RDS

- Introduction to RDS
- Features
- Benefits
- Pricing

Amazon RDS

- **Amazon Relational Database Service (Amazon RDS)** is a database environment, which we can easily set up, operate in the cloud.
- It provides cost-efficient and resizable capacity freeing us to focus on our applications and business.
- Amazon RDS provides we six familiar database engines to choose from, including
- **Amazon Aurora**
- **Oracle**
- **Microsoft SQL Server**
- **PostgreSQL**
- **MySQL** and
- **MariaDB**

Few Customers using Amazon RDS are Vodafone, Unilever Flipboard etc

Benefits

Easy to Administer

- Amazon RDS makes it easy to go from project conception to deployment.
- Use the **AWS Management Console**, AWS RDS Command-Line Interface, or simple API calls to access production-ready relational database in minutes.
- No need of infrastructure provisioning, and no need of installing and maintaining database software.

Scalable

- we can **scale** our database's storage resources with only a few mouse clicks or an API calls, often with no downtime. Many Amazon RDS engine types allow us to launch one or more **Read Replicas** to offload read traffic from our primary database instance.

Inexpensive

- we pay very low rates and only for the resources we actually consume.
- In addition, we benefit from the option of On-Demand pricing with no up-front or long-term commitments, or even lower hourly rates via our reserved pricing option.

Available and Durable

- Amazon RDS runs on the same highly reliable infrastructure used by other Amazon Web Services.
- When we provision a **Multi-AZ DB Instance**, Amazon RDS synchronously replicates the data to a standby instance in a **different** Availability Zone (AZ).

- Amazon RDS has many other features that enhance reliability for critical production databases, including automated backups and database snapshots.

Fast

- Amazon RDS offers database server sizing choices up to **32 vCPUs and 244 GiB**, as well as storage choices for a wide range of application performance requirements.
- We can choose **SSD-backed** storage optimized for high-performance **OLTP** applications or for cost-effective general-purpose use.
- We can also choose **magnetic** storage for workloads in which data is accessed less frequently.

Secure

- Amazon RDS also lets us to run our database instances in Amazon **Virtual Private Cloud** (Amazon VPC), which enables us to isolate our database instances and to connect our existing IT infrastructure through an industry-standard encrypted **IPsec VPN**.
- Many Amazon RDS engine types offer **encryption** at rest and **encryption in transit**

- Amazon RDS gives us online access capabilities with **Amazon Aurora, MySQL, Oracle, Microsoft SQL Server, or PostgreSQL** relational database management system.
- **Code, applications**, and **tools** we already use today with our existing databases can be used with **Amazon RDS**.
- Amazon RDS handles routine database tasks such as **provisioning, patching, backup, recovery, failure detection, and repair**
- Amazon RDS has **replication** to enhance availability and reliability for production workloads.
- Using **Multi-AZ** deployment option, we can run **mission-critical** workloads with high availability and built-in automated fail-over from our **primary database** to a synchronously replicated **secondary database.**
- Using Read Replicas, Amazon RDS for MySQL, PostgreSQL, and Amazon Aurora, enable us to **scale out** beyond the capacity of a **single** database deployment for read-heavy database workloads
- As with all Amazon Web Services, there are no up-front investments required, and we **pay only for the resources we use**

Amazon Aurora

- Amazon Aurora is a relational database engine that **combines the speed and reliability of high-end commercial databases** with the simplicity and cost-effectiveness of open source databases.
- It delivers up to **five times the throughput of standard MySQL** running on the same hardware.
- Amazon Aurora is designed to be compatible with **MySQL 5.6**, so that existing MySQL applications and tools can run without requiring modification.
- Amazon Aurora joined, MySQL, Oracle, Microsoft SQL Server, and PostgreSQL, as the fifth database engine available to customers through Amazon RDS.

MariaDB

- Amazon RDS makes it easy to set up, operate, and scale MariaDB deployments in the cloud. With Amazon RDS, we can deploy scalable MariaDB databases in minutes with cost-efficient and resizable hardware capacity. Amazon RDS frees us up to focus on our application by managing time-consuming database administration tasks including backups, software patching, monitoring, scaling and replication

Oracle Database Editions

- Amazon RDS currently supports multiple Oracle Database Editions.
- Support for a given edition varies by licensing model.
- Standard Edition One: License Included, **Bring-Your-Own License**
- Standard Edition: Bring-Your-Own-License
- Enterprise Edition: Bring-Your-Own License

SQL Server Editions

- Amazon RDS supports multiple SQL Server (2008 R2 and 2012) Editions.
- Support for a given edition varies by licensing model.
- License Included: Express, Web and Standard Edition
- **BYOL**: Standard and Enterprise Edition

