MariaDB

CLOUD DATABASE COMPARISON: SKYSQL VS. GOOGLE CLOUD SQL

WHITEPAPER

TABLE OF CONTENTS

3 INTRODUCTION

4 SCALABILITY

6 HIGH AVAILABILITY

7 DEVELOPMENT

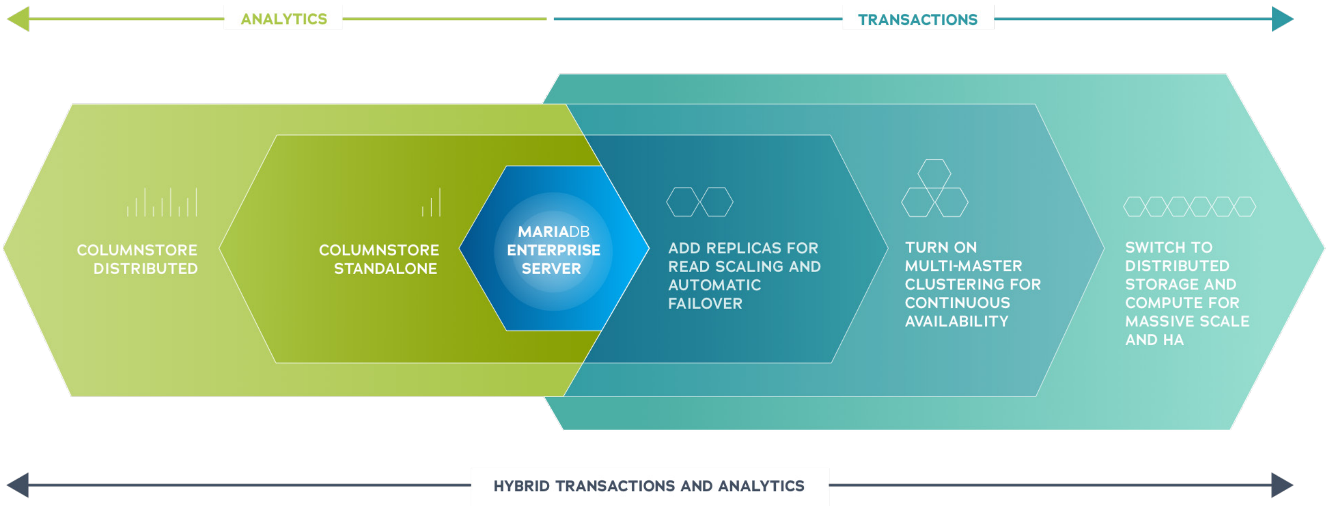
8 SUMMARY

9 CONCLUSION

INTRODUCTION

Database-as-a-Service (DBaaS) is one of the fastest-growing cloud markets. According to Gartner, 75% of all databases will be running in the cloud by 2022.[1] The first relational DBaaS offerings were created by cloud service providers – Amazon RDS, Microsoft Azure Database and Google Cloud SQL. These services started with MariaDB, MySQL and/or Postgres, all open source, and made using relational databases in the cloud a lot easier.

SkySQL is a DBaaS built to unlock the full power of MariaDB Platform in the cloud. It combines innovative use of public cloud infrastructure, including Kubernetes services such as Amazon EKS and Google Kubernetes Engine, with the knowledge of MariaDB's expert support and remote DBA teams. SkySQL can deploy MariaDB Platform as a standard transactional database, a data warehouse, a distributed SQL database or a hybrid transactional/analytical database (HTAP) – all with the click of a button. It supports Amazon Web Services (AWS) and Google Cloud Platform (GCP), with Microsoft Azure coming soon.


ANALYTICS
TRANSACTIONS
COLUMNSTORE DISTRIBUTED
COLUMNSTORE STANDALONE
MARIADB ENTERPRISE SERVER
ADD REPLICAS FOR READ SCALING AND AUTOMATIC FAILOVER
TURN ON MULTI-MASTER CLUSTERING FOR CONTINUOUS AVAILABILITY
SWITCH TO DISTRIBUTED STORAGE AND COMPUTE FOR MASSIVE SCALE AND HA
HYBRID TRANSACTIONS AND ANALYTICS

Cloud SQL supports MySQL and Postgres, is limited to standard transactional databases and is only available on GCP. Like Amazon RDS and Azure Database, it makes running relational databases in the cloud a lot easier, but as businesses begin to expand the scope of their cloud initiatives to include mission-critical applications, ease of use is no longer enough. It is important to consider operational aspects such as scalability and high availability as well as the impact on application development when running production databases in the cloud.

