# DynamoDB or Aurora

Abhishek Tiwari

Citation: *A. Tiwari*, ”DynamoDB or Aurora”, Abhishek Tiwari, 2015.
doi:10.59350/sdfqb-qga68

Published on: July 06, 2015

Aurora is a new offering from Amazon Relational Database Service (RDS). It is advertised as drop-in replacement for the MySQL - full compatibility with MySQL 5.6. Aurora is designed for fault-tolerance, availability and storage elasticity. It is a relational database and a highly cost effective one. With Aurora you never need to over-provision the storage, database instance volume will grow in increments of 10 GB up to a maximum of 64 TB.

## So how Aurora compares with DynamoDB?

I am glad you asked. Well, DynamoDB and Aurora are two different database as a service offerings from Amazon. Fundamentally DynamoDB is non-relational database, hence there is no apple to apple comparison with Aurora which is a relational database. Both are designed to solve different problems, but there are few aspects you may want to look closely. If you are using DynamoDB to store semi-structure or structure data with read and write throughputs as primary consideration, then Aurora may be a good news for you.

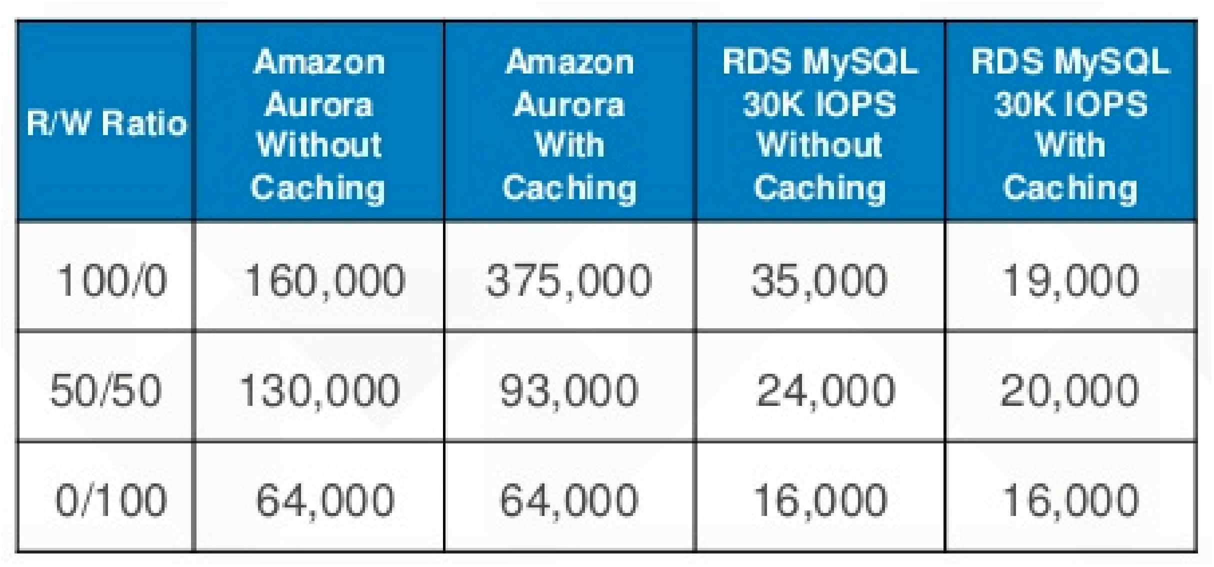
## Read/write throughput cost

I am not suggesting that read/write throughput is the primary reason one should choose between DynamoDB or Aurora but it is worth comparing the cost of throughput in both cases. If DynamoDB is fast, then Aurora is super-fast at least according to Amazon's own benchmarks.

An on-demand Aurora db.r3.8xlarge instance with 32vCP and 244GB RAM cost 4.64$ per hr or 3340$ per month. This instance reported,

- DML throughput of 105,000 per second (4000 concurrent connections) - 100% write
- Select throughput of 546,000 per second (1000 concurrent connections) - 100% read

A more generic read/write throughput comparison using 250 tables and 1000 concurrent connections reported following numbers,



| R/W Ratio | Amazon Aurora Without Caching | Amazon Aurora With Caching | RDS MySQL 30K IOPS Without Caching | RDS MySQL 30K IOPS With Caching |
|---|---|---|---|---|
| 100/0 | 160,000 | 375,000 | 35,000 | 19,000 |
| 50/50 | 130,000 | 93,000 | 24,000 | 20,000 |
| 0/100 | 64,000 | 64,000 | 16,000 | 16,000 |

**Figure 1:** Aurora read/write throughput benchmarks with or without caching

With a 50/50 read/write ratio, we expect 130,000 throughput per second. That will be 65,000 read or write per second just for **3340$ per month** using the largest Aurora instance. Now to achieve the same kind of throughput with strong consistency, Amazon DynamoDB will cost you about **39,995$ per month**. That means DynamoDB throughput is 11 times more costly than Aurora. In a nutshell, Aurora throughput is super cost effective.

