

Amazon RDS

Introduction to Amazon RDS



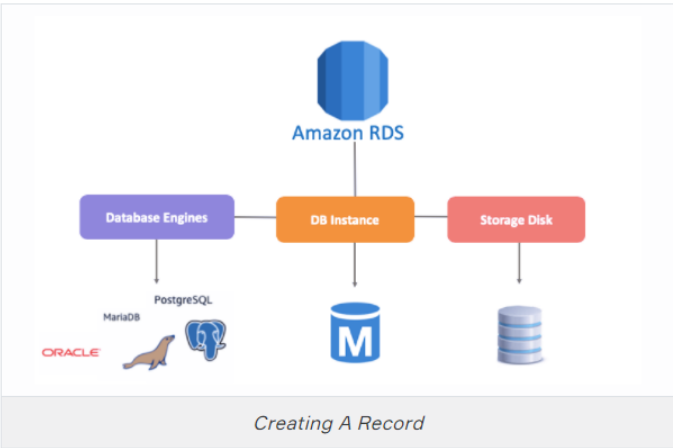
Amazon RDS is the SQL database service managed by AWS.

Since Amazon RDS base is a managed service as we said above, the setup, resource management and maintenance operations of the computer environment where the database runs are performed by AWS.

Naturally, the design and data modification processes of the database belong to the user.

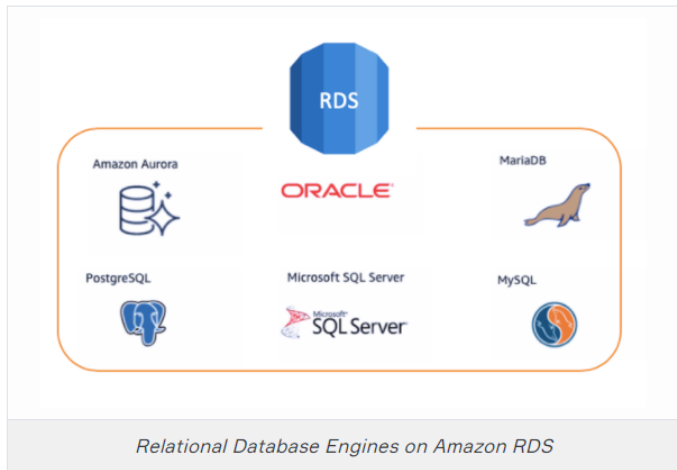
Amazon RDS enables to use of popular database engines in the database environment like Oracle, Microsoft SQL Server, MySQL, PostgreSQL, and MariaDB. Amazon RDS also offers its own relational database product, Amazon Aurora.

Components of Amazon RDS



- Amazon RDS basically consists of 3 main components:
 - The first and most basic component is a suitable **Database Engines** for our needs such as MySQL, PostgreSQL, MariaDB, etc. It is a kind of database software.
 - The second component is the **DB Instance** on which the database software we choose will run.
 - The third component is the virtual **Storage Disk** which will be connected to the DB instance.
- In fact, viewed from the perspective of the components, we can say the Amazon RDS is similar to EC2.

Relational Database Engines on Amazon RDS



There are 6 Relational database engines within Amazon RDS, one of which is Amazon's own product.

- **Amazon Aurora:**

Amazon Aurora is a **MySQL and PostgreSQL-compatible** relational database engine that combines the speed and availability of high-end commercial databases with the simplicity and cost-effectiveness of open source databases. Amazon Aurora provides up to **five times** better performance than MySQL with the security, availability, and reliability of a commercial database at **one-tenth the cost**.
- **Oracle:**

Amazon RDS allows you to deploy multiple editions of Oracle Database in minutes with cost-efficient and re-sizable hardware capacity. You can also bring existing Oracle licenses or pay for license usage by the hour.
- **Microsoft SQL Server:**