This white paper compares SkySQL with Cloud SQL for MySQL. SkySQL is compatible with applications using standard MariaDB/MySQL drivers because its underlying database, MariaDB Platform, implements the MySQL protocol. In particular, this white paper focuses on the most important differences between SkySQL and Cloud SQL.

[1] https://www.gartner.com/en/newsroom/press-releases/2019-07-01-gartner-says-the-future-of-the-database-market-is-the

SCALABILITY

Cloud SQL deploys databases in a traditional primary/replica configuration, with a read/write primary and zero or more read-only replicas. If higher write throughput is required, the only option is to scale up (i.e., move to a bigger and more expensive instance type). However, once the largest instance type is used, write throughput is effectively capped. There is no way to increase it.

In terms of reads, having enough memory to cache the working set results in the best performance. However, as with writes, once the largest instance is used and the entire working set exceeds the amount of available memory, read performance suffers and there is no way to fix it.

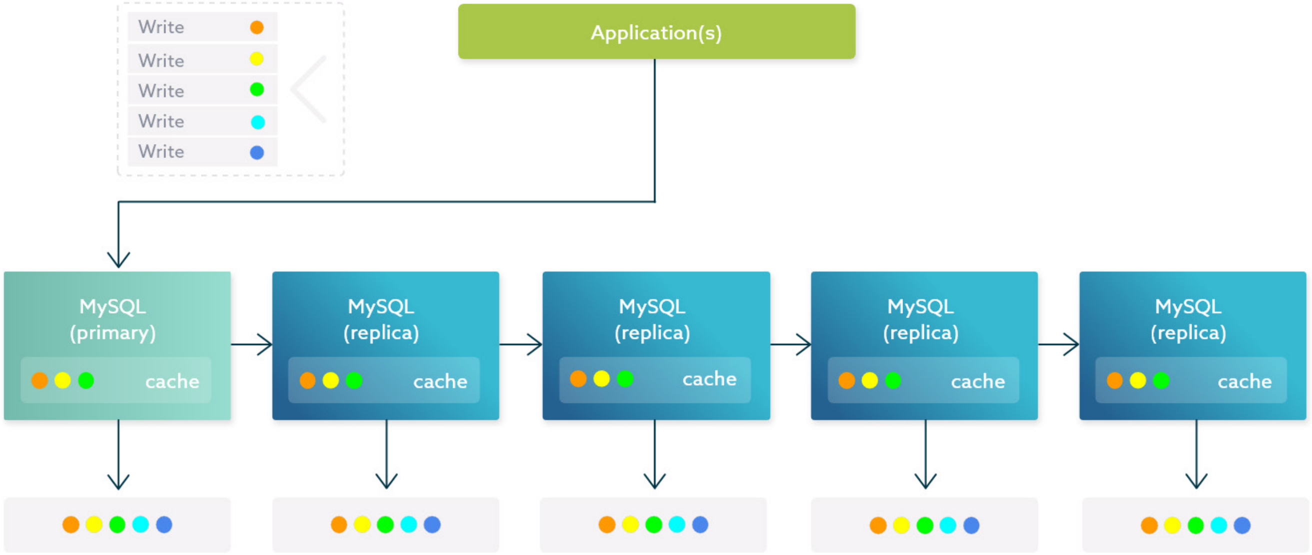


Figure 1: Cloud SQL primary/replica topology

SkySQL can deploy MariaDB in one of three ways for transactional workloads: a traditional primary/replica configuration, a multi-master configuration for continuous availability and a distributed SQL configuration for high throughput and elastic scalability.

