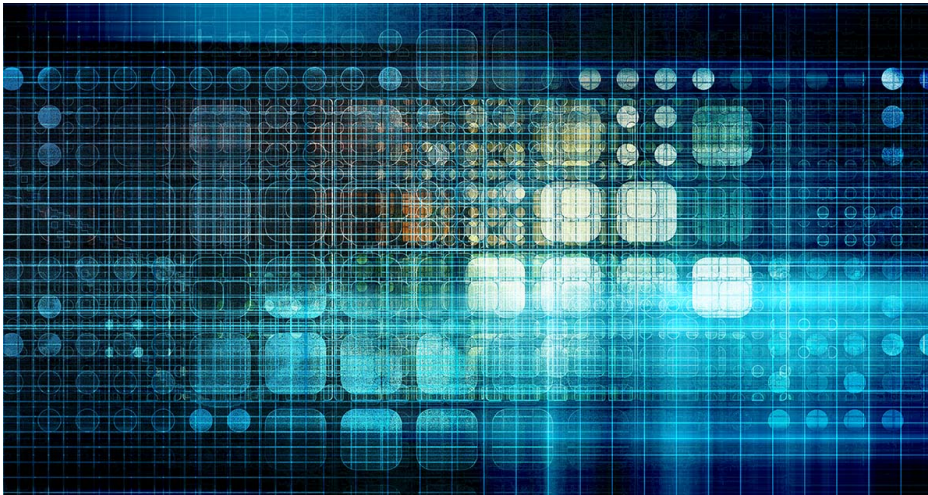


🕒 10.03.20 | Technical matters

How did you like the article? 4 ★★★★★ (6) [f](#) [t](#) [x](#) [in](#)



MariaDB vs. MySQL

MySQL or MariaDB? Which [database management system](#) takes the upper hand? The open source project **MySQL** prides itself in being on the market for over 20 years. **MariaDB**, its so-called “fork,” first came onto the scene in 2009. It initially focused on the development of a drop-in replacement with high open source community standards. Currently, however, both projects are pursuing very different developmental objectives. Find out more about them in this article.

Related Products



[See packages ▶](#)

Web hosting for agencies

Provide powerful and reliable service to your clients with a web hosting package from IONOS.

[View packages ▶](#)

Contents +

1. [Development of MySQL und MariaDB](#)
2. [MySQL and MariaDB: a comparison](#)
3. [MariaDB vs. MySQL – comparing the most important features](#)
4. [Summary](#)

Development of MySQL und MariaDB

Finnish software engineer **Michael “Monty” Widenius** is the key figure behind the development of both MySQL and MariaDB. In 1995, together with the Swedish duo of David Axmark and Alan Larsson, Widenius formed his stock company **MySQL AB**, which (only one year after releasing the project for the first time) published its first open source database management system (DBMS) called MySQL. This consequently gave rise to their worldwide success. The system reached its peak in 2008 when it was sold to **Sun Microsystems** for 1 billion dollars.

In 2009, Widenius left the software project together with other key MySQL developers and immediately began laying foundations for **MySQL’s new project fork – MariaDB**. The reason for Widenius’ break-up with his previous project was the imminent take-over by Sun Microsystems lead by the highly controversial software giants Oracle in 2010. Widenius’ development team feared that Oracle would get into a conflict of interest with them if the giants were to redevelop the open

Popular Articles

Create a blog

This guide shows you how to start blogging with success in a few simple steps...

Personal email domains

Find out about the advantages of an individual domain and learn how to secure your own in just a few steps...

Can you make money with domains?

A high profit can be made with domain trading! We show you what aspects to

source software – MySQL – as well as any proprietary database software.

According to Widenius, MariaDB should:

- Keep talented team members contributing to the rise of MySQL under one roof
- Promote the continuation of community development engines
- Make sure that a free version of MySQL is always available to all users

It is for this reason that Widenius formed the [MariaDB-Foundation](#) in 2012, the establishment of which has been made possible by Monty Program AB – a company formed in 2009 by Widenius himself. In 2014, MariaDB Foundation fused with SkySQL, subsequently forming the **MariaDB Corporation**.

Fact

When naming his projects, Michael “Monty” Widenius always drew inspiration from his children. The prefix “My” in MySQL corresponds to the real life name of his eldest daughter, the name of MySQL’s fork to his youngest daughter Maria. Max, the name of his only son, can be found in database management systems such as MaxDB or MaxScale.

consider when trying your hand at this...

The best online backup services

We compare the 7 best online data backup providers...

Create a bootable USB

A bootable USB drive will get your PC up and running again after a system problem. We show you how...

During the initial development phase of **MariaDB and its servers** (the community project of the MariaDB Foundation), the main figures of the MariaDB Corporation took care of its commercial business units. MariaDB Corporation simply wanted to distance itself from a situation in which it would be managed by the same administrative entity as Oracle. This split was taken rather positively by the open source community in question.

MariaDB's main focus is the **open development system**, to which there are two main principles:

- The source code of the program parts under development is made available by means of public repositories
- All steps taken during the development stage are made public

Contrary to this, Oracle's dual licensing system has been strongly criticized by the open source community. The criticism is based on the increasing differences between the open source community version and the proprietary MySQL products. In the past, what has also received its fair share of criticism is the ambiguous manner in which software errors were dealt with.

Timeline presentation of MySQL's and MariaDB's progress

1995	First international release of the open source DBMS MySQL thanks to MySQL AB
2008	MySQL is sold to Sun Microsystems

2009	Release of MySQL fork project – MariaDB – thanks to Monty Program AB
2010	Sun Microsystems is taken over by Oracle
2012	MariaDB-Foundation is established
2014	Fusion of Monty Program AB with SkySQL – MariaDB Corporation is established as a result

Tip

Create a safe "home" for your web projects with [SQL server hosting](#) from IONOS.

MySQL and MariaDB: a comparison

The database management systems of MySQL and MariaDB are, in their essence, based on the same software solution. Although MariaDB is a fork (in other words, a spin-off version) of MySQL 5.1, over the course of time, it has evolved into a stand-alone database management system. Here, we present the main differences of the current GA (general availability) versions of MySQL 8.0.11 and Maria 10.2.8. The following points will be discussed:

- Database structure
- Database engines
- Database queries
- Performance
- Availability
- Safety of use
- Ecological compatibility
- Support
- Documentation and community

Database structure

The development of MariaDB has been based on the need to guarantee its users a **drop-in-replacement** for MySQL which would be fully compatible with the database structure, APIs, and configuration data. Upgrading from MySQL to MariaDB or vice versa should be as easy as upgrading from one version of MySQL to another. MariaDB's development team have managed to achieve this objective up to and including the 7th version of MySQL.

Essentially, the same database management systems are used for both database structures in question. Both comply with the [relational database model](#). Their data and table definitions are compatible, while identical protocols, structures, and programming interfaces (API) also apply to both. All MySQL connectors can be used with MariaDB without the need to modify them in such a way that applications and tools of standard database interfaces such as ODBC or JDBC are interlinked. MariaDB's command line tools such as *mysqldump* or *mysqladmin* are also valuable during administrative tasks and the creation of backups.

To completely eradicate any compatibility issues, the development team of MariaDB performs monthly synchronizations of the fork's source code with that of MySQL. The **binary compatibility** of both database management systems has however come to an end with the release of MySQL 8. From this version, MySQL is no longer downwardly compatible neither with any of its older versions nor with MariaDB.

New transactional [data dictionary of MySQL 8](#) proves that the development of the open source projects in question has diverged into two paths. This is part of the metadata processing strategy, which differs substantially to any of its previous concepts. Since MariaDB has not come up with a similar feature so far, it is safe to assume that a data file-oriented compatibility will not be available in the near future between the two systems.

! Note

MariaDB was initially a binary-compatible substitute for MySQL. However, since MySQL's release of its 8th version, this is no longer the case.

Database engines

For reasons motivated by the need for flexibility, MariaDB intends to move away from MySQL in the near future. Aside from standard

engines supported by MySQL, users would also be able to avail of a **large number of alternative database engines** for special application scenarios.

! Note

Database engines are all about storage subsystems, which enable the creation, gathering, update, and erasure of data in database tables by various database management systems. See [CRUD](#) for a better understanding.

The following table shows what database engines are available for MySQL 8.0.11 and MariaD 10.3 (as documented by the suppliers themselves).