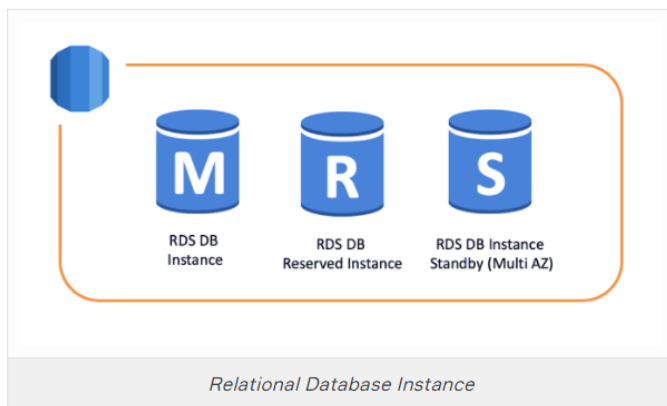
Amazon RDS for SQL Server makes it easy to set up, operate, and scale SQL Server in the cloud. You can deploy multiple editions of SQL Server including Express, Web, Standard, and Enterprise. Since Amazon RDS for SQL Server provides you direct access to the native capabilities of the SQL Server, your applications and tools should work without any changes.
- **MySQL:**

MySQL is an open-source relational database management system (RDBMS) used by a very large number of web-based applications. Amazon RDS for MySQL gives you access to the capabilities of a familiar MySQL database engine. This means that the code, applications, and tools you already use today with your existing databases can be used with Amazon RDS without any changes.
- **PostgreSQL:**

PostgreSQL is a powerful, enterprise-class open source object-relational database system with an emphasis on extensibility and standards-compliance. PostgreSQL boasts many sophisticated features and runs stored procedures in more than a dozen programming languages, including Java, Perl, Python, Ruby, Tcl, C/C++, and its own PL/pgSQL, which is similar to Oracle's PL/SQL.
- **MariaDB:**

MariaDB is a MySQL-compatible database engine which is a fork of MySQL and is being developed by the original developers of MySQL. With Amazon RDS, you can deploy scalable MariaDB databases in minutes with cost-efficient and resizable hardware capacity

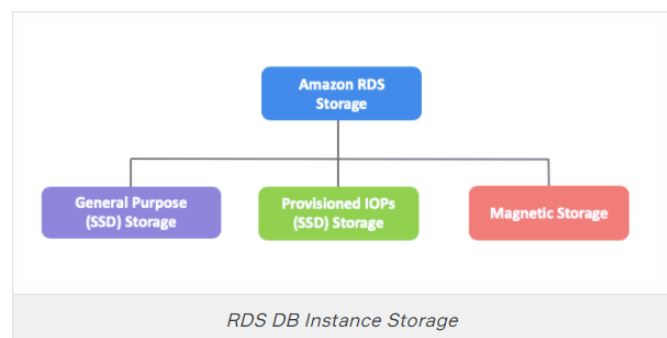
Relational Database Instance



You can think of a DB instance as a database environment in the cloud with the compute and storage resources you specify.

- According to the size of our database, we select a DB Instance considering the CPU and RAM power in the RDS environment.
- RDS offers **On-Demand** and **Reserved Instance** options.
- **Start** and **Stop** status are available just like EC2 virtual machines. So we can stop for a while and run it again. However, in the Amazon RDS service, a DB instance can only remain in "Stop" status for 7 days. If the machine is not put into operation after the 7th day, the machine is automatically started.
- RDS also allows the DB instance to be transferred to a more advanced and powerful machine without any interruption. This feature is valid for increasing the storage disk size.
- You can create and delete DB instances, define/refine infrastructure attributes of your DB instance(s), and control access and security via the AWS Management Console, Amazon RDS APIs, and AWS Command Line Interface.
- You can run one or more DB instances, and each DB instance can support one or more databases or database schemas, depending on the engine type.