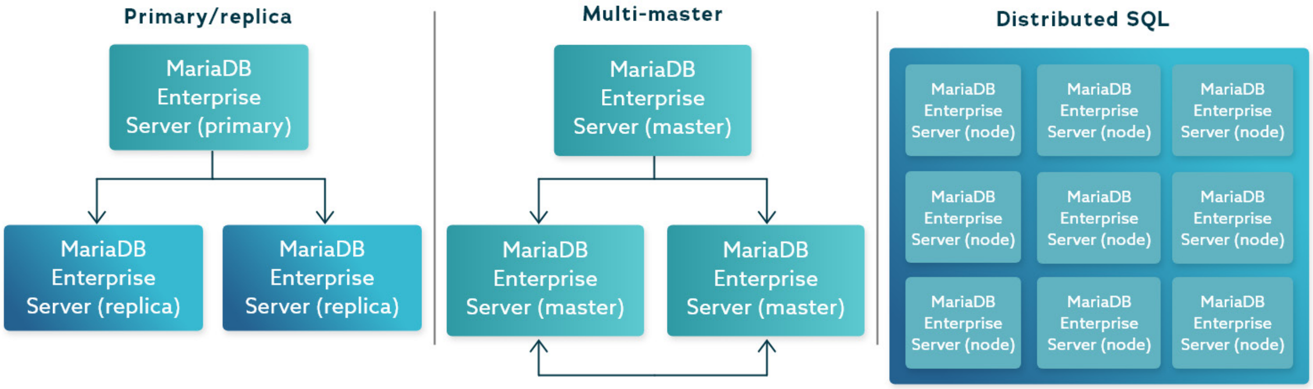

Primary/replica
MariaDB Enterprise Server (primary)
MariaDB Enterprise Server (replica)
MariaDB Enterprise Server (replica)
Multi-master
MariaDB Enterprise Server (master)
MariaDB Enterprise Server (master)
MariaDB Enterprise Server (master)
Distributed SQL
MariaDB Enterprise Server (node)

Figure 2: SkySQL transactional topologies

In a distributed SQL configuration, data is partitioned with different partitions sitting on different database instances. It is read from/written to using distributed transactions and queries spanning one or more database instances. It uses the disks of multiple database instances to increase disk I/O and thus write throughput. When more database instances are added, more disks are used and write throughput increases. In terms of reads, it effectively combines the memory of multiple database instances to create a single large cache. As a result, each database instance only needs enough memory to cache its part of the working set rather than the entire working set, allowing for the use of smaller instance types. When more database instances are added, the size of the cache is increased and more data can be cached in memory (for the fastest reads possible).

