

How to Display the Storage Engine in MariaDB?

written by sysadmin | 17 September 2025

If you are a user of the MariaDB database, you should know about the term storage engine.

Problem

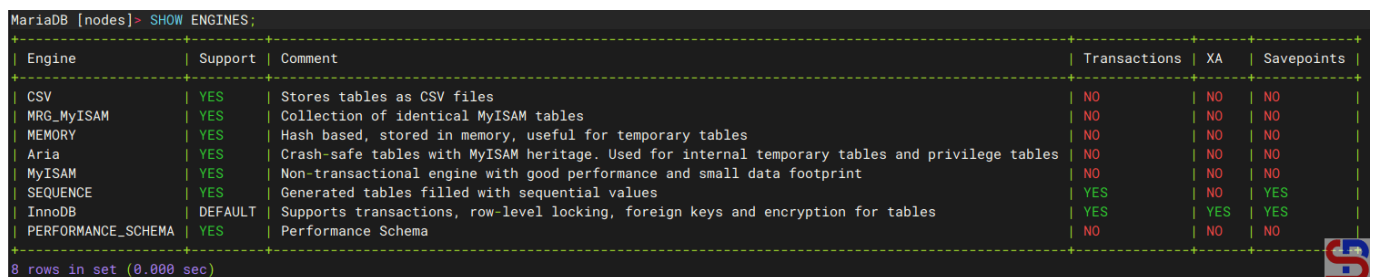
How to display the storage engine in MariaDB?

Solution

A database engine (or storage engine) is the underlying software component that a database management system (DBMS) uses to create, read, update, and delete (CRUD) data from a database. To see a list of all available storage engines on the server, use the command below in the MariaDB prompt:

```
SHOW ENGINES;
```

and it will appear as shown in the image below:



Engine	Support	Comment	Transactions	XA	Savepoints
CSV	YES	Stores tables as CSV files	NO	NO	NO
MRG_MYISAM	YES	Collection of identical MyISAM tables	NO	NO	NO
MEMORY	YES	Hash based, stored in memory, useful for temporary tables	NO	NO	NO
Aria	YES	Crash-safe tables with MyISAM heritage. Used for internal temporary tables and privilege tables	NO	NO	NO
MyISAM	YES	Non-transactional engine with good performance and small data footprint	NO	NO	NO
SEQUENCE	YES	Generated tables filled with sequential values	YES	NO	YES
InnoDB	DEFAULT	Supports transactions, row-level locking, foreign keys and encryption for tables	YES	YES	YES
PERFORMANCE_SCHEMA	YES	Performance Schema	NO	NO	NO

8 rows in set (0.000 sec)

Display all the storage engines in MariaDB

From the image above, you can see that InnoDB is the default storage engine in MariaDB. You can also use the command below to see the default storage engine in MariaDB:

```
SHOW GLOBAL VARIABLES LIKE 'default_storage_engine';
```

```
MariaDB [nodes]> SHOW GLOBAL VARIABLES LIKE 'default_storage_engine';
```

Variable_name	Value
default_storage_engine	InnoDB

```
1 row in set (0.006 sec)
```



Display the default storage engine in MariaDB

You can change the default storage engine by using the command below, which changes the default storage engine to MyISAM:

```
SET GLOBAL default_storage_engine='MyISAM';
```

Or you can use the below command if the session default storage engine supersedes the global default during this session:

```
SET SESSION default_storage_engine='MyISAM';
```

You should know that the storage engine is used per table and not per database, and to see it, you can use the command below:

```
SELECT TABLE_NAME, ENGINE
FROM information_schema.TABLES
WHERE TABLE_SCHEMA = 'nodes';
```

and the result will look like below:

```
MariaDB [nodes]> SELECT TABLE_NAME, ENGINE  
-> FROM information_schema.TABLES  
-> WHERE TABLE_SCHEMA = 'nodes';
```

```
+-----+-----+  
| TABLE_NAME          | ENGINE |  
+-----+-----+  
| flaggedrevs          | InnoