Migrating RDS MySQL to Aurora with Read Replica

**SPL-TF-200-DBARAA - Version 1.0.10**

© 2024 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [*AWS Training and Certification*](https://support.aws.amazon.com/#/contacts/aws-training).

**Lab Overview**

This lab gives learners the opportunity to perform a migration from a standard Amazon Relational Database Service (Amazon RDS) MySQL platform to an Amazon Aurora MySQL cluster with even more robust high availability features.

OBJECTIVES

By the end of this lab, you will be able to do the following:

* Create an Aurora read replica cluster.
* Promote the Aurora read replica cluster to a standalone DB cluster.
* Add an additional reader instance to the cluster with highest failover priority.
* Perform a “failover” on the cluster.

TECHNICAL KNOWLEDGE PREREQUISITES

This lab requires:

* A basic knowledge of AWS services as defined in the AWS Cloud Practitioner Essentials course.
* That you should be comfortable with navigating the AWS Management Console.
* A good understanding of SQL databases and running simple SQL queries.

ICON KEY

Various icons are used throughout this lab to call attention to different types of instructions and notes. The following list explains the purpose for each icon:

* **Command:** A command that you must run.
* **Expected output:** A sample output that you can use to verify the output of a command or edited file.
* **Note:** A hint, tip, or important guidance.
* **Additional information:** Where to find more information.
* **Refresh:** A time when you might need to refresh a web browser page or list to show new information.
* **Hint:** A hint to a question or challenge.

**Start lab**

1. To launch the lab, at the top of the page, choose **Start lab**.

**Caution:** You must wait for the provisioned AWS services to be ready before you can continue.

1. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**Warning:** Do not change the **Region** unless instructed.

COMMON SIGN-IN ERRORS

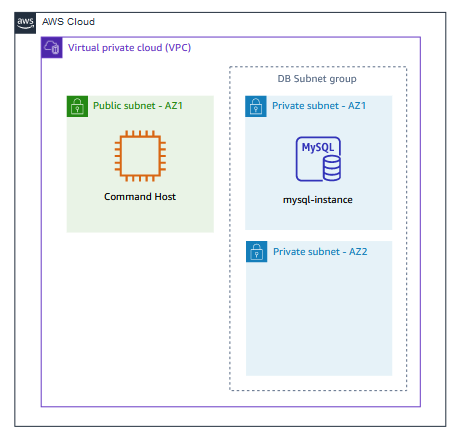
**Error: Choosing Start Lab has no effect**

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

* Add the lab domain name to your pop-up or script blocker’s allow list or turn it off.
* Refresh the page and try again.

LAB ENVIRONMENT

The following diagram shows the basic architecture of the lab environment:



*High-level Architecture diagram of the lab environment. More details in the image description.*

**Image description:** The following resources have been deployed when you log in to your lab environment:

* An Amazon VPC with one public subnet and two private subnets. Each private subnet is in a different availability zone.
* A DB subnet group which contains both private subnets.
* An Amazon Elastic Compute Cloud (Amazon EC2) instance hosted in the public subnet. This instance is referred to as **Command Host** in the lab instructions.
* A MySQL database on Amazon RDS with a single instance named **mysql-instance**. This database is running MySQL version 8.0.35.
  + The instance is hosted in one of the private subnets.
  + The instance is using a security group containing the string **DBsecuritygroup** in its name.
  + The instance contains a **dev** database with a table named **players**. This table has the following schema:

(

player\_num integer,

player\_name text,

player\_pos text,

player\_age integer,

PRIMARY KEY (player\_num)

)

MIGRATING DATA FROM A MYSQL RDS DB INSTANCE TO AURORA

In the lab, you migrate data from the existing MySQL DB instance to an Aurora MySQL DB cluster by using an Aurora read replica. To achieve this, you need to:

* Create an Aurora read replica cluster from the existing MySQL DB instance.
* Promote the new Aurora read replica to a standalone Aurora DB cluster.

**Learn more:** Refer to *Migrating data from a MySQL DB instance to an Amazon Aurora MySQL DB cluster by using an Aurora read replica* in the **Additional resources** section for more information.