Instance Type	vCPU	Memory (GiB)	PIOPS-Optimized	Network Performance
Standard - current generation				
db.m3.medium	1	3.75	-	Moderate
db.m3.large	2	7.5	-	Moderate
db.m3.xlarge	4	15	Yes	Moderate
db.m3.2xlarge	8	30	Yes	High
Memory optimized - current generation				
db.r3.large	2	15	-	Moderate
db.r3.xlarge	4	30.5	Yes	Moderate
db.r3.2xlarge	8	61	Yes	High
db.r3.4xlarge	16	122	Yes	High
db.r3.8xlarge	32	244	-	10 Gigabit
Burstable performance instances				
db.t2.micro	1	1	-	Low to Moderate
db.t2.small	1	2	-	Low to Moderate
db.t2.medium	2	4	-	Low to Moderate

Migrating to Amazon RDS for MySQL

- If our application already relies on a MySQL database, importing data to Amazon RDS is simple. In general, to migrate our data to Amazon RDS we simply:
- Create a DB Instance with the compute, storage capacity and access controls.
- For smaller databases (e.g. up to 1 GB), extract the data with mysqldump and pipe it directly into Amazon RDS.
- Below is an example showing the acme database being copied to Amazon RDS:
- mysqldump acme | mysql --host=hostname --user=username --password acme
- For larger databases, build our database schema in Amazon RDS, then convert the data into a flat file and import it into our DB Instance using the mysqlimport utility.
- Below is an example showing the acme database being copied to Amazon RDS:

  mysqlimport --local --compress --user=username --password --host=hostname --fields-terminated-by=',' Acme sales.part_*
- Update the database connection string in our application config file

Features

Lower Administrative Burden

Easy to Use

- we can use the AWS Management Console, the Amazon RDS Command Line Interface, or simple API calls to access the capabilities of a production-ready relational database in minutes.
- Amazon RDS database instances are pre-configured with parameters and settings appropriate for the engine and class we have selected.
- We can launch a database instance and connect our application within minutes.

Monitoring and Metrics

- Amazon RDS provides Amazon CloudWatch metrics for our database instances at no additional charge.
- We can use the AWS Management Console to view key operational metrics, including compute/memory/storage capacity utilization, I/O activity, and instance connections.

Automatic Software Patching

- Amazon RDS will make sure that the relational database software powering our deployment stays up-to-date with the latest patches.
- We can exert optional control over when and if our database instance is patched.

Performance

General Purpose (SSD) Storage

- Amazon RDS General Purpose Storage is an SSD-backed storage option delivers a consistent baseline of 3 IOPS per provisioned GB and provides the ability to burst up to 3,000 IOPS.
- This storage type is suitable for a broad range of database workloads.

Provisioned IOPS (SSD) Storage

- Amazon RDS Provisioned IOPS Storage is an SSD-backed storage option designed to deliver fast, predictable, and consistent I/O performance.
- This storage type is optimized for I/O-intensive transactional **(OLTP) database workloads**.
- we can provision up to 30,000 IOPS per database instance, although our actual realized IOPS may vary based on our database workload, instance type, and database engine choice.

Scalability

Push-button Compute Scaling

- we can scale the compute and memory resources powering our deployment up or down, up to a maximum of 32 vCPUs and 244 GiB of RAM.

Easy Storage Scaling

- As our storage requirements grow, we can also provision **additional** storage.
- The Amazon Aurora engine will automatically grow the size of our database volume as our database storage needs grow, up to a maximum of **64 TB** or a maximum we define.
- **The MySQL, Oracle, and PostgreSQL engines allow we to provision additional storage on-the-fly with zero downtime**.
- The MySQL, Oracle, and PostgreSQL engines also allow us to scale the throughput of our database instance: with **General** Purpose (SSD) storage, up to a maximum of **3000** IOPS by provisioning more storage; and with **Provisioned** IOPS (SSD) storage, up to a maximum of **30,000** IOPS by increasing the amount of IOPS provisioned.

Availability and Durability

Automated Backups

- Turned on by default, the automated backup feature of Amazon RDS enables **point-in-time recovery** for our database instance.
- Amazon RDS will backup our database and transaction logs and store both for a user-specified retention period.
- This allows us to restore our database instance during our **retention period**, up to the last five minutes. our automatic backup retention period can be configured to up to **thirty-five days**.

Database Snapshots

- Database snapshots are user-initiated backups of our instance stored in Amazon S3 that are kept until we **explicitly delete them**.
- We can create a new instance from a database snapshots whenever we desire. Although database snapshots serve operationally as full backups, we are billed only for incremental storage use.