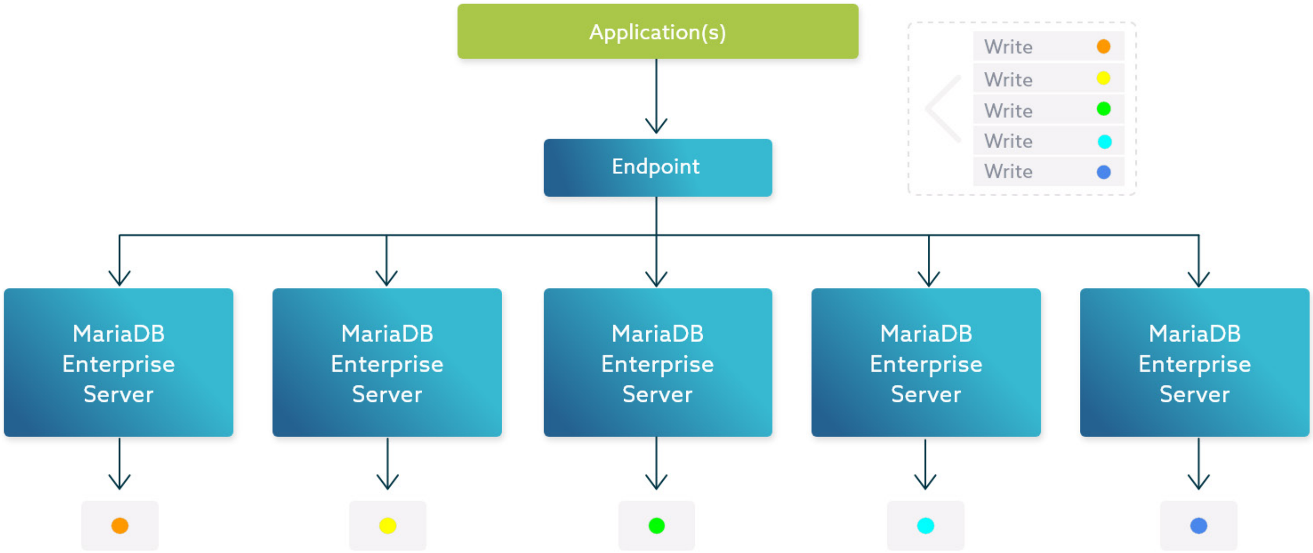

Application(s)
Endpoint
Write
Write
Write
Write
Write
MariaDB Enterprise Server
MariaDB Enterprise Server
MariaDB Enterprise Server
MariaDB Enterprise Server
MariaDB Enterprise Server

Figure 3: SkySQL write scaling with distributed SQL

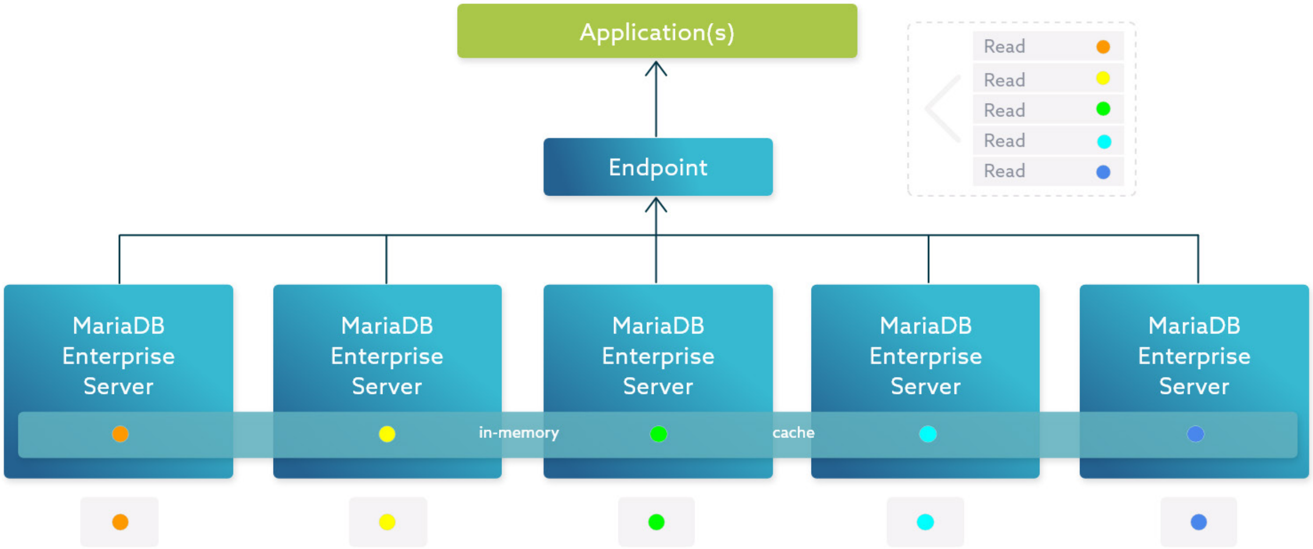

Application(s)
Endpoint
Read
Read
Read
Read
Read
MariaDB Enterprise Server
MariaDB Enterprise Server
MariaDB Enterprise Server
MariaDB Enterprise Server
MariaDB Enterprise Server
in-memory
cache

Figure 4: SkySQL read scaling with distributed SQL

HIGH AVAILABILITY

Cloud SQL provides high availability with two database instances in a primary/standby configuration sharing an IP address. The primary instance handles reads and writes while the standby remains idle. If the primary database instance fails, application connections to the primary and its replicas, if any, will be closed and an automatic failover process will swap out the failed primary for its standby.

