
- **Managed Service**: Automates database setup, patching, backups, and scaling.
- **Scalability**:
  - **Vertical scaling**: Easily resize compute and memory by changing instance type.
  - **Horizontal scaling**: Add **Read Replicas** to scale read-heavy workloads and improve perform
- **Availability and Reliability**:
- **Multi-AZ deployments**:
  - Provide high availability with **synchronous standby instances** in different Availability Zones
  - **Automated failover**: Ensures minimal downtime during outages or maintenance.
- **Performance**:
  - **Optimized SSD-backed storage** (gp3, io1) with support for **Provisioned IOPS**.
  - Designed for **high throughput** and **low latency** workloads.
  - Supports integration with **Amazon ElastiCache** for enhanced caching.

Amazon RDS

- **Security**
  - **Encryption** at rest (using KMS) and in transit (SSL/TLS).
  - **IAM integration** for fine-grained access control.
  - **Network isolation** via Amazon VPC.
  - **Supports compliance** with standards like **HIPAA**, **GDPR**, and **PCI DSS**.
- **Automated Backups with Point-in-Time Recovery (PITR)**
  - AWS automatically takes a **daily full snapshot** of your RDS database during yo preferred backup window.
  - **Transaction logs** are continuously backed up, enabling **restores to any speci second** within your retention period (up to **35 days**).
  - The entire process is **fully managed** by AWS, ensuring continuous protection w minimal overhead.

RTO & RPO in RDS

- **Recovery Point Objective (RPO)**
  - Maximum Acceptable **data loss**.
  - In RDS, RPO is **very low**, minimizing data loss.
- **Recovery Time Objective (RTO)**
  - Maximum acceptable **downtime**.
  - In RDS, RTO is typically **a few minutes** during failover.