**Task 1: Explore the lab environment and connect to the database**

In this task, you explore how to connect to the database, verify the data by running a simple SQL query, insert a new data item to the database, and check your progress. You can always refer back to this task when needed.

DIRECTIONS TO CONNECT TO THE DB

1. Copy the **CommandHostSessionUrl** value that is listed to the left of these instructions. Paste the URL into a new browser tab and press **Enter** to access the command host terminal.
2. **Command:** Run the following command on the command host after replacing the **DBUserName**, **DBUserPasswd**, and **RDSEndpoint** strings with their respective values listed to the left of these instructions. Also note that there is no space between the “-p” parameter and the actual password.

**Note:** You may want to copy the command to your favorite text editor, edit the specific variables, then paste back into the terminal.

mysql -u DBUserName -pDBUserPasswd -h RDSEndpoint

**Expected output:** Your values might differ from what is seen below.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MySQL connection id is 58

Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>

This should log you into the database and give you a prompt where you can enter **Structured Query Language (SQL)** commands used in this lab.

DIRECTIONS TO RUN A SIMPLE QUERY AND VERIFY THE DATA

1. **Command:** To run a simple query to the database and verify the data, use the following SQL command from the **MySQL [(none)]>** prompt:

select \* from dev.players;

**Expected output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

+------------+-------------+------------+------------+

| player\_num | player\_name | player\_pos | player\_age |

+------------+-------------+------------+------------+

| 1 | Higuita | GK | 34 |

| 2 | Moore | DF | 26 |

| 7 | Mbappe | MF | 23 |

| 10 | Pele | FW | 29 |

+------------+-------------+------------+------------+

4 rows in set (0.00 sec)

DIRECTIONS TO INSERT NEW ITEMS TO THE TABLE

1. **Command:** To insert a new item to the table, run the following command at the **MySQL [(none)]>** prompt:

INSERT INTO dev.players

(player\_num, player\_name, player\_pos, player\_age)

VALUES

(14, 'Bergkamp', 'FW', 29);

**Expected output:** Your values might differ from what is seen below.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Query OK, 1 row affected (0.02 sec)

1. **Command:** To verify the new item is inserted successfully, run a simple query to the database by running the following SQL command from the **MySQL [(none)]>** prompt:

select \* from dev.players;

**Expected output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

+------------+-------------+------------+------------+

| player\_num | player\_name | player\_pos | player\_age |

+------------+-------------+------------+------------+

| 1 | Higuita | GK | 34 |

| 2 | Moore | DF | 26 |

| 7 | Mbappe | MF | 23 |

| 10 | Pele | FW | 29 |

| 14 | Bergkamp | FW | 29 |

+------------+-------------+------------+------------+

5 rows in set (0.00 sec)

DIRECTIONS TO EXIT THE DATABASE CONNECTION

1. **Command:** To exit the database connection and return back to the **Command Host** terminal, run the following command at the **MySQL [(none)]>** prompt:

exit

**Expected output:** Your values might differ from what is seen below.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Bye

RESOLVING LAB CHALLENGES AND VERIFYING YOUR PROGRESS

The remaining tasks of the lab provides only high-level instructions with reference to resources that can help you to complete the task. The detailed instructions to complete each task are hidden in a collapsible section at the end of the task and you can expand them to view the step-by-step instructions.

At the end of each task, you can run a Python script from the Command Host that verifies that you have successfully completed each task. The script also returns a score for each task as below:

* Task 2 - 10 points for creating the Aurora read replica cluster and 10 points for data accessibility.
* Task 3 - 10 points for promoting the cluster.
* Task 4 - 10 points for creating the additional reader and 10 points for making it a highest priority for failover.
* Task 5 - 10 points for completing an internal cluster failover.

This gives a possible total of 60 points.

Try running the script before you start task 2.

1. **Command:** To run the verification script, run the following command from the **Command Host** prompt:

python3.11 /temp/verifymigration.py

**Expected output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Got an error while describing cluster: This is likely because you did not create the Aurora read replica cluster correctly.

You will receive a score of 0. An error occurred (DBClusterNotFoundFault) when calling the DescribeDBClusters operation: DBCluster dbaraa-cluster not found.