Amazon RDS DB Instance Storage



There is an important factor in the databases as much as CPU and RAM power, which is the value of IOPs. We determine the IOPs value while choosing a storage disk. There are 3 types of Storage disks in Amazon RDS:

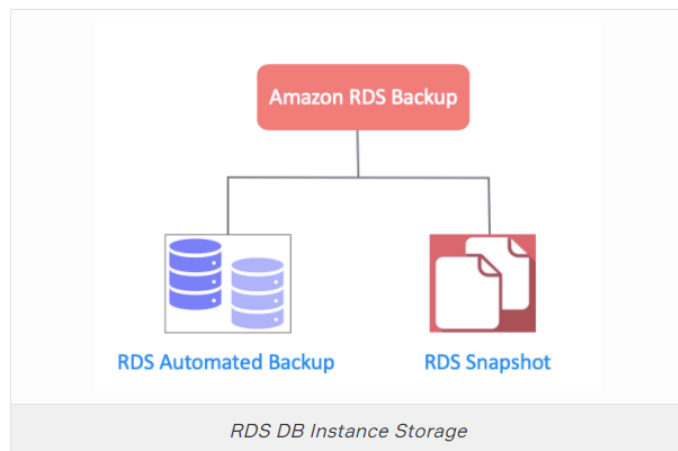
- **General Purpose (SSD) Storage:**
 - General Purpose SSD storage offers **cost-effective storage** that is acceptable for most database workloads.
 - RDS offers the **General Purpose (SSD) Storage** option, which has a storage capacity of 20 GB to 64 TB as part of the disk infrastructure and gives a value of **3 IOPS per GB**, provided that it has an upper limit of up to 3000 IOPS in total.
- **Provisioned IOPS (SSD) Storage:**
 - For production application that requires **fast and consistent I/O performance**, AWS recommends Provisioned IOPS storage. Provisioned IOPS storage is a storage type that delivers predictable performance and consistently low latency.

- Amazon RDS offers the **Provisioned IOPS (SSD) Storage** option, with 1000 to 80000 IOPs and a storage capacity of 20GB to 64 TB.

- **Magnetic Storage:**

- Amazon RDS also supports magnetic storage for backward compatibility. AWS recommends that you use General Purpose SSD or Provisioned IOPS SSD for any new storage needs.
- Magnetic storage doesn't allow you to scale storage when using the SQL Server database engine. It is limited to a maximum size of 3 TB and 1,000 IOPS. And it also doesn't support elastic volumes.

Automated Backups



Amazon RDS provides two different methods for backing up your DB instance:

- **Automated Backups**
- **Database Snapshots**
 - **Automated Backups:**

Thanks to Automated Backups, RDS performs the necessary updates and maintenance of the database at the time intervals we will choose. AWS call this time interval as a **Backup Window**.

We can choose a backup window while creating a DB instance. In this backup window, RDS provides an automated backup opportunity so that we can return up to 35 days.

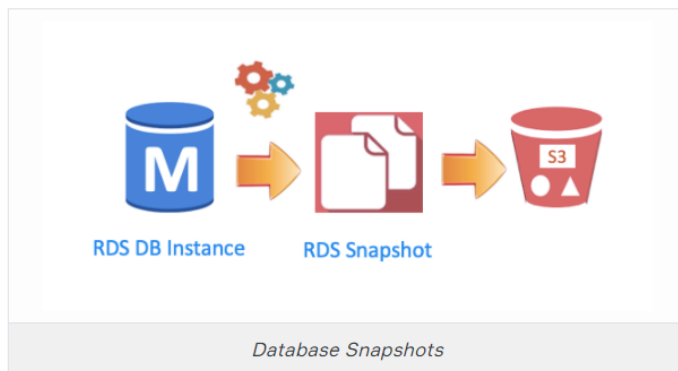
In addition to creating a complete copy of the database in automated backups, RDS backs up the **Transaction Log** records on S3 every 5 minutes, where all the transactions made in the database are kept. In this way, we have an opportunity to return to a desired moment of the day in seconds.