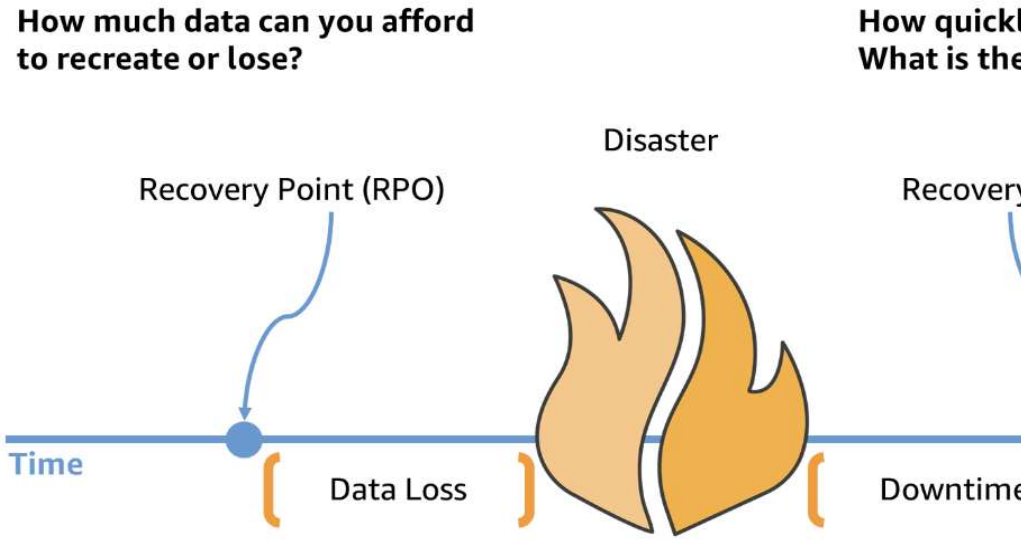
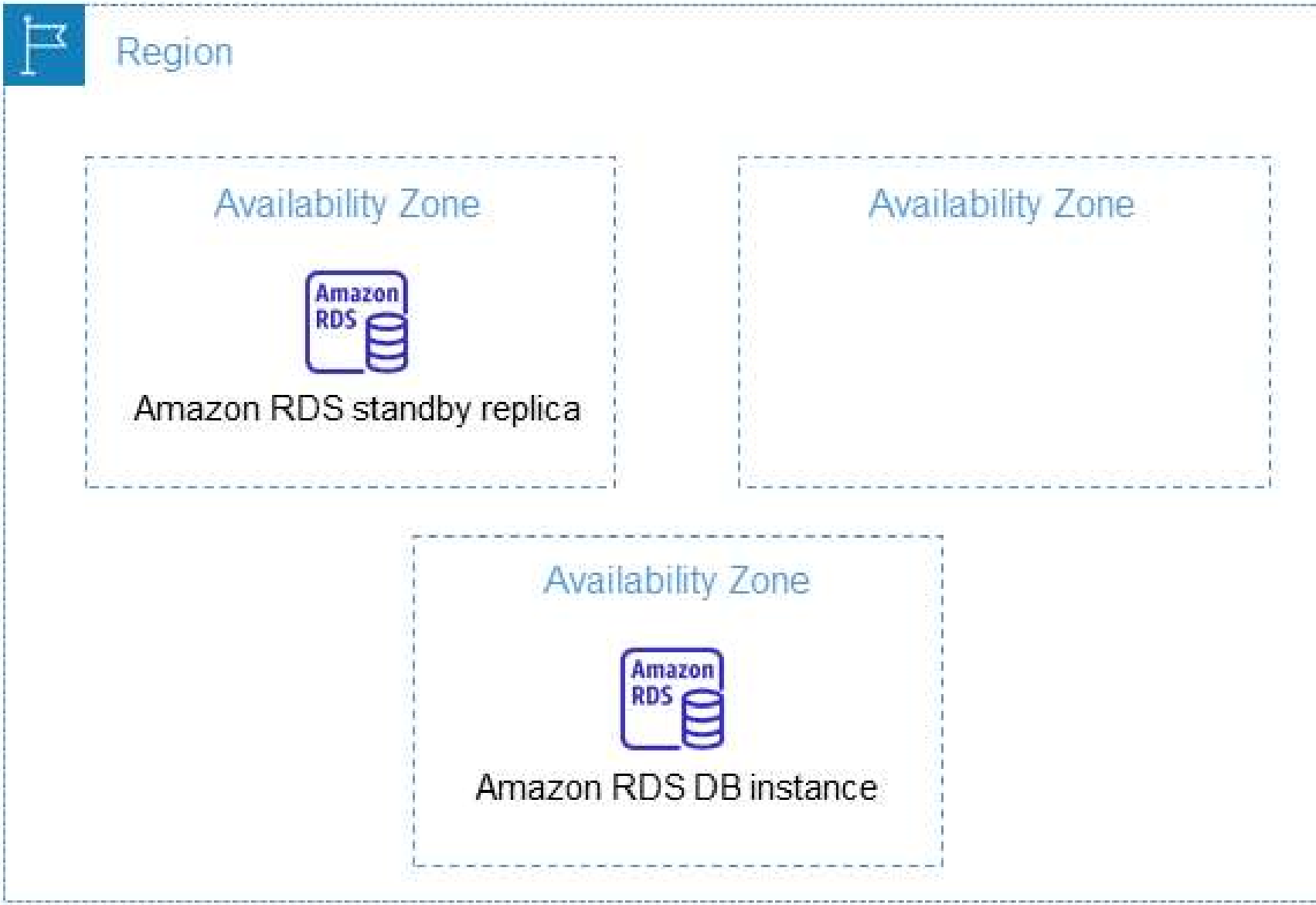


Image1:Data loss is measured from most recent backup to the point of d
measured from the point of disaster until fully recovered and available f

