DevOps

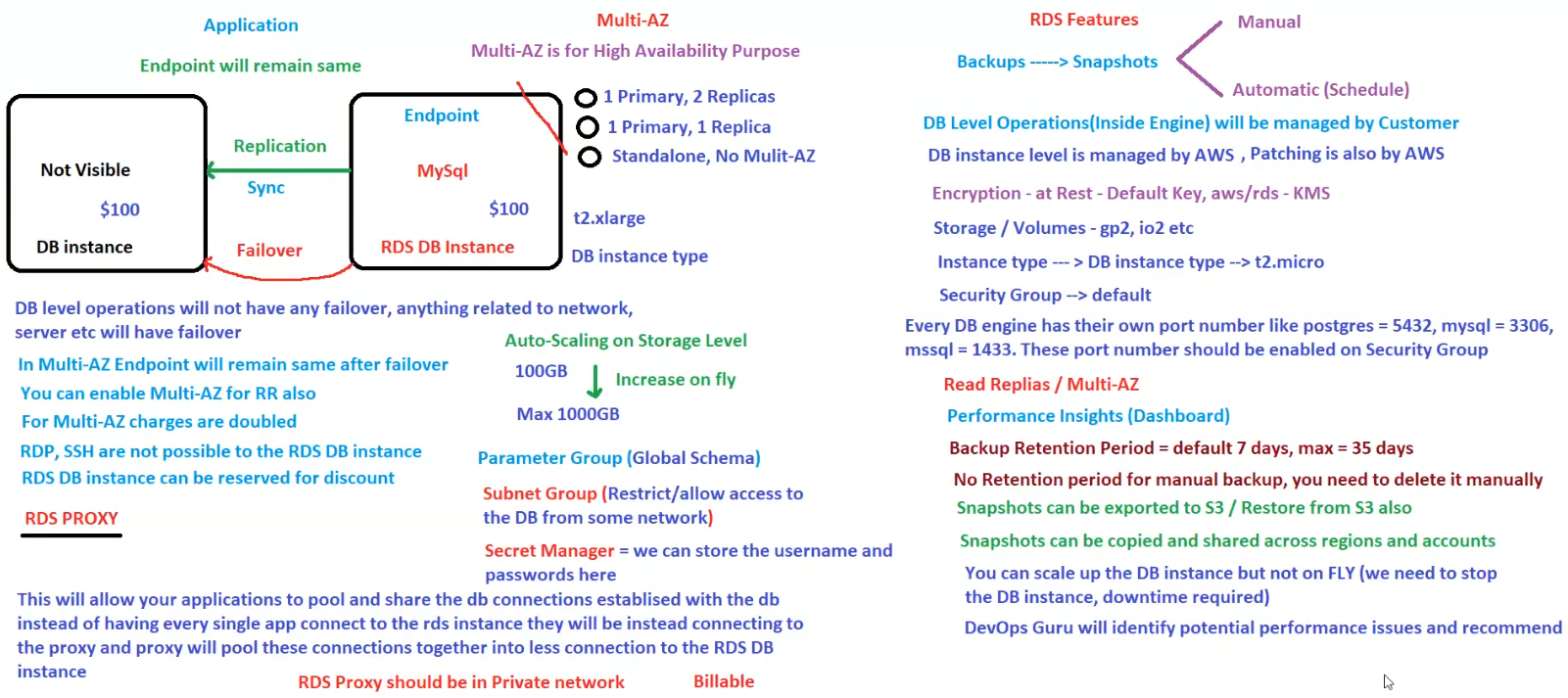
[RDS (Relational Database Service) 2](#_Toc193687630)

[Amazon RDS Overview 2](#_Toc193687631)

[Amazon RDS with Read Replicas (PostgreSQL) 5](#_Toc193687632)

# RDS (Relational Database Service)

## Amazon RDS Overview



**Amazon RDS (Relational Database Service) Overview**

Amazon RDS provides managed database instances with built-in **high availability, scalability, security, and automation** for backups and maintenance.

**1. High Availability with Multi-AZ**

* **Purpose**: Multi-AZ is for **high availability (HA)**.
* **Replication**: Synchronous replication between primary and secondary instances.
* **Failover Mechanism**:
  + If the primary instance fails, **AWS automatically switches to the standby replica**.
  + **Application endpoint remains the same** after failover.
* **Multi-AZ Configurations**:
  + **1 Primary, 2 Replicas**.
  + **1 Primary, 1 Replica**.
  + **Standalone (No Multi-AZ)**.
* **Additional Costs**:
  + **Multi-AZ charges are doubled** (since there are additional replicas).
* **Multi-AZ for Read Replicas (RR) is possible**.