Database engine	MySQL 8.0.11	MariaDB 10.3.8	Specification
InnoDB/ XtraDB	✓	✓	InnoDB is the standard storage subsystem of MySQL (version 5.5 onwards). Up until version 10.1, MariaDB availed of InnoDB's fork – XtraDB – as the standard database engine. Ever since the release of

			version 10.2, InnoDB is also used as the only storage subsystem of MariaDB. It offers transaction-oriented read and writing access.
MyISAM	✓	✓	MyISAM – the standard storage subsystem of old MySQL versions – offers quick access to database tables and indices.
MEMORY	✓	✓	MEMORY is a storage subsystem for temporary data. As table definitions remain on the hard drive, data is retained on the working memory. MEMORY is suited for in-memory databases with quick access times and low latency.
CSV	✓	✓	This engine saves data as data fields in CSV format separated by commas.
Archive	✓	✓	This database engine is designed for long-term storage of large amounts of data. It offers various compression algorithms enabling a space-saving data retention.

BLACKHOLE	✓	✓	BLACKHOLE serves as an inspector of SQL Statements. Instead of saving data, BLOCKHOLE simply logs any SQL syntax elements.
Merge	✓	✓	Merge, as its name already suggests, facilitates the merging of numerous MyISAM tables with identical structures into one, joint table.
Federated/ FederatedX	✓	✓	Federated lets MySQL access tables from other servers via remote configuration. MariaDB avails of its fork: FederatedX.
ColumnStore	✗	✓	ColumnStore is all about transferring InfiniDB , which allows for a column-based processing of petabyte-sized data.
Aria	✗	✓	Since 2007, Aria has grown into a crash-proof alternative to MyISAM.
Cassandra	✗	✓	Cassandra offers MariaDB users a database engine managing large, structured sets of data. This software follows a non-relational approach

			(NoSQL) and is designed to provide high availability and safety against any failures (i.e. without any single point of failure).
CONNECT	✗	✓	The CONNECT storage engine facilitates local and remote access to various external data sources such as Dbase, CSV, DOS, FMT, and XML.
Mroonga	✗	✓	Mroonga is a column-based database engine which offers a full-text search in diverse languages including Chinese, Japanese, and Korean.
MyRocks	✗	✓	MyRocks has been created for the purpose of quick, low-latency data storages and offers a distinctly improved compression than that of InnoDB.
OQGRAPH	✗	✓	The open query computation engine GRAPH (OQGRAPH for short) processes data from hierarchical database structures into complex graphs.

Sequence	✗	✓	This storage engine lets its users generate ascending and descending numerical sequences of positive integers including initial and end values as well as increments.
SphinxSE	✗	✓	Sphinx Storage Engine (SphinxSE for short) is an alternative option to the integrated full-text searches offered by MariaDB. This is made possible by means of Sphinx's search daemon <i>searchd</i> .
Spider	✗	✓	Spider is a database engine with an integrated sharding function (partitioning). Spider supports database partitions and eXtended Architecture (XA). In this way, various resources are able to interact with the same transaction. Tables in different database systems are therefore treated as if they belonged to the same system.
TokuDB	✗	✓	TokuDB is used to process large amounts of data and is therefore

intended for the so-called [big data](#) sets.

Database queries

The scope of application of MySQL does not differ to that of MariaDB. Both database management systems aim to achieve a **100% compatibility rate with the database query language SQL**. Users avail of identical SQL Statements when creating, updating, retrieving, or erasing database entries.



Tip

A [MySQL tutorial for beginners](#) offers a basic introduction into the world of SQL statements.

Performance

The efficiency of database systems has the most direct influence on business operations of any enterprise. Users interested in one of the open source projects of Widenius often ask themselves whether **MariaDB offers better performance levels than MySQL**. Benchmark tests such as [DBT-3](#) point to this conclusion. The decisive factor here is the type of database engines used by both the former and latter

system. Benchmark tests for set-ups with differing database engines are available to all users of the developer communities in question on their respective project websites.

The informative value of each practical application test is nevertheless bounded by certain limitations. After all, the performance of each database largely depends on the extent to which the implementation of new software has been tailored to respective needs and requirements. To be able to match the software to individual application cases, users should concentrate on the options offered by each database management system.