RDS Multi-AZ

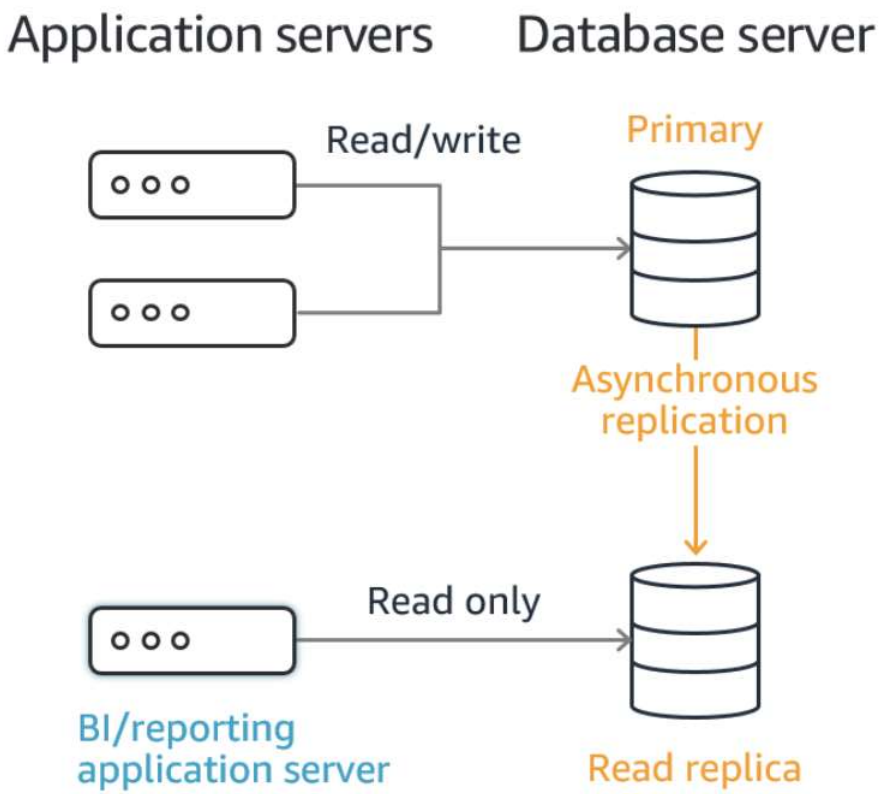
- Multi-AZ deployment improves database availability by maintaining a standby Replica in a different Availability Zon
- **For RDS (Non-Aurora)**:
  - Creates a **synchronous** standby instance in another AZ.
  - **Standby Instance**: is not readable – only for failover.
- **Failover Process**:
  - In case of an outage, the standby instance or read replica is prom primary
  - Failover typically completes within **~1–2 minutes**.
  - The database API remains the same during failover, so applicatio continues without changes.


Region
Availability Zone
Amazon RDS
Amazon RDS standby replica
Availability Zone
Availability Zone
Amazon RDS
Amazon RDS DB instance