**2. RDS Database Instance Details**

* **DB Engine Example**: MySQL
* **Instance Type**: t2.xlarge
* **Endpoint**: Remains the same even after failover.
* **DB Operations**:
  + **AWS manages** instance-level operations (like patching, maintenance).
  + **Customer manages** DB-level operations (inside the engine).

**3. Auto-Scaling on Storage Level**

* **Initial Storage**: 100GB.
* **Can increase dynamically** up to **1000GB**.

**4. Security and Access Control**

* **RDP and SSH are NOT possible** on RDS DB instances.
* **Security Group**:
  + Every database engine has **default port numbers**:
    - PostgreSQL → 5432
    - MySQL → 3306
    - SQL Server → 1433
  + Ports must be **allowed in security groups**.
* **Subnet Groups**:
  + Restrict/allow database access **from specific networks**.
* **Secret Manager**:
  + Store **database usernames and passwords** securely.

**5. RDS Features**

**Backups & Snapshots**

* **Manual** or **Automatic (Scheduled)**.
* **Backup Retention Period**:
  + Default **7 days**, max **35 days**.
* **Manual Backups have NO retention policy** (must be deleted manually).
* **Snapshots**:
  + Can be **exported to S3** and **restored from S3**.
  + Can be **copied and shared** across **regions & accounts**.

**Scaling**

* **Storage Auto-Scaling** is possible.
* **Instance Type Scaling**:
  + **Not possible on the fly** (requires stopping the instance, causing downtime).

**Performance Insights**

* **Performance Insights Dashboard** helps monitor and optimize performance.
* **DevOps Guru** identifies potential performance issues and suggests improvements.

**6. RDS Proxy (For Connection Pooling)**

* **Why Use RDS Proxy?**
  + Allows applications to **pool and share database connections**.
  + Reduces **overhead** when multiple apps connect to RDS.
* **Important Considerations**:
  + **Must be deployed in a Private Network**.
  + **Billable** (additional cost applies).