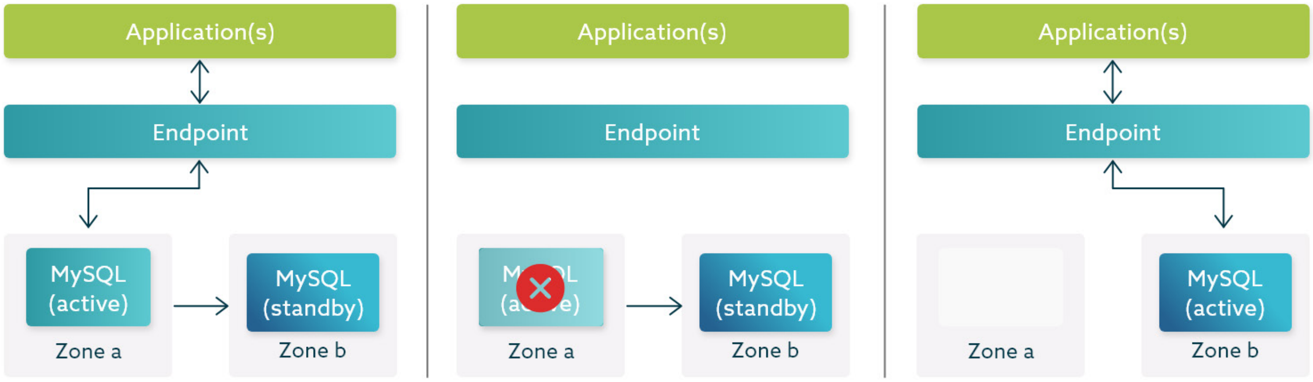
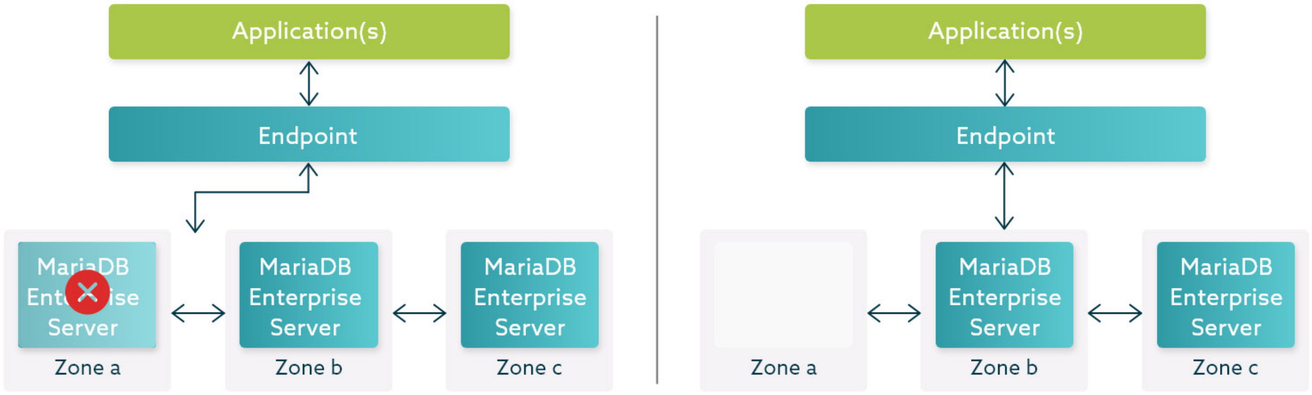


Figure 5: Cloud SQL automatic failover with an active/standby (i.e., HA) configuration

Note:
According to Cloud SQL documentation, it takes 2-3 minutes before applications can create new connections after an automatic failover.

Alternatively, SkySQL automatic failover with primary/replica deployments often takes less than one second. With multi-master clusters and distributed SQL databases, automatic failover is not needed because reads and writes can be executed on any database instance. If a database instance fails within a multi-master deployment, any failed queries will automatically be retried on a different one and subsequent queries will no longer be routed to it. In distributed SQL deployments, queries will no longer be routed to a failed database instance, and because there are multiple copies of data for redundancy, each stored on a different database instance, there is no data loss. In addition, if a database instance fails, any lost copies of data are automatically recreated in order to restore full fault tolerance.