Your total score is 0

Task 2 score: 0 points for creating the Aurora read replica cluster and 0 points for data accessibility

Task 3 score: 0 points for promoting the cluster

Task 4 score: 0 points for creating the additional reader and 0 points for making it a highest priority for failover

Task 5 score: 0 points for completing an internal cluster failover

The output above is expected as you did not complete any of these tasks yet in the lab.

**Note:** The scoring calculated when running the script above resides inside the temporary lab environment, and hence is not persistent.

 Congratulations! You now have successfully connected to the database, verified the data, inserted new items to the database, and checked your progress.

**Task 2: Create an Aurora read replica cluster**

In this task, you create an Aurora read replica cluster for your source MySQL DB instance.

1. At the top of the AWS Management Console, in the search bar, search for and choose

RDS

.

1. In the navigation pane at the left of the page, choose **Databases**.

The **mysql-instance** listed under the **Databases** section is the MySQL DB instance that you intend to migrate to Aurora.

**High-level instructions**

1. Use the AWS Management Console to create the Aurora read replica cluster. The new cluster should have the following settings:

* The instance identifier is

dbaraa

* High availability is enabled (Multi-AZ)
* Must use the same VPC and Security Group as the existing RDS instance
* The instance type/size is the same as the existing RDS instance
* Must use Aurora
* Use the latest available Aurora version that is compatible with MySQL 8.0
* The database instances in the cluster must use TCP port 3306 for the application connections.
* For lab purpose, disable detailed monitoring

**Learn more:** Refer to *Migrating data from a MySQL DB instance to an Amazon Aurora MySQL DB cluster by using an Aurora read replica* in the **Additional resources** section for more information.

**Note:** Creating the read replica can take up to 45 minutes. While you wait, consider viewing some of the additional resources found at the end of this lab.

Once completed, you should see two instances under the **dbaraa-cluster** with different roles: **Writer instance** and **Reader instance**. Each one has a different endpoint name.

**Refresh:** You may need to refresh the browser page to retrieve the latest status of the instances.

To verify that the new Aurora cluster is available and operational, you can run the following checks:

* Connect to the new Aurora cluster **writer** endpoint from the **Command Host** and insert the following item:

INSERT INTO dev.players

(player\_num, player\_name, player\_pos, player\_age)

VALUES

(12, 'Cafu', 'DF', 25);

* Connect to the Aurora cluster **reader** endpoint from the **Command Host** and run a data query. You should be able to read the players data successfully including the the new item inserted above.
* Verify your progress by running the Python verification script. At this stage, the script should indicate that you have successfully completed Task 2 and scored 20 points.

**Hint:** Refer to **Task 1** if you need help on how to connect to the database, insert a new data item, run a query, or verify your lab progress.

**Detailed instructions**

Expand the *Detailed instructions* below for the full solution.

**Detailed instructions**

 Congratulations! You now have successfully created the Aurora read replica cluster from your source MySQL DB instance and verified write and read operations.

**Task 3 Promote the Aurora read replica cluster to a standalone DB cluster**

In this task, you promote Aurora read replica to a standalone DB cluster and verify that you can use the database (add items and view contents of the table) without having lost any data.

**High-level instructions**

1. Use the AWS Management Console to promote the Aurora read replica cluster.

**Learn more:** Refer to *Migrating data from a MySQL DB instance to an Amazon Aurora MySQL DB cluster by using an Aurora read replica* in the **Additional resources** section for more information.

Once you completed the cluster promotion, you can verify it by running the following checks:

* Connect to the Aurora cluster **Writer instance** endpoint from the **Command Host** and insert the following item:

INSERT INTO dev.players

(player\_num, player\_name, player\_pos, player\_age)

VALUES

(13, 'Matthaus', 'MF', 30);

* Connect to the Aurora cluster **Reader instance** endpoint from the **Command Host** and run a data query. You should be able to read players data successfully including the the new item inserted above.
* Verify your progress by running the Python verification script. At this stage, the script should indicate that you have successfully completed Tasks 2 and 3 and scored 30 points.

**Hint:** Refer to **Task 1** if you need help on how to connect to the database, insert a new data item, run a query, or verify your lab progress.