Please note there is not much difference in storage cost between Aurora and DynamoDB.

## To partition or not to partition

DynamoDB is partitioning logic bit complicated and requires a good understanding of internals in order to produce good throughput performance. DynamoDB orchestrate table partitioning for you automatically. A single DynamoDB partition can hold approximately 10 GB of data which is coincidentally same as Aurora storage blocks of 10GB. If your table size grows beyond 10 GB, DynamoDB will spread your data across additional partitions. By growing the number of partitions you are effectively distributing you provisioned read and write throughput which means as data grow your queries may take longer unless keep adding additional throughput.

With Aurora there are no partitions, but one could greatly benefit by adding Aurora read replicas to reduce read load away from master instance. Aurora read replicas share the same underlying storage as the master instance also known as cluster volume so there are no write overheads.

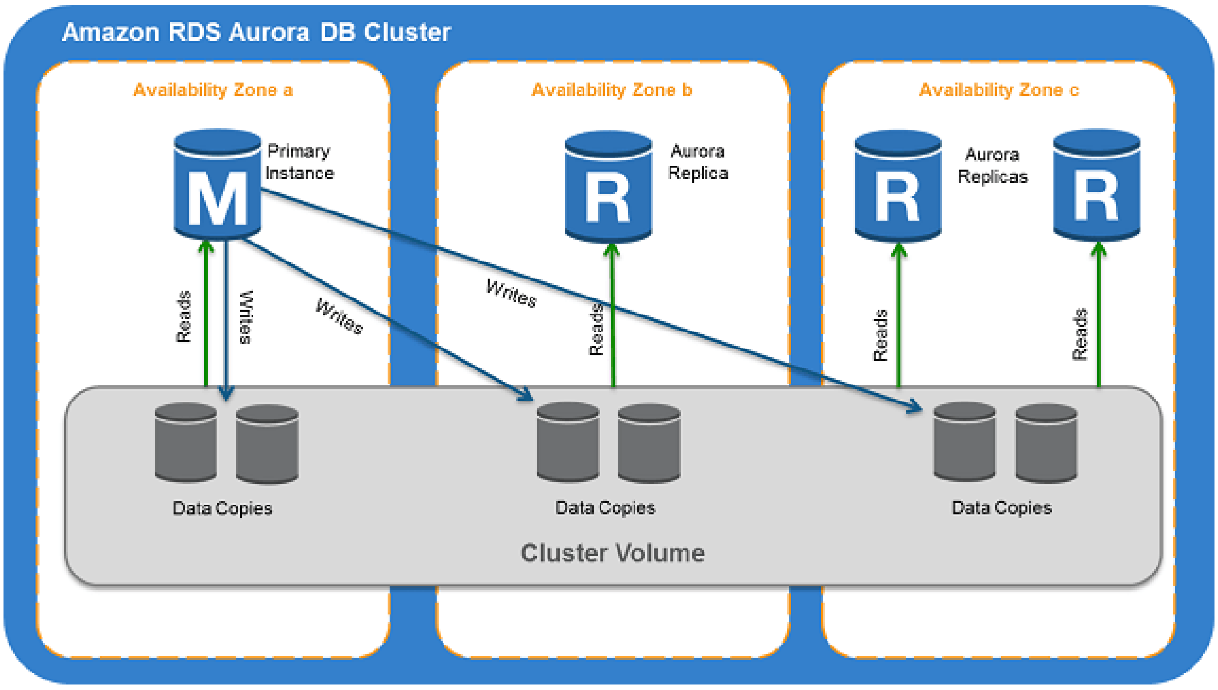


**Figure 2:** Amazon Aurora Architecture

## Closing thoughts

One of the key reasons people love DynamoDB is its simplicity. DynamoDB's ability to scale and transition it from a prototype database to full-blown high-load application database was unchallenged. Now it seems like, DynamoDB has a competitor with Aurora. Aurora offers both simplicity and continuity offered by DynamoDB. Moreover, it has the added benefit of super cost-effective throughputs.