MariaDB takes the upper hand not only with its large selection of alternative database engines, but also with its efficient **SQL Query Optimizer**. In other words, when optimizing queries, MariaDB (from version 10.0.1 onwards) no longer looks at internal statistics of respective database engines, but rather takes into account [engine-independent table statistics](#). These are saved in various databases as conventional tables, which record and evaluate noticeably larger values, consequently determining the ideal plan for the implementation of SQL Statements.

Those who are interested in availing of [guidelines for the optimization of database performance](#) can find them on MariaDB's official website. Its corresponding articles offer information on operating systems, configurations, optimization strategies for data structure, index and table optimizations, performance strategies for various queries and subqueries, as well as tips and tricks on system variables, buffers,

caches, threads, database-type performance improvements, encryptions, hashing, and compressions.

MySQL also offers large-scale [informative materials on database tuning](#). These are, however, made available to a selected number of users who avail of an Oracle web account.

High availability

Both MySQL and MariaDB are suited for cases of distributed databases. To be able to offer users highly available systems with linear scalability, both software projects provide the following **cluster solutions**:

- MySQL Cluster
- MariaDB Galera Cluster

MySQL Cluster is all about ACID-based model of real-time transaction databases, which achieve a 99.9999% availability rate without any single point of failure by means of multi-master architecture.

Databases can be scaled horizontally thanks to standard hardware and offer an auto-sharing function for read- and write-intensive workloads. Accessing data depends on the SQL or NoSQL interface requirements. MySQL Cluster is released separately, with 7.5 as the current version. Just like MySQL, this cluster solution is sold with a dual license system. The community version, however, is provided with the GNU General Public License.

In distributed database systems, MariaDB is based on Codership's (Finnish software company) [cluster software Galera](#). Appropriate

interface structure has been integrated by default into MariaDB from version 10.1 onwards. Downloading a separate cluster version is therefore not necessary. Multi-master architecture is also an underlying structural component of MariaDB's Galera Cluster. InnoDB (or XtraDB) steps in as the sole database engine for cluster operations. To be able to withdraw individual cluster solutions from MySQL products, MariaDB's developer team have come up with **MaxScale** –a special extension to the database management system in question, which offers its users additional features enabling routing in distributed systems.

MaxScale is a database proxy primarily used for load distributions. It can be configured by means of plugins in such a way that it does not only forward database access, but also conforms to certain requirements when necessary. The proxy in question anonymizes sensitive data by depersonalizing it (by means of usernames), blocks requests in accordance with preconfigured rules, protects from denial of service (DoS) attacks, automatically initiates network failover mechanisms, and enhances the performance of databases by means of caching and read-write splitting (redirecting from read-and-write access to various database instances).

Contrary to MariaDB, MaxScale has not been covered by GPL license since its 2.0 version. Instead, MariaDB has developed its own license model – **Business Source License (BSL)** – which combines the demand for an open source community with the economic requirements of software development.

! Note

According to various developers, the Business Source License (BSL) is an alternative to Closed Source and Open Core. Though BSL's source code is always available for free, the use of software has a fixed time limit. MaxScale can be used free of charge in scenarios with up to three databases. Users who need more databases must acquire a specific license. When the **time limit set by the service supplier runs out**, BSL is automatically converted to a GPL-equivalent open source license. Each MaxScale version is available to all users without any limitations not later than 3 years after its release under GPL license.

Chief evangelist Kaj Arnö explains MariaDB's objectives related to BSL in a keynote during the 2016 IT Days in Frankfurt (starting from 35:35):



In order to protect your privacy, the video will not load

Kaj Arnö

until you click on it.

Die Geschichte von MySQL & MariaDB

Warum jetzt der richtige Zeitpunkt zum Umstieg auf MariaDB ist

Safety of use

Encryption, authentication, and user roles are among the most important safety features of a database management system.

Both MySQL and MariaDB offer **encryption methods for inactive data** (called data-at-rest). MySQL implements the encryption of data already saved in databases by means of the InnoDB engine at tablespace level. The encryption of individual database tables is not possible.

! Note

Tablespaces are logical storage units of relational database engines such as InnoDB, which contain all data of its corresponding database system. Each tablespace encompasses at least one physical file of its underlying operating system, in which database tables as well as indices are saved.

