

# Amazon RDS Aurora vs RDS MySQL vs MySQL on EC2?

Asked 7 years, 2 months ago   Modified 1 year, 7 months ago

Viewed 25k times    Part of [AWS](#) Collective



52



I've been looking around for best practices when setting up your database on the cloud but it still isn't clear to me which of the following solutions should we be going for?

- Amazon RDS Aurora
- Amazon RDS MySQL
- MySQL on EC2 instances

I see Amazon Aurora being marketed as the better alternative however after some research it doesn't seem like people are using it. Is there a problem with it?

aws

mysql

amazon-web-services

amazon-ec2

amazon-rds

amazon-aurora

Share

Improve this question

Follow

edited Sep 25, 2017 at 15:33



Mark B 

200k ● 27 ● 327 ● 321

asked Sep 25, 2017 at 9:27

Khalid Al-Mutawa

869 ● 1 ● 9 ● 17

- 4 Why do you think people aren't using Aurora exactly? Amazon has stated that Aurora is their fastest growing service in terms of user adoption. Maybe you just aren't finding many articles about it because if people were already using MySQL in the past there is no code change necessary to switch to Aurora, so there are no tutorials or articles out there about how to make the switch. Also, Aurora is an RDS database, I edited your question appropriately to clarify.
- Mark B 🌟 Sep 25, 2017 at 15:36

### 3 Answers

Sorted by:

Highest score (default)



56



You should benchmark Aurora carefully before you consider it. Launch an instance and set up a test instance of your application and your database. Generate as high of load as you can. I did at my last company, and I found that despite Amazon's claims of high performance, Aurora failed spectacularly. Two orders of magnitude slower than RDS. Our app had a high rate of write traffic.

Our conclusion: if you have secondary indexes and have high write traffic, Aurora is not suitable. I bet it's good for read-only traffic though.

(Edit: the testing I'm describing was done in Q1 of 2017. As with most AWS services, I expect Aurora to improve over time. Amazon has an explicit strategy of "[Release ideas at 70% and then iterate.](#)" From this, we should conclude that a new product from AWS is worth testing,

but probably not production-ready for at least a few years after it's introduced).

At that company, I recommended RDS. They had no dedicated DBA staff, and the automation that RDS gives you for DB operations like upgrades and backups was very helpful. You sacrifice a little bit of flexibility on tuning options, but that shouldn't be a problem.

The worst inconvenience of RDS is that you can't have a MySQL user with SUPER privilege, but RDS provides stored procs for most common tasks you would need SUPER privilege for.

I compared a multi-AZ RDS instance versus a replica set of EC2 instances, managed by Orchestrator. Because Orchestrator requires three nodes so you can have quorum, RDS was the clear winner on cost here, as well as ease of setup and operations.

[Share](#) [Improve this answer](#)

[edited Jan 12, 2018 at 17:18](#)

[Follow](#)

answered Sep 25, 2017 at 16:17



[Bill Karwin](#)

561k ● 87 ● 698 ● 854

- 
- 1 there is no multi-master option for RDS but there is for aurora  
- that alone should increase performance with writes  
– [Anthony](#) Jul 3, 2019 at 18:45
-

1 @Anthony have you benchmarked writes on a multi-master Aurora instance? – [Bill Karwin](#) Jul 3, 2019 at 18:59

---

5 It will be great if this answer is updated with the latest improvements. – [abbas](#) Jul 7, 2019 at 20:13

---



14



I don't use Aurora personally, but I can HIGHLY recommend RDS over running your own on EC2. Having the failover happen automatically and also the backups is just worth every penny. Especially since RDS isn't that much more expensive.

Aurora looks really good on paper, but the more flexible choice of instances has kept me at PostGreSQL until now. We're looking at migrating to Aurora though, mainly because of the autoscaling storage provisioning and the higher performance.

Share Improve this answer

Follow

edited May 22, 2023 at 19:52



[Matthias](#)

15.2k ● 9 ● 48 ● 64

answered Sep 25, 2017 at 9:32



[Exelian](#)

5,878 ● 1 ● 32 ● 51

---

1 Have you benchmarked Aurora under load-testing with your application? – [Bill Karwin](#) Sep 25, 2017 at 16:08

---

Hmm, interesting. High write traffic is problematic? Because that's our main issue. Will definitely look into this – [Exelian](#) Sep 25, 2017 at 16:10

---

I moved part of my comment into my answer on this thread.  
Also see my answer to  
[stackoverflow.com/questions/46383763/...](https://stackoverflow.com/questions/46383763/) – Bill Karwin Sep 25, 2017 at 16:20

---



5



AWS RDS is the managed database solution which provides support for multiple database options Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle, and Microsoft SQL Server. When you go with RDS it will provide inbuilt configuration options such as.



- Database Replication for High Availability
- Read Replicas for Scalability
- Backups & Restore
- Operating system and software patches & etc.

This simplifies the overhead of database administration. However the flexibility is limited to the RDS offerings.

Alternatively if you host your database in EC2 instance, you can install the required versions of the database engines, install needed extensions & etc. which provides more flexibility but also requires expertise & adds administration overhead.

When you consider Amazon Aurora in RDS, it differs from the rest of the engines because, its new and fully implemented by Amazon from ground up and offers higher performance, reliability out of the box (As marketed by Amazon) with reasonable pricing. However

one limitation with Aurora is that its not included in AWS free-tier, where the smallest instance type it supports is "small".

Note: Some of the features offered by RDS and cost differs, based on the database option you select.

Share Improve this answer

edited Sep 25, 2017 at 15:42

Follow

answered Sep 25, 2017 at 10:19



Ashan

19.7k ● 5 ● 52 ● 68

---

2 Running standard MySQL locally for development, and using Aurora in production is a perfectly fine solution in most cases. I wouldn't consider the inability to run Aurora locally a challenge. – [Mark B](#) 🌟 Sep 25, 2017 at 15:33

---

@MarkB Though Aurora is MySQL compatible, does it mean all the variations in MySQL versions supported? – [Ashan](#) Sep 25, 2017 at 15:34

---

I'm not sure I understand your question. Obviously you would want to run the version of MySQL locally that corresponds to the version of Aurora you are running in production.

– [Mark B](#) 🌟 Sep 25, 2017 at 15:37

---

Ok. Thanks! Will update the answer with information.

– [Ashan](#) Sep 25, 2017 at 15:40

---

1 Aurora/MySQL is was baselined against MySQL Server 5.6.10 -- the GA release of 5.6 -- for feature compatibility. The most appropriate MySQL Server for local development against Aurora would be the latest 5.6.x, since it should

contain all 5.6 bugfixes but not feature changes.

– [Michael - sqlbot](#)  Sep 25, 2017 at 17:05

---

---