When you initiate a point-in-time recovery, transaction logs are applied to the most appropriate daily backup in order to restore your DB instance to the specific time you requested

⚠️ Avoid ! :

- In the RDS environment, system restores are never made on the existing RDS DB instance. Instead, a new DB instance is created and registered for each occasion

Database Snapshots



• Database Snapshots

Amazon RDS supports manual backups as well as automated backups. Amazon RDS allows taking Snapshot of RDS DB instance at any time.

DB Snapshots are user-initiated and enable you to back up your DB instance in a known state as frequently as you wish, and then restore to that specific state at any time.

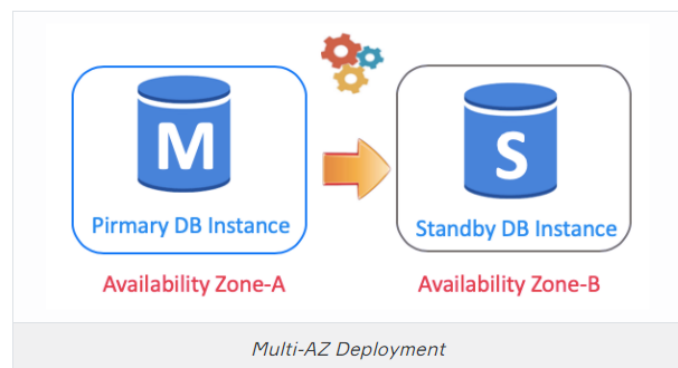
You can also identify DB Snapshots as an Automated Snapshot Type. So it take Snapshot in a specific time you determined.

⚠️ Avoid ! :

- When the RDS DB instance is deleted, an Automated Backups are deleted along with it. However, DB Snapshots remain on AWS even if the RDS DB instance is deleted.

Amazon RDS DB snapshots and automated backups are stored in S3.

RDS Multi-AZ Deployment



Thanks to the Multi-AZ Deployment, RDS creates a replica of the **primary database** for multiple Availability Zones within the same region. We call the second replica database as a **standby database**.

Amazon RDS automatically provisions and maintains a synchronous standby replica. Every information recorded in the primary database is synchronized instantly to the standby database located in the other AZ.

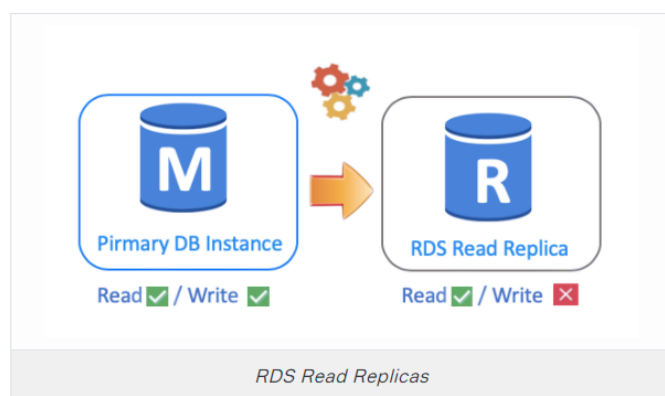
However, only the **primary database** responds to data queries. The standby database is promoted in failover scenarios. If a fault occurs in the primary database, the standby database is automatically activated.

Also, automated backups are taken from the standby database. In this way, possible interruptions and delays are also prevented.

💡 Tips:

- Multi-AZ Deployment certainly doesn't provide an increase of performance like a Load Balancer. It only ensures the continuity of the data flow.

Amazon Read Replicas



Actually, the main workloads of the databases are caused by reads.

For example, while looking for something on the Amazon website you make dozens of read processes from the Amazon databases. But only data is being written when you purchase something.

So, the biggest load of a database is to fetch data from it, that is, to read.

That's why RDS offers Read Replica. Thanks to Read Replica, RDS allows creating asynchronous database copies that we can create in the same or different regions.

But, unlike Multi-AZ, Read Replicas are designed to serve read traffic. Data writing processes are made only through our **source database**, and then Read Replica is synchronized with our source DB instance.

💡 Tips:

- Unlike RDS Multi-AZ feature, Amazon Read Replica improves database performance.