As of version 10.1, the encryption functions of MariaDB differ substantially from those of its competitors, as they include encryption methods for data-at-rest for the following database tiers:

- InnoDB tablespaces
- InnoDB tables
- InnoDB log files
- Aria tables
- Temporary datasets
- Binary log data

With the aid of Rolling Encryption Keys, MariaDB offers an additional function, which determines the expiry date of encryption keys.

Both database management systems support the **Key Management Service (KMS) of Amazon Web Services** by means of a special plug-in.

User authentication of both MySQL and MariaDB therefore relies on a plug-in system. MySQL's development team has made two authentication plug-ins for its users *-sha256_password* and *caching_sha2_password*. Apart from its standard authentication by means of a Secure Hash Algorithm, the latter also offers server-based caching enabling much quicker re-authentications.

Up until version 10.1.21, MariaDB, too, relied on **Secure Hash Algorithms in the form of SHA-1**. As of version 10.1.22, the ed25519-plugin has been implemented, which combines SHA-2 with Curve25519 by means of an EdDSA digital signature scheme.

The encryption technology used by MariaDB primarily stems from Google and the German software company Eperi.

Both database management systems in question support **SSL-encrypted connections**.

When it comes to database security, MariaDB's unique selling point is based on **role-based access control (RBAC)**, which has been implemented into its system from version 10.0.5 onwards. Above all, it eases the management of access rights and reduces the amount of errors related to the manual distribution of rights. So far, MySQL has been unsuccessful at implementing similar features.

Ecosystem

The dual licensing system used by Oracle in MySQL follows an open core strategy. While MySQL's community edition and its public source code are available for free, its various extensions are only supplied in conjunction with one of the commercial editions. Thanks to a strong cross-link with the open source community, MariaDB and its development team enabled the implementation of interfaces for open source **alternatives for the majority of proprietary MySQL products**.

Function	MySQL	MariaDB servers
Monitoring	MySQL Enterprise Monitor (proprietary)	Webyog Monyog (proprietary)
Backup	MySQL Enterprise Backup (proprietary)	MariaDB Backup (fork of Percona XtraBackup, GPL)
SQL	MySQL Workbench	Webyog SQLyog

Management	(GPL/proprietary)	(GPL/proprietary)
Load Balancing & Routing	MySQL Router (GPL/proprietary)	MariaDB MaxScale (BSL)
Firewall	MySQL Enterprise Firewall (proprietary)	MariaDB MaxScale (BSL)

Support

Professional support is only made available to a selected amount of MariaDB and MySQL users.

Those who avail of a license for proprietary MySQL products also gain access to **Oracle Premier Support**, which offers unlimited around-the-clock help, access to knowledge databases, maintenance releases, bug fixes, patches, and updates. As well as that, the service in question offers remote troubleshooting and professional advice on performance tuning and installation. If necessary, users can review and examine elements such as custom replications, partitions, outline plans, searches, APIs, server extensions, functions, and routines with the aid of professional MySQL support technicians.

The MariaDB Corporation sells support services, which can be acquired with a **MariaDB subscription**. The subscription also encompasses a 24/7 support with a maximum reaction time of 30 minutes as well as a notification service, patches, and bug fixes. Fee-based trainings, remote

database administration, performance-based advice, high availability, safety of use, software testing, and migration services are also included. According to MariaDB representatives, the support team offers professional assistance for both MariaDB and MySQL. Since the latter's development team has almost entirely switched to the former company, MariaDB prides itself in having employees with a more advanced understanding of MySQL's code than Oracle's support team itself.

Documentation and community

MySQL is owned by the Oracle Corporation, which has published an extensive amount of [documentation on all of MySQL's open source projects](#). Furthermore, to get a better grasp on MySQL and its corresponding concepts, users can avail of the following options:

- [MySQL Wiki](#)
- [MySQL Events](#)
- [MySQL User Groups](#)

The MariaDB Foundation is responsible for any [MariaDB-related documentation](#) and the support of respective user communities.

Those who want to get involved in the development and documentation process of MariaDB servers or open source software application of MariaDB's ecosystem can find relevant information on the [official website of the MariaDB Foundation](#).

MariaDB vs. MySQL – comparing the most important features

MySQL is a proven database management system made available to all users in the form of open source software. Furthermore, it is supplemented by numerous enterprise-oriented extensions. MariaDB's development team lead by Michael Widenius has taken over the most important components of MySQL's software and enhanced it with additional features. By performing a comparison of the main characteristic of the two database management systems in question, the following table shows the possible benefits of a switchover from MySQL to MariaDB.