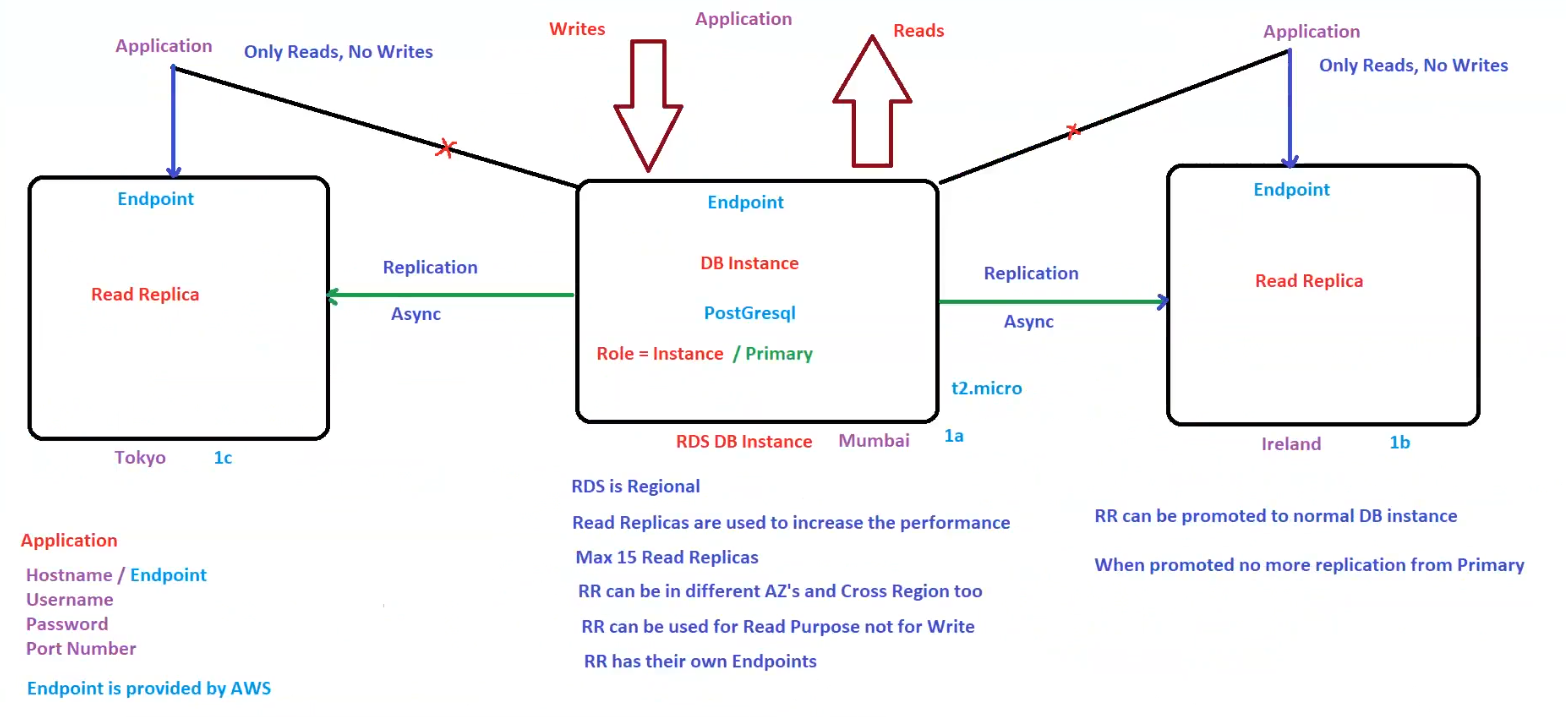
**7. Summary & Key Takeaways**

✅ **Multi-AZ ensures High Availability** – automatic failover with no endpoint change.  
✅ **Auto-scaling** increases storage dynamically.  
✅ **Security best practices**: Use security groups, subnet groups, and Secret Manager.  
✅ **RDS Proxy improves connection handling** and should be in a **private network**.  
✅ **Backups and Snapshots** provide disaster recovery but have different retention policies.

**Conclusion**

AWS RDS is a fully managed **relational database service** with **high availability, security, automated maintenance, and performance insights**. **RDS Proxy, Multi-AZ, and Auto-Scaling** improve efficiency and reliability.

## Amazon RDS with Read Replicas (PostgreSQL)



**Amazon RDS with Read Replicas (PostgreSQL)**

This architecture explains how **Amazon RDS Read Replicas** improve database performance and enable **cross-region replication**.

**1. Primary RDS DB Instance (Mumbai - 1a)**

* **Database Engine**: PostgreSQL
* **Role**: **Primary Instance**
* **Endpoint**: Provided by AWS.
* **Handles**:
  + **Write Operations**.
  + **Read Operations** (if read replicas are not used).

**2. Read Replicas (RR)**

**Purpose of Read Replicas**

* Improve **performance and scalability** by offloading read queries.
* **Asynchronous Replication** from **Primary Instance**.
* Applications can **read from Read Replicas**, but **cannot write** to them.
* **Each Read Replica has its own endpoint**.

**Read Replica Locations**

* **Tokyo (1c)**
* **Ireland (1b)**

**Key Features**

* **Max 15 Read Replicas** per primary DB.
* Read Replicas **can be across different Availability Zones (AZs) and Regions**.
* **Used for Read Operations only**.
* Can be **promoted to a standalone DB instance**.
  + Once promoted, it **stops receiving replication** from the primary DB.

**3. Replication Process**

* **Asynchronous Replication**:
  + Changes made in the **Primary DB Instance** are **replicated asynchronously** to Read Replicas.
  + There may be a **lag** depending on network and workload.

**4. Application Integration**

* Applications should **write only to the Primary Instance**.
* Applications should **read from Read Replicas** using their **specific endpoints**.

**5. Key Notes**

✅ **RDS is Regional** (Cannot span multiple regions except for Read Replicas).  
✅ **Read Replicas help scale read-heavy workloads**.  
✅ **Replication is asynchronous**, meaning some delay in data updates is expected.  
✅ **Read Replicas can be promoted to standalone DBs**, but will stop receiving updates from the primary instance.

**Conclusion**

AWS RDS Read Replicas provide **scalability and high availability** by offloading read operations, making them ideal for **read-heavy applications** like analytics and reporting.