Security

Encryption at Rest and in Transit

Network Isolation

- AWS recommends that we run our database instances in Amazon VPC, which allows us to isolate our database in our own virtual network and connect to our on-premises IT infrastructure using industry-standard encrypted IPsec VPNs.

Resource-level Permissions

- Amazon RDS is integrated with AWS Identity and Access Management (IAM) and provides we the ability to control the actions that our AWS IAM users and groups can take on specific Amazon RDS resources, from database instances through snapshots, parameter groups, and option groups

Manageability

Monitoring and Metrics

- Amazon RDS for Aurora provides **Amazon CloudWatch metrics** for our database instances at no additional charge.

- We can use the AWS Management Console to view over 40 key operational metrics for our database instances, including compute, memory, storage, query throughput, cache hit ratio, and active connections.

Automatic Software Patching

- Amazon RDS will keep our database up-to-date with the latest patches. we can exert **optional control over when and if our instance is patched**.

DB Event Notifications

- Amazon RDS can notify us via email or SMS text message of database events through Amazon SNS. we can use the AWS Management Console or the Amazon RDS APIs to subscribe to over **40 different database events** associated with our database instances.

Pricing

- When we run our DB Instance as a Multi-AZ deployment for enhanced data durability and availability, Amazon RDS provisions and maintains a standby in a different Availability Zone for automatic failover in the event of a scheduled or unplanned outage.

Region:US East (N. Virginia)

db.t2.micro	0.034
db.t2.small	0.068
db.t2.medium	0.136
db.t2.large	0.272
db.m3.medium	0.18
db.m3.large	0.37
db.m3.xlarge	0.74
db.m3.2xlarge	1.48

db.r3.large	0.48
db.r3.xlarge	0.95
db.r3.2xlarge	1.89
db.r3.4xlarge	3.78
db.r3.8xlarge	7.56

If we Need	Consider Using
A relational database service with minimal administration	Amazon RDS, a managed relational database service that offers a choice of MySQL, Oracle, SQL Server, PostgreSQL, or Amazon Aurora database engines, scale compute & storage, Multi-AZ availability, Read Replicas, and more.
A fast, highly scalable nonrelational database service	Amazon DynamoDB, a managed NoSQL database that offers extremely fast performance, seamless scalability and reliability, low cost and more.
An easy-to-operate in-memory cache	Amazon ElastiCache, a managed in-memory cache cloud service that offers a choice of two in-memory caching engines, Memcached and Redis
A fast, petabyte-scale data warehouse	Amazon Redshift, a managed data warehousing service that makes it simple and cost-effective to efficiently analyze all our data using our existing business intelligence tools
A relational database we can manage on our own	our choice of relational AMIs on Amazon EC2 and EBS that provide scale compute & storage, complete control over instances, and more.

What does Amazon RDS manage on my behalf?

- Amazon RDS manages the work involved in setting up a relational database: from provisioning the infrastructure capacity we request to installing the database software. Once our database is running on its own DB Instance, **Amazon RDS automates common administrative tasks, such as performing backups and patching the database software that powers our DB Instance**.
- For optional Multi-AZ deployments, Amazon RDS also manages synchronous data replication across Availability Zones and automatic failover.

- Since Amazon RDS provides native database access, we interact with the relational database software as we normally would.
- This means you're still responsible for managing the database settings that are specific to our application.

  You'll need to build the relational schema that best fits our use case and are responsible for any **performance** tuning to optimize our database for **our application's** workflow

Free Tier

Q: What does the AWS Free Tier for Amazon RDS offer?

The AWS Free Tier for Amazon RDS offer provides free use of Single-AZ Micro DB instances running MySQL, PostgreSQL, Oracle ("Bring-Your-Own-License (BYOL)" licensing model) and SQL Server Express Edition. The free usage tier is capped at 750 instance hours per month. Customers also receive 20 GB of database storage, 10 million I/Os and 20 GB of backup storage for free per month.

Q: For what time period will the AWS Free Tier for Amazon RDS be available to me?

- New AWS accounts receive 12 months of AWS Free Tier access.

Q: Can I run more than one DB instance under the AWS Free Usage Tier for Amazon RDS?

- Yes. we can run more than one Single-AZ Micro DB instance simultaneously and be eligible for usage counted under the AWS Free Tier for Amazon RDS. However, any use exceeding 750 instance hours, across all Amazon RDS Single-AZ Micro DB instances, across all eligible database engines and regions, will be billed at standard Amazon RDS prices.

For example: if we run two Single-AZ Micro DB instances for 400 hours each in a single month, we will accumulate 800 instance hours of usage, of which 750 hours will be free. we will be billed for the remaining 50 hours at the standard Amazon RDS price