Feature	MySQL	MariaDB
Release year	1995	2009
Current version	MySQL 8.0.11	MariaDB 10.3.8
Developer	Oracle	MariaDB Corporation AB (MariaDB Enterprise) MariaDB Foundation (MariaDB Server)
License	Dual licensing system (Proprietary and GPL 2)	GPL 2

Possible operating systems	Oracle Linux	Red Hat/CentOS
	Red Hat	Ubuntu
	CentOS	Debian
	Oracle Solaris	Mint
	Ubuntu	SUSE Enterprise Linux
	Debian	openSUSE
	SUSE Enterprise Linux	Microsoft Windows Server
	Microsoft Windows	Microsoft Windows
	Microsoft Windows Server	Manjaro
	Apple macOS	Fedora
	FreeBSD	Arch Linux
	Solaris	Apple macOS
Programming language	C and C++	C, C++, Bash, and Perl
Primary database model	Relational database model	Relational database model

Secondary database model	Document-oriented database model	Document-oriented database model
	Key-value database	Key-value database
		Graph-oriented database
SQL interface	Yes	Yes
NoSQL interface	Yes	Yes
Protocol	MySQL-Protocol	MySQL-Protocol
Replication	Replication with GTID (global transaction identifier)	Replication with GTID (global transaction identifier)
	- Master-master replication	- Master-master replication
	- Master-slave replication	- Master-slave replication
In-memory support	Yes	Yes
Routing	MySQL Router (GPLv2)	MariaDB MaxScale (BSL)
Partitioning	Horizontal partitioning,	Horizontal partitioning,

(sharding)	sharding by means of MySQL Cluster or MySQL Fabric	sharding by means of Spider, CONNECT, or Galera
SQL management	MySQL Workbench (Microsoft Windows, macOS, and Linux)	SQLyog of Webyog (Microsoft Windows, Linux)
Monitoring	MySQL Enterprise Monitor (proprietary)	Monyog of Webyog (Microsoft Windows and Linux) (proprietary)
Encryption	Encryption of inactive data - InnoDB tablespace	Encryption of inactive data - InnoDB tablespace - InnoDB tables - InnoDB log files - Aria tables - Temporary datasets - Binary log data
Role-based access control (RBAC)	No	Yes

Authentication	Authentication via SHA-256	Authentication via ed25519 plugin
Data masking	Via ProxySQL	Via MariaDB MaxScale (BSL)
Firewall	Via MySQL Enterprise Firewall (proprietary)	Via MariaDB MaxScale (BSL)
Auditing	Via MySQL Enterprise Audit Plugin (proprietary)	Via MariaDB Audit Plugin
Analysis	No	Via MariaDB ColumnStore
Routing & load-balancing	Via MySQL Router	Via MariaDB MaxScale (BSL)
Backup	Via MySQL Enterprise Backup (proprietary)	Via MariaDB Backup (fork of Percona XtraBackup)
Common table Expression (CTE)	Yes	Yes
Window function	Yes	Yes

Temporary tables by means of version management	No	Yes
Query rewriting	Yes	No
Data types	String (CHAR, VARCHAR, TINYTEXT, TEXT, MEDIUMTEXT, LONGTEXT, ENUM, SET, BINARY, VARBINARY, JSON)	String (CHAR, VARCHAR, TINYTEXT, TEXT, MEDIUMTEXT, LONGTEXT, ENUM, SET, BINARY, VARBINARY, JSON)
	Numerical (BIT, TINYINT, SMALLINT, MEDIUMINT, INT, BIGINT, DECIMAL, FLOAT, DOUBLE, BOOLEAN)	Numerical (BIT, TINYINT, SMALLINT, MEDIUMINT, INT, BIGINT, DECIMAL, FLOAT, DOUBLE, BOOLEAN)
	Date/time (DATE, DATETIME, TIMESTAMP, TIME, YEAR)	Date/time (DATE, DATETIME, TIMESTAMP, TIME, YEAR)
	Large Object Datatypes (TINYBLOB, BLOB, MEDIUMBLOB, LONGBLOB)	Large Object Datatypes (TINYBLOB, BLOB, MEDIUMBLOB, LONGBLOB)