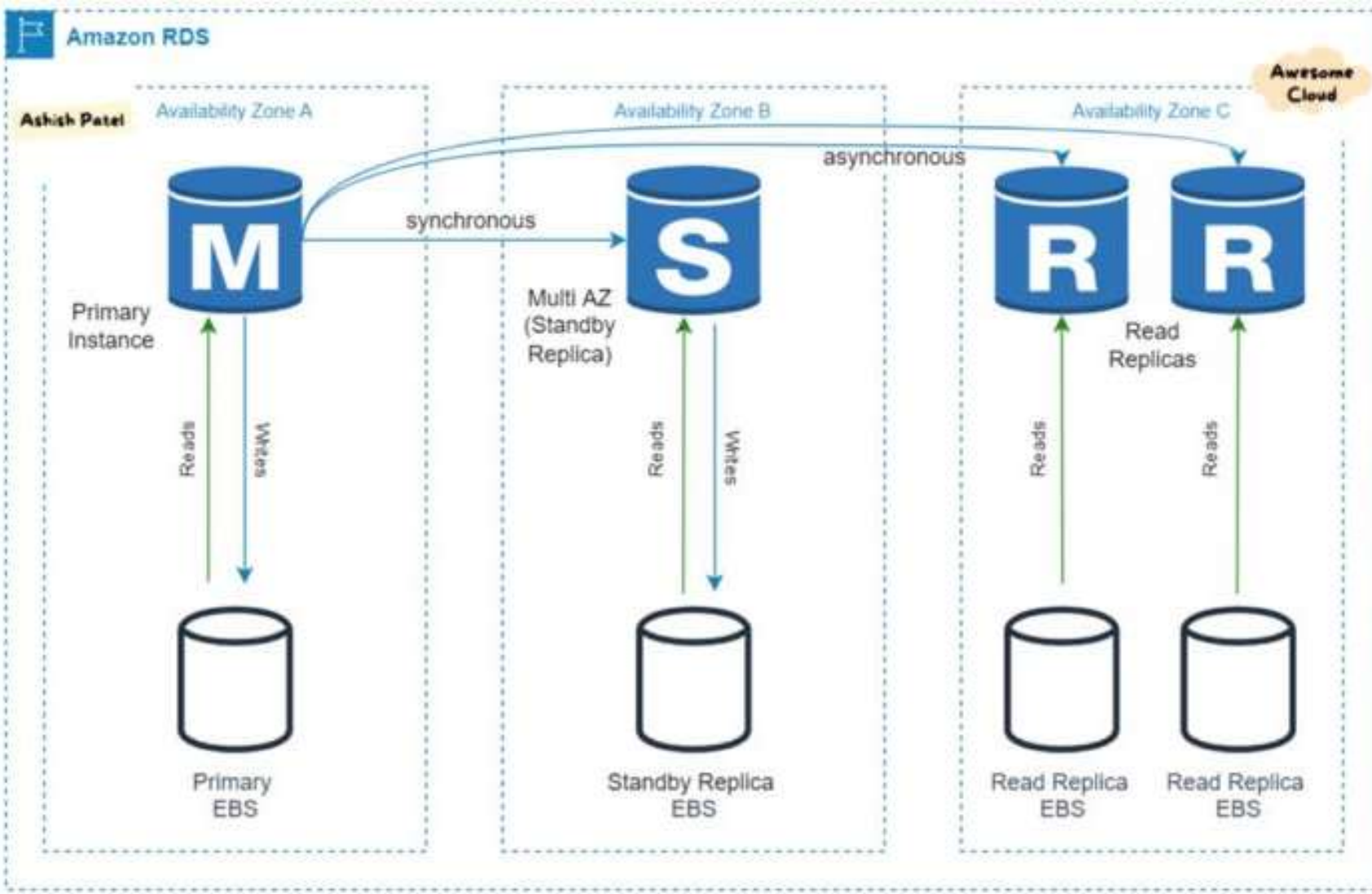
RDS Read Replica

- Amazon RDS Read Replicas enhance database performance and sca by enabling read-heavy workloads to be offloaded from the primary database.
- **Benefits**:
  - **Enhanced Performance**: Offloads read operations to replicas.
  - **Scalability**: Supports high read throughput by distributing read traffic.
  - **Redundancy**: Adds redundancy for disaster recovery.
- **Replication**:
  - **Asynchronous**: Ensures eventual consistency for reads.
  - **Secure**: Transactions during replication are encrypted.
- **Flexibility**:
  - Replicas can be **promoted** to standalone databases during failover or migratio
  - Improve availability: Read replicas can be deployed in **Multi-AZ**.

