

Overview

Mysql Database Migration Recommendation

Azure Database for MySQL provides fully managed, enterprise-ready community MySQL database as a service for app development and deployment. Being MySQL community edition, this service allows you to easily lift and shift to the cloud and use languages and frameworks for your choice. On top of that, you get built-in high availability and capability to scale in seconds, helping you easily adjust to changes in customer demands. Additionally, you benefit from the unparalleled security and compliance, including Azure IP advantage, as well as Azure's industry leading reach with more datacenters than any other cloud provider. All this with a flexible pricing model so you can choose resources for your workload with no hidden cost.

Database Details		Azure Infrastructure Cost
Project Name	SQL	Azure Database for MySQL Gen 5, 4 vCore (USD \$ 255.79)
Database Name	firmdeals	MySQL running on VM F4als v6 (USD \$ 310.98)
DB Version	11.2.5-MariaDB-ubuntu2204	
DB Engine	MyISAM	

Recommendations

Datapoint Name

DB Engine

Reason of Change

Azure provides innodb database engine, so if an application uses myisam database engine, then that will not be supported by Azure. InnoDB has row-level locking for better reliability and scalability. InnoDB has what is called referential integrity, which involves supporting foreign keys(RDBMS) and relationship constraints. MyISAM does not have referential integrity. (DMBS).

Recommendation

Innodb is recommended for large database because it provides a transaction safe (ACID compliant) storage engine for MySQL that has commit, rollback and crash recovery capabilities to protect user data. Innodb row-level locking and oracle-style consistent non-locking reads increase multiuser concurrency and performance. Mylsam doesn't provide data integrity and transaction, and it is less secure than innodb, generally Mylsam is used for smaller database that is the reason innodb is supported by azure cloud not Mylsam.

Convert Storage Engine via Terminal

Run the **ALTER TABLE** command in the MySQL shell to convert the storage engine from MyISAM to InnoDB.

- To convert MyISAM to InnoDB, run:

```
ALTER TABLE database_name.table_name ENGINE=InnoDB;
```

Impact

Mandatory

Estimated Hours

4 Hours task for one developer

Help URL

<https://dev.mysql.com/doc/refman/5.6/en/converting-tables-to-innodb.html>

Datapoint Name

DB Version

Reason of Change

Azure supports innodb storage engine for MySQL, that is supported by MySQL 5.7 and later versions. If an application uses a Prior version of MySQL 5.7, it will not support innodb as their default storage engine is MyISAM.

Recommendation

Use MySQL 5.7 , 8.0 or 8.4(LTS) for improved performance nicely for multi-threaded workloads. And Read/Write and Read Only workloads testing requests per second show a marked improvement for both MySQL 5.7,8.0 over MySQL 5.1.

There are two approaches for upgrading MySQL on Windows:

- Using MySQL Installer
- Using the Windows ZIP archive distribution

To perform an upgrade using **MySQL Installer**:

1. Start MySQL Installer.
2. From the dashboard, click **Catalog** to download the latest changes to the catalog. The installed server can be upgraded only if the dashboard displays an arrow next to the version number of the server.
3. Click **Upgrade**. All products that have a newer version now appear in a list.
4. Deselect all but the MySQL server product, unless you intend to upgrade other products at this time, and click **Next**.
5. Click Execute to start the download. When the download finishes, click Next to begin the upgrade operation.
6. Configure the server.

To perform an upgrade using the Windows ZIP archive distribution:

1. Download the latest Windows ZIP Archive distribution of MySQL from this

[link](#)

2. If the server is running, stop it. If the server is installed as a service, stop the service with the following command from the command prompt:
 - C:\> SC STOP mysqld_service_name
3. Extract the ZIP archive. You may either overwrite your existing MySQL installation (usually located at **C:\mysql**), or install it into a different directory, such as C:\mysql8. Overwriting the existing installation is recommended.
4. Restart the server. For example, use the SC START mysqld_service_name or NET START mysqld_service_name command if you run MySQL as a service, or invoke mysqld directly otherwise.
5. Prior to MySQL 8.0.16, run mysql_upgrade as Administrator to check your tables, attempt to repair them if necessary, and update your grant tables if they have changed so that you can take advantage of any new capabilities. See Section 4.4.5, “mysql_upgrade — Check and Upgrade MySQL Tables”. As of MySQL 8.0.16, this step is not required, as the server performs all tasks previously handled by mysql_upgrade.
6. If you encounter errors, see

[Section 2.3.5, “Troubleshooting a Microsoft Windows MySQL Server Installation”](#)

Impact

Mandatory

Estimated Hours

4 Hours task for one developer

Help URL

<https://dev.mysql.com/doc/refman/8.0/en/windows-upgrading.html>