Application(s)
Endpoint
MariaDB Enterprise Server
Zone a
MariaDB Enterprise Server
Zone b
MariaDB Enterprise Server
Zone c
Application(s)
Endpoint
Zone a
MariaDB Enterprise Server
Zone b
MariaDB Enterprise Server
Zone c

Figure 6: SkySQL automatic failover with a multi-master configuration

Further, transaction replay can be enabled to completely hide database instance failures. Instead of requiring applications to create new connections and retry failed in-flight transactions, their connections remain open and in-flight transactions are automatically recovered and migrated to a different database instance where they can continue.

DEVELOPMENT

Cloud SQL does not provide load balancing. Rather, it requires applications to use a separate connection for each replica and round-robin reads across them. Further, if applications have reads and writes, and read scaling is needed, developers must implement read/write splitting in order to ensure writes are executed on the primary and reads on the replicas.

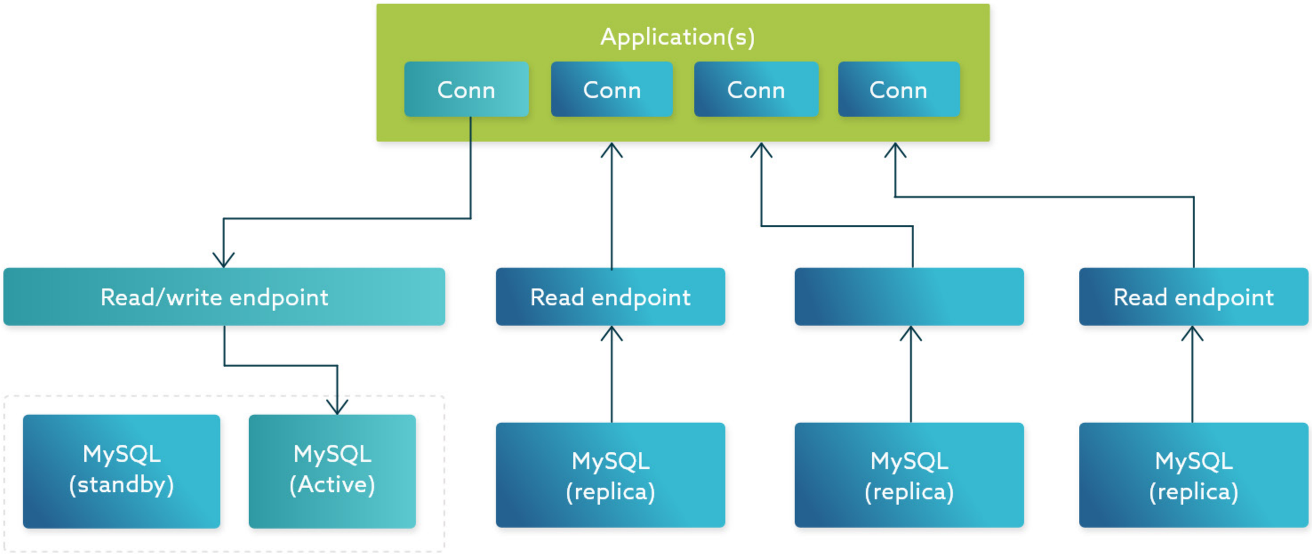
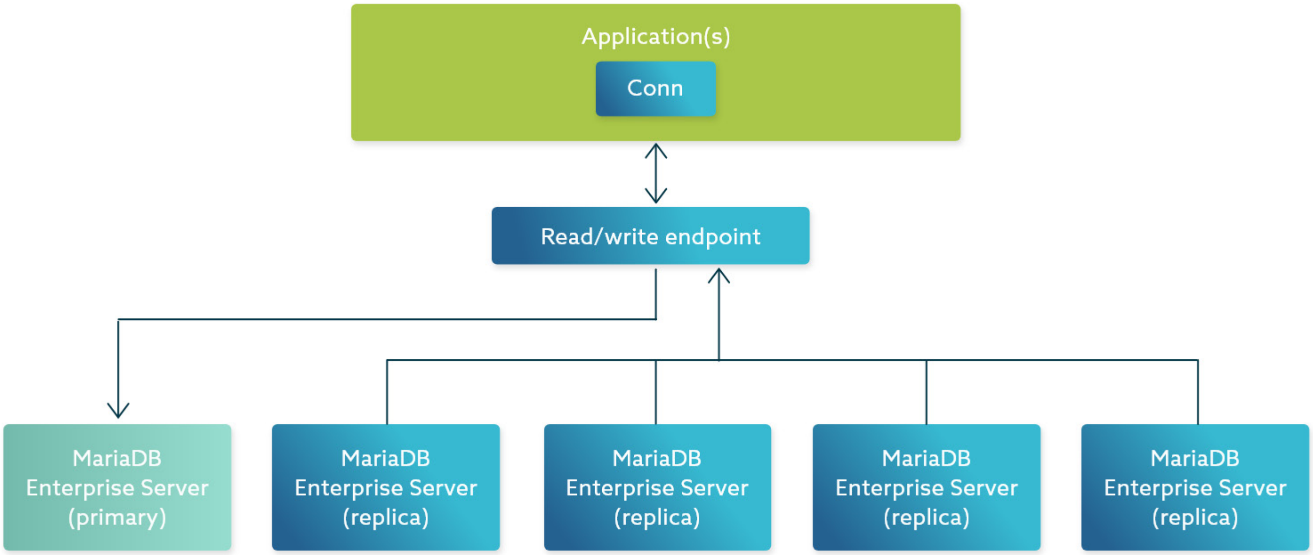


