

# Aurora



**Fully Managed** Postgres or MySQL compatible database  
**designed by default to scale** and **fine-tuned to be really fast**.



## Introduction to Aurora

Combines the **speed** and **availability** of **high-end databases** with the **simplicity** and **cost-effectiveness** of **open source databases**.

Aurora can run either MySQL or Postgres compatible engines



Aurora MySQL is **5x better performance** than traditional MySQL



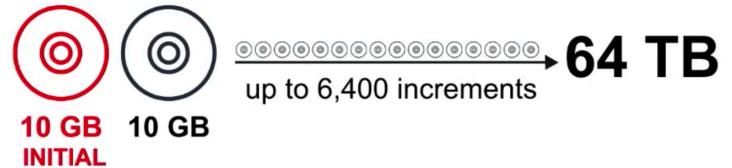
Aurora Postgres is **3x better performance** than traditional Postgres

**1/10th** the costs of other solutions offering similar performance and availability.



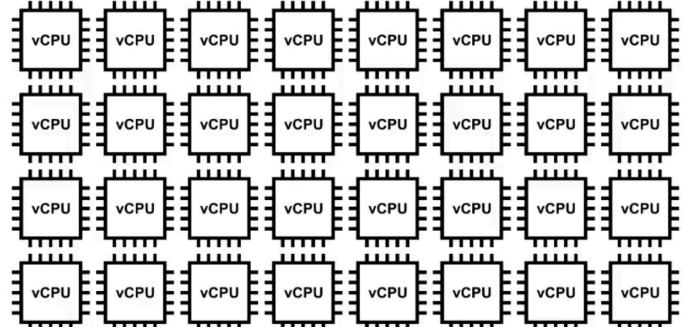
## Aurora - Scaling

Start with 10GB of storage, and scale in **10GB** increments up to 64TB.



Storage is autoscaling.

Computing resources can scale all the way up to **32 vCPUs** and **244GB** of memory.





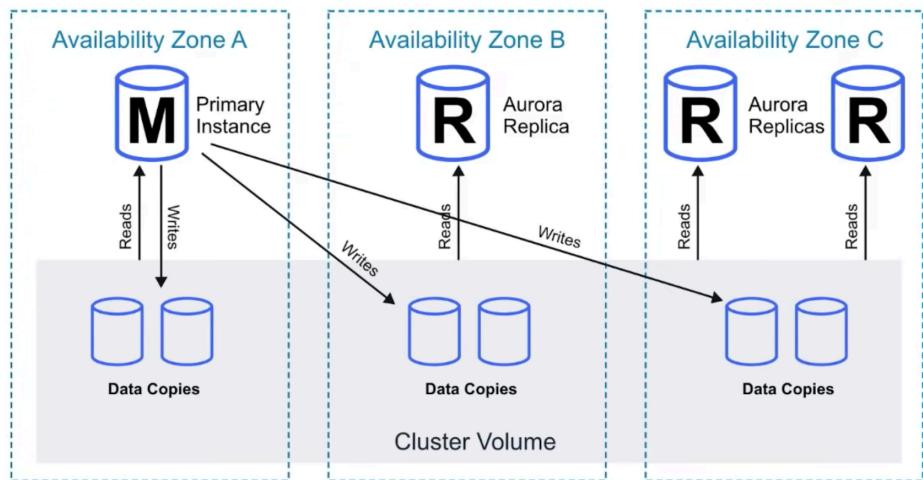
## Aurora - Availability

A minimum of **3 availability zones** each contain **2 copies** of your data at all times.

That means there are **6 copies**

Lose up to 2 copies of your data without affecting **write** availability.

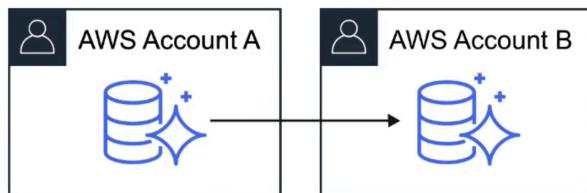
Lose up to 3 copies of your data without affecting **read** availability.





## Aurora – Fault Tolerance and Durability

Aurora Backup and Failover is handled **automatically**



Snapshots of data can be  
**shared** with other AWS  
accounts

Storage is **self-healing**, in that data blocks and  
disks are continuously scanned for errors and  
repaired automatically.



## Aurora - Replicas

There are  **2 types** of replicas available:

### Amazon Aurora Replicas    MySQL Read Replicas

Number of Replicas	Up to 15	Up to 5
Replication Type	Asynchronous (ms)	Asynchronous (s)
Performance impact on primary	Low	High
Act as failover target	Yes (no data loss)	Yes (potentially minutes of data loss)
Automated failover	Yes	No
Support for user-defined replication delay	No	Yes
Support for different data or schema vs primary	No	Yes



## Aurora - Serverless

Engine type [Info](#)



Aurora except the database will automatically **start up, shut down, and scale capacity up or down** based on your application's needs.

Apps used a few minutes several times per day or week, eg. low-volume blog site

pay for database **storage** and the **database capacity** and **I/O** your database consumes while it is active

### Database features

#### One writer and multiple readers

Supports multiple reader instances connected to the same storage volume as a single writer instance. This is a good general-purpose option for most workloads.

#### Serverless

You specify the minimum and maximum amount of resources needed, and Aurora scales the capacity based on database load. This is a good option for intermittent or unpredictable workloads.

### Capacity settings

This billing estimate is based on published prices. [Learn more](#)

Minimum Aurora capacity unit [Info](#)

2  
4GB RAM

Maximum Aurora capacity unit [Info](#)

384  
768GB RAM

#### ▼ Additional scaling configuration

Force scaling the capacity to the specified values when the timeout is reached [Info](#)

Enable to force capacity scaling as soon as possible. Disable to cancel the capacity changes when a timeout is reached

Pause compute capacity after consecutive minutes of inactivity [Info](#)

You are only charged for database storage while the compute capacity is paused



SUBSCRIBE



## Aurora CheatSheet

- When you need a **fully-managed** Postgres or MySQL database that needs to scale, automatic backups, high availability and fault tolerance *think* Aurora
- Aurora can run MySQL or Postgres database engines
- Aurora MySQL is 5x faster over regular MySQL
- Aurora Postgres is 3x faster over regular Postgres
- Aurora is 1/10 the cost over its competitors with similar performance and availability options.
- Aurora replicates **6 copies** for your database across **3 availability zones**.
- Aurora is allowed up to **15 Aurora Replicas**
- An Aurora database can span multiple regions via **Aurora Global Database**
- **Aurora Serverless** allows you to stop and start Aurora and scale automatically while keeping costs low
- Aurora Serverless is ideal for new projects or projects with infrequent database usage