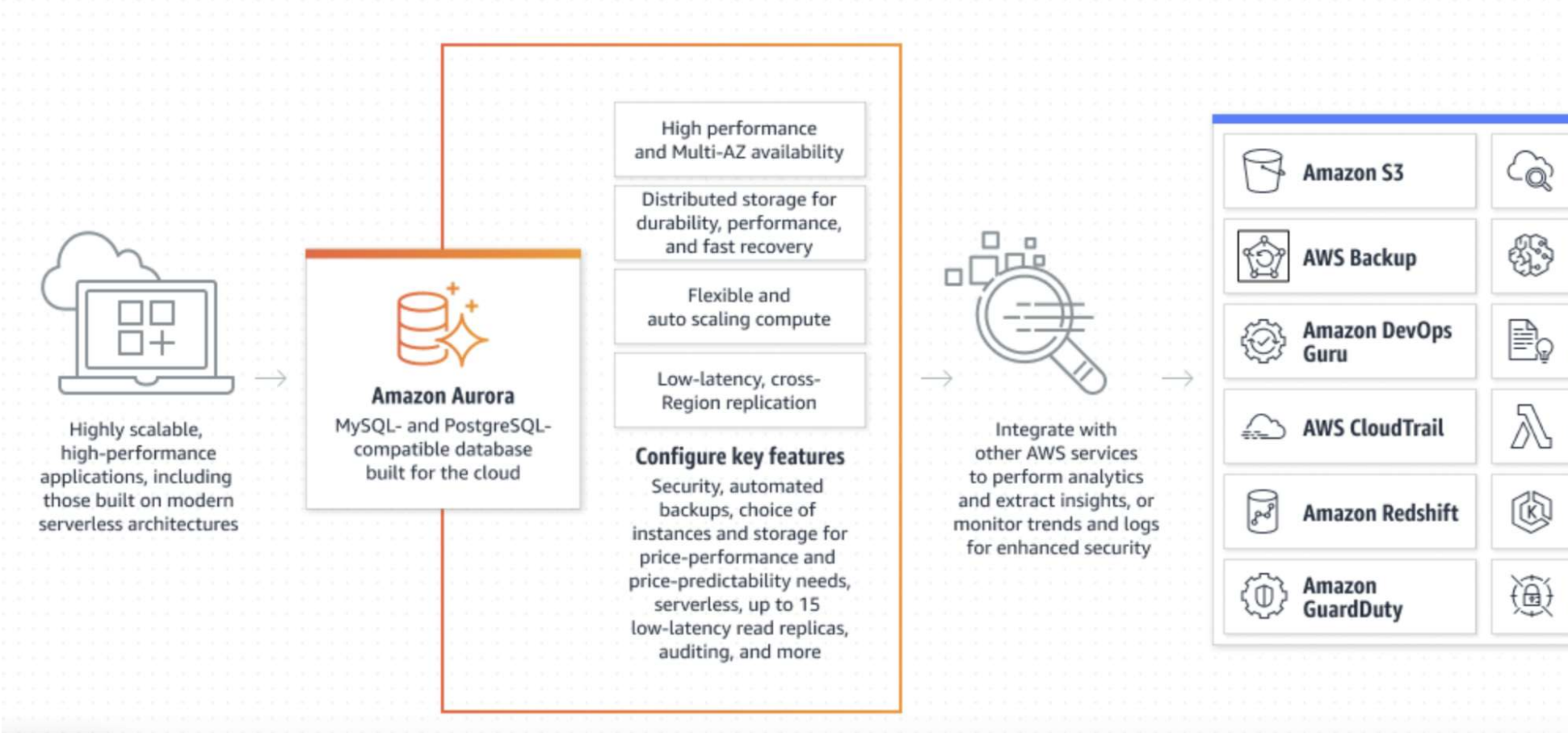
RDS Read replica



Amazon RDS (Non-Aurora)



Amazon Aurora



Amazon Aurora

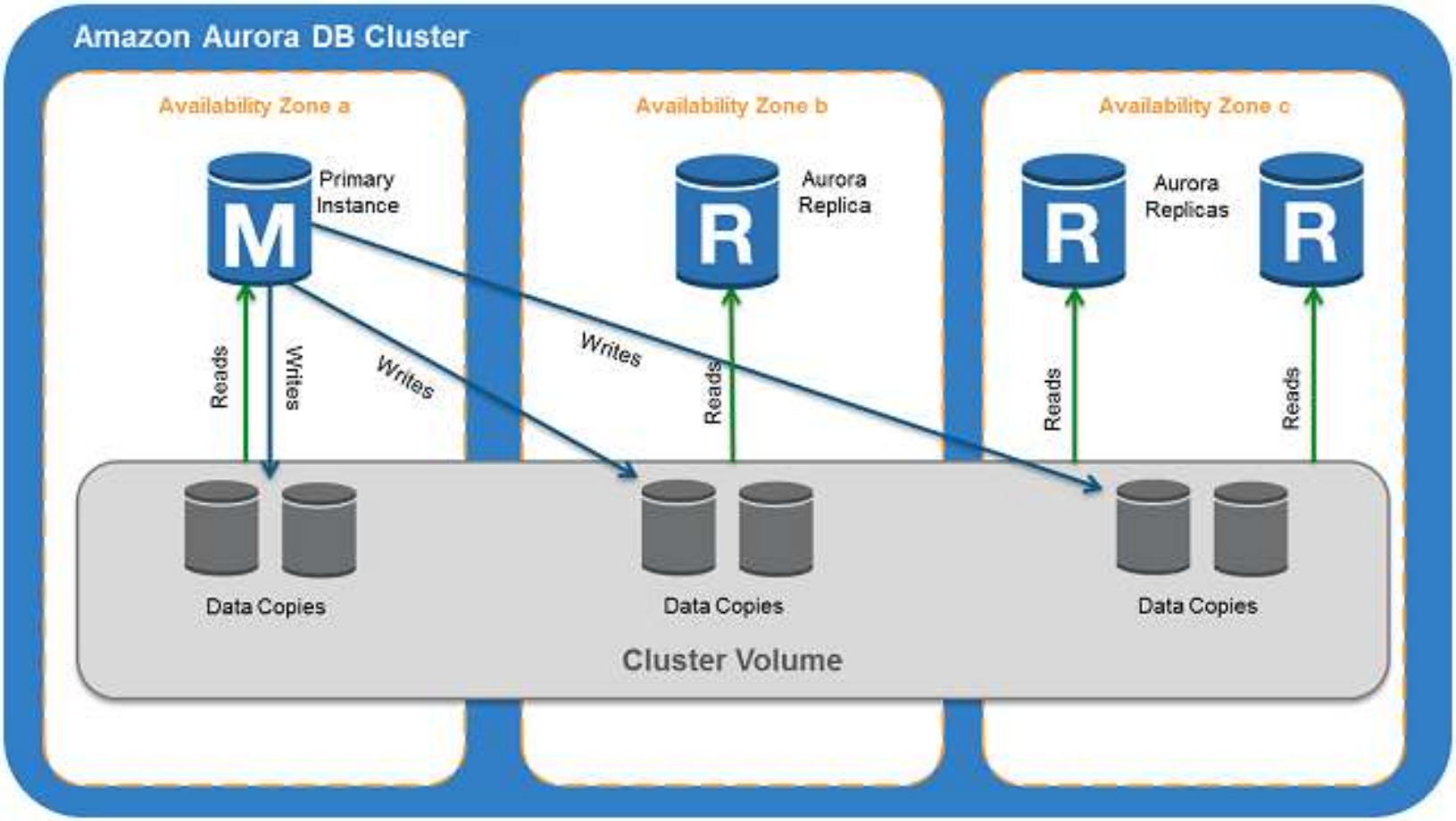
- Amazon Aurora is a fully managed relational database engine design high performance, scalability, and availability, compatible with **MySQ PostgreSQL**.
- **Performance**:
  - Up to **5x faster** than standard MySQL and **3x faster** than PostgreSQL.
  - Supports millions of transactions per second.
- **Scalability**:
  - Automatically scales storage up to **128 TiB**.
  - Supports up to **15 low-latency read replicas** for read-heavy workloads.
- **High Availability**:
  - Replicates data across **6 copies** in 3 Availability Zones (AZs).
  - Automatic failover occurs in less than 30 seconds, ensuring high resilience.
- **Cost-Effective**: Offers the performance of commercial databases at fraction of the cost, with pay-as-you-go pricing.