Figure 7: Cloud SQL with application-side load balancing and read/write splitting

SkySQL has built-in load balancing, providing applications with a single endpoint and automatically distributing read requests across multiple database instances. As a result, developers don't have to worry about using separate database



Application(s)
Conn
Read/write endpoint
MariaDB Enterprise Server (primary)
MariaDB Enterprise Server (replica)
MariaDB Enterprise Server (replica)
MariaDB Enterprise Server (replica)
MariaDB Enterprise Server (replica)

Figure 8: SkySQL with built-in load balancing and read/write splitting

instance connections and implementing load balancing themselves. They don't have to know about the physical database infrastructure – the number of replicas, their IP address or if any have been added or removed.

In addition, SkySQL has built-in read/write splitting for standard transactional databases, automatically routing writes to the primary and load balancing reads across replicas.

SkySQL not only makes it easier to run a relational database in the cloud, it makes application development easier too, abstracting away the underlying physical database infrastructure so primary/replica, multi-master and distributed SQL databases look no different than a standalone database.

SUMMARY

		Cloud SQL	SkySQL
Scalability	Read scaling	Yes	Yes
Scalability	Write scaling	No	Yes
High availability	Automatic failover	Yes	Yes
High availability	Multi-master clustering	No	Yes
High availability	Connection migration	No	Yes
High availability	Session restore	No	Yes
High availability	Transaction replay	No	Yes
High availability	Max database instance failures	1	n-1
High availability	Max zone failures	1	2
Development	Single endpoint	No	Yes
Development	Load balancing	No	Yes
Development	Read/write splitting	No	Yes

CONCLUSION

SkySQL and Cloud SQL both make it easy to run MySQL-compatible databases in the cloud. However, as businesses begin moving mission-critical applications to the cloud, they need a DBaaS capable of providing the same level of scalability and high availability as enterprise on-premises databases while at the same time simplifying application development.

SkySQL, with its support for distributed SQL, provides businesses with unlimited scalability, allowing them to start small and grow as big as needed and as fast as needed, paying only for what they need, when they need it. And with advanced features such as transaction replay, developers and DBAs alike can rest easy knowing database instance failures (or even entire zone failures) will not interrupt production applications, allowing them to continue as though nothing happened.

Get started with SkySQL today by taking advantage of a $500 free credit.

mariadb.com

Americas: sales-AMER@mariadb.com
Europe, Middle East, Africa: sales-EMEA@mariadb.com
Asia Pacific: sales-APAC@mariadb.com