**Detailed instructions**

Expand the *Detailed instructions* below for the full solution.

**Detailed instructions**

 Congratulations! You now have successfully promoted Aurora read replica to a standalone DB cluster and verified write and read operations.

**Task 4 Add an additional reader instance to the Aurora cluster with highest failover priority**

In this task, you add an extra reader instance to the Aurora cluster and set it with the highest failover priority. This provides extra resiliency to the solution.

**High-level instructions**

1. Use the AWS Management Console to add an additional reader instance. Set the new reader instance as below:

* Instance identifier:

aurora-reader-instance

.

* Instance type and size: **db.t3.medium**.
* Make this instance the highest priority for any cluster failover.

**Learn more:**:

* Refer to *Adding Aurora Replicas to a DB cluster* in the **Additional resources** section for more information.
* Refer to *High availability for Amazon Aurora* in the **Additional resources** section for more information.

Once the new reader instance is available, you can verify it by running the following checks:

* Connect to the new reader instance endpoint and perform read operations on the database. The data should be synchronized across all replicas in the cluster.
* Verify your progress by running the Python verification script. At this stage, the script should indicate that you have successfully completed Tasks 2, 3, and 4 and scored 50 points.

**Hint:** Refer to **Task 1** if you need help on how to connect to the database, run a query, or verify your lab progress.

**Detailed instructions**

Expand the *Detailed instructions* below for the full solution.

**Detailed instructions**

 Congratulations! You now have successfully added an extra reader instance to the Aurora cluster and set it with the highest failover priority.

**Task 5 Perform a “failover” on the cluster**

In this task, you perform a failover on the cluster.

**High-level instructions**

1. Use the AWS Management Console to manually failover the **dbaraa-cluster Writer instance** so that the instance you manually added in the previous task becomes the primary instance. Verify that you can use the database (add items and view contents of the table) without having lost any data.

**Learn more:** Refer to *Failover with Amazon Aurora PostgreSQL* in the **Additional resources** section for more information.

1. Verify that you can connect to the new reader instance endpoint and perform read operations on the database. The data should be synchronized across all replicas in the cluster.
2. Verify your progress by running the Python verification script. At this stage, the script should indicate that you have successfully completed Tasks 2, 3, 4, and 5 and scored 60 points.

**Hint:** Refer to **Task 1** if you need help on how to connect to the database, run a query, or verify your lab progress.

**Detailed instructions**

Expand the *Detailed instructions* below for the full solution.

**Detailed instructions**

 Congratulations! You now have successfully performed a failover on the cluster.

**Conclusion**

 Congratulations! You now have successfully:

* Created an Aurora read replica cluster.
* Promoted the Aurora read replica cluster to a standalone DB cluster.
* Added an additional reader instance to the cluster with highest failover priority.
* Performed a “failover” on the cluster.

**End lab**

Follow these steps to close the console and end your lab.

1. Return to the **AWS Management Console**.
2. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
3. Choose **End lab** and then confirm that you want to end your lab.

**Additional Resources**

* [Migrating data from a MySQL DB instance to an Amazon Aurora MySQL DB cluster by using an Aurora read replica](https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Migrating.RDSMySQL.Replica.html).
* [Adding Aurora Replicas to a DB cluster](https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/aurora-replicas-adding.html).
* [High availability for Amazon Aurora](https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/Concepts.AuroraHighAvailability.html).
* [Failover with Amazon Aurora PostgreSQL](https://aws.amazon.com/blogs/database/failover-with-amazon-aurora-postgresql/).
* [Upgrade Amazon Aurora MySQL Version 1](https://www.youtube.com/watch?v=R5XMny416vo&list=PLhr1KZpdzukdT09ev-lchzwrgJo33l8RX&index=7)
* [High Availability and Failover in Amazon Aurora](https://www.youtube.com/watch?v=ERMHycDc8ck&list=PLhr1KZpdzukdT09ev-lchzwrgJo33l8RX&index=47)

For more information about AWS Training and Certification, see [*https://aws.amazon.com/training/*](https://aws.amazon.com/training/).

*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [*AWS Training and Certification Contact Form*](https://support.aws.amazon.com/#/contacts/aws-training).