Amazon Aurora

- **Backup and Durability**:
  - Continuous: Automatically backs up to S3 in real time
  - Incremental: Only changes are backed up
  - Support PITR.
- **Security:**
  - Encryption at rest and in transit.
  - Integrated with AWS IAM.
- **Global Database**:
  - **Cross-region replication** for global applications.
  - Enables **low-latency reads** and **disaster recovery** across continents.
  - Failover from primary to secondary regions in **under a minute**.

Amazon Aurora

- It contains a primary instance and multiple replicas



Aurora Cluster

- **Cluster Components**:
  - **DB Instances**: One **Primary DB Instance** + up to **15 Aurora Repl**
  - **Cluster Volume**: A **distributed, virtual storage layer** spanning **3 Availability Zones (AZs)**, with **6 copies** of the data (2 per AZ).
- **Primary DB Instance**:
  - Handles **read and write operations**.
  - Performs all **data modifications** to the cluster volume.
  - Each cluster has **only one primary instance**.

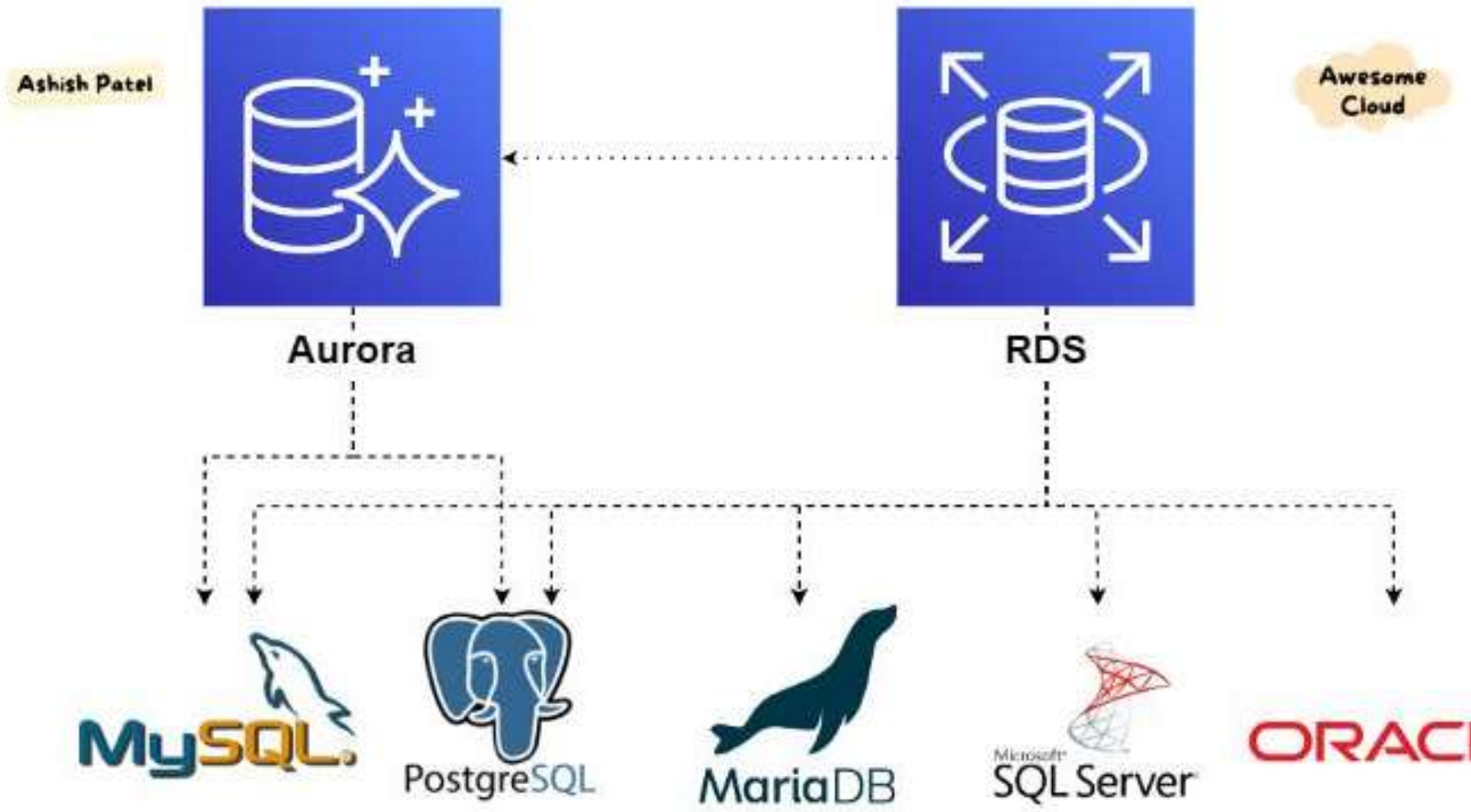
Aurora Replicas

- **Shared Storage**: Aurora replicas share the same underlyin storage as the primary instance, reducing costs and avoidi duplication.
- **Scalability**: Each Aurora DB cluster supports up to **15 repl** addition to the primary DB instance.
- **High Availability**:
  - Replicas can be placed in **separate Availability Zones (AZs)** for f tolerance.
  - Automatic failover to a replica occurs within **30 seconds** if the pr instance becomes unavailable.

Aurora Failover

- **Replication Efficiency**:
  - N**ot copy or transfer data** during replication.
  - Minimize **replication lag**, typically **less than 10 milliseconds** aft
    updates by the primary instance.
- **Failover Mechanisms**:
  - **With Aurora Replica**: Promotes an existing replica to primary wit
    seconds.
  - **Without Aurora Replica**: Recreates a new primary instance, whi
    about **10 minutes**.

Feature	Amazon RDS (Non-Aurora)	Amazon Aurora
Database Engines	Supports MySQL, PostgreSQL, Oracle, SQL Server, MariaDB.	Compatible with MySQ
Performance	Performance depends on instance type and storage configuration.	Up to 5x faster than My than PostgreSQL.
Replication	Synchronous standby replica in Multi-AZ for high availability.	Aurora Replicas share t distributed storage with typical replication lag is
Scalability	Manual scaling; limited to a single instance for writes.	Auto-scales storage (up supports up to 15 low-la replicas.
Failover	Promotes standby replica during failover (~1-2 minutes).	Automatic failover to Au seconds); creates new replica is available (~10
Storage	Single AZ or Multi-AZ replication for durability.	Distributed, fault-tolera AZs with 6-way replicati
Cost	Lower cost; charges based on instance type and storage.	Higher cost but optimiz performance and scala
Use Case	Suitable for standard workloads with traditional RDBMS needs.	Ideal for high-performan availability, and large-sc



Source: [AWS — Difference between Amazon Aurora and Amazon RDS]