Spatial data types	GEOMETRY	GEOMETRY
	POINT	POINT
	LINESTRING	LINESTRING
	POLYGON	POLYGON
	MULTIPOINT	MULTIPOINT
	MULTILINESTRING	MULTILINESTRING
	MULTIPOLYGON	MULTIPOLYGON
	GEOMETRYCOLLECTION	GEOMETRYCOLLECTION
Database engines	InnoDB	InnoDB/XtraDB
	MyISAM	MyISAM
	MEMORY	MEMORY
	CSV	CSV
	Archive	Archive
	BLACKHOLE	BLACKHOLE
	Merge	Merge
	Federated	FederatedX

		ColumnStore
		Aria
		Cassandra
		CONNECT
		Mroonga
		MyRocks
		OQGRAPH
		Sequence
		SphinxSE
		Spider
		TokuDB
Official connectors	ODBC	ODBC
	C++	C++
	C	C
	ADO.NET	ADO.NET
	JDBC	JDBC
	PHP	PHP

Python	Python
Perl	Perl
Ruby	Ruby
Node.js	Excel
	JavaScript
	Swift
	R

Summary

Oracle's MySQL is a well-respected database management system which has enjoyed huge popularity for more than 20 years. Under Oracle's patronage, it can assert itself as an independent software project with space for continuous maintenance, development, and further improvements. However, those wishing to exhaust all potentialities of MySQL's ecosystem can quickly **encounter various limitations of MySQL's free community edition**. Professional tools for monitoring, backup, load balancing, routing, and database safety purposes are only offered in conjunction with an appropriate fee-based license.

Though MariaDB emerged from MySQL, the former is nowadays an autonomous database solution for productive use.

Fact

MariaDB has established itself as a serious alternative to MySQL.

This is mainly evident in the fact that various Linux distributions replaced MySQL with MariaDB as the standard installation form at the end of 2012. Fedora, CentOS, openSUSE, Arch Linux, Debian, and Red Hat Enterprise Linux were among the affected distribution systems. To top it off, Google, Mozilla, the Wikipedia Foundation, TeamSpeak, and XAMPP have also begun availing of MariaDB's services.

MariaDB consists of MySQL's development team which can rely on the large support of the open source community. This makes it possible for the MariaDB Foundation to render open source alternatives available to numerous enterprise-oriented extensions of MySQL's ecosystem, which do not necessarily comply with all open source standards envisaged by Bruce Perens and Eric S. Raymond.

Both software projects in question will diverge even further away from each other in the near future. Up until now, MariaDB has been primarily seen as an open source drop-in replacement for MySQL. However, it will soon prioritize **the development of exclusive features**

and extensions such as MaxScale and ColumnStore. Such application possibilities give MariaDB the edge over MySQL. The reputation of the development team is therefore no longer the decisive factor between MySQL or MariaDB. Instead, it is worth questioning the extent to which the spectrum of functions of each database management system covers your needs.

🕒 10.03.20 | Technical matters

How did you like the article? 4 ★★★★★ (6) [f](#) [t](#) [x](#) [in](#)

MySQL Database

Related articles



MongoDB – the flexible and scalable NoSQL database



MongoDB tutorial: installation and setup



Prepared statements in PHP: Basics and examples

🕒 14.10.2019

| Web development

NoSQL databases have long since been serious competitors to classic relational database systems, like MySQL. They replace the strictly-defined storage structures of traditional models through flexible saving procedures. This allows data pools to be easily distributed onto multiple servers in order to guarantee high availability of web applications. But are these new systems really all that they're...



🕒 19.06.2020

| Web development

MongoDB is a document-based database system that competes with the more traditional relational systems. The setup and configuration is comparable to classics like MySQL, making it fairly straightforward to switch over. So how do you save and call up data with this modern software? Our MongoDB tutorial will explain the installation process and first steps to get you started on this open source...



🕒 24.06.2020

| Web development

In order to improve the performance of repeated SQL queries and also significantly reduce the risk of manipulation in database access, you can take advantage of prepared statements in PHP and other programming languages. But how does this work, and how can you use prepared statements for managing your database systems?





[About 1&1](#) [Terms and Conditions](#) [Privacy Policy](#) [Help Center](#) [Tell a friend](#)
© 2020 1&1 IONOS Inc.