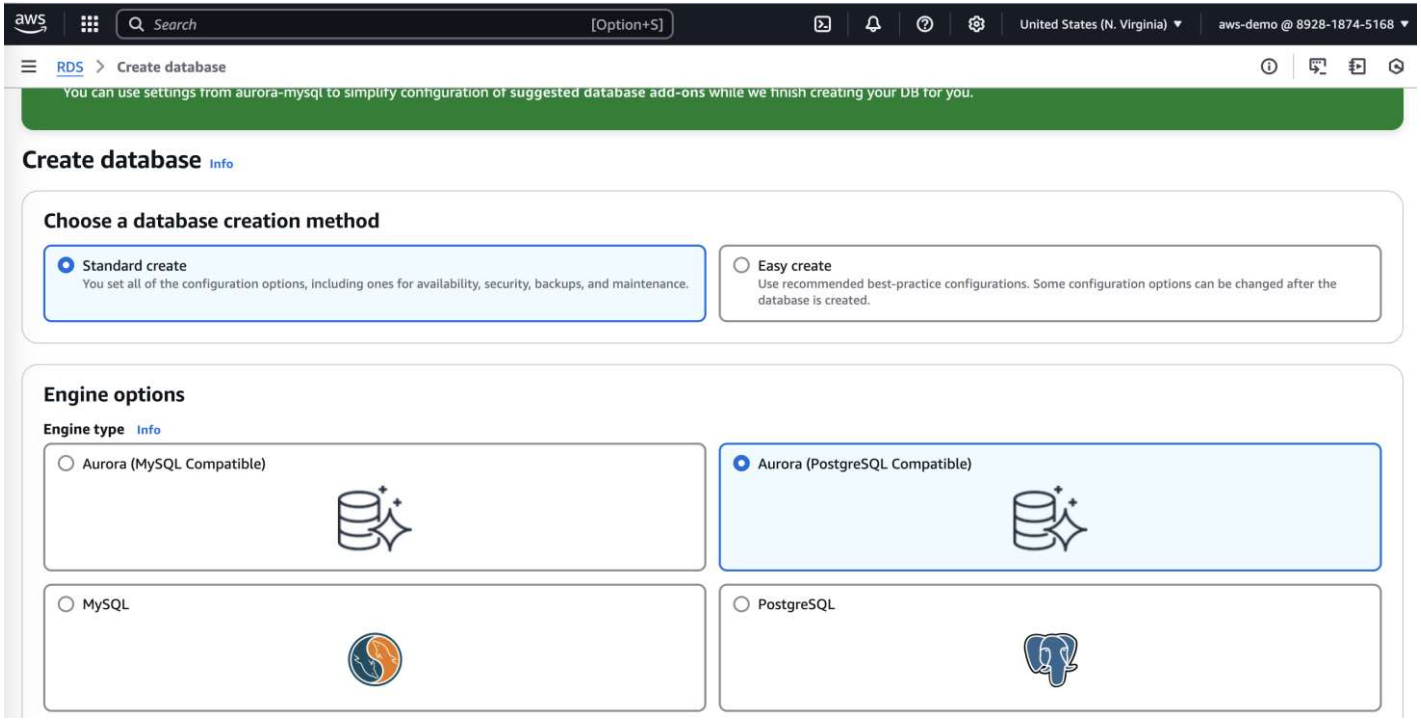
Best practices and considerations

- Security:
  - **Encryption**: Encrypt data both at rest using AWS Key Management Service (KMS) a transit using SSL/TLS to protect sensitive data.
  - **Access Control**: Implement least privilege access using IAM to control who can m access database resources.
  - **Network Security**: Advocate for hosting databases in private subnet within a VPC them from the public internet and utilize security groups and network ACLs effecti
- Performance Optimization
  - **Right sizing**: choose the correct instance size and database type based on the wo specific needs for appropriate provisioning.
  - **Indexing and queries**: Using indexing strategies and writing efficient SQL queries performance.
  - **Monitoring and tuning**: Regularly monitor databases to identify the performance and tune the database accordingly by using tools like CloudWatch or Performance

Best practices and considerations

- Cost Management:
  - **Managed Services**: Select the right types of services like Amazon RDS
  - **Storage Management**: Select the appropriate storage type and use au storage scaling
  - **Reserved Instances**: Purchasing Reserved Instances for databases wi predictable workloads can save up to 75% over equivalent on-demand
- High Availability and Disaster Recovery
  - **Multi-AZ deployments**: Deploy databases across multiple AZs to ensu availability and failover protection.
  - **Backup and Restore**: Regular backups, point-in-time recovery, and tes restore process to ensure data durability and recoverability.
  - **Read Replicas**: For read-heavy database workloads, recommend the u read replicas to enhance application performance and distribute loads

Demo


aws
Search
[Option+S]
United States (N. Virginia)
aws-demo @ 8928-1874-5168
RDS
Create database
You can use settings from aurora-mysql to simplify configuration of suggested database add-ons while we finish creating your DB for you.
Create database
Info
Choose a database creation method
Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.
Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.
Engine options
Engine type
Info
Aurora (MySQL Compatible)
Aurora (PostgreSQL Compatible)
MySQL
PostgreSQL

References

- https://docs.aws.amazon.com/
- ChatGPT: https://chatgpt.com/
- Google AI: https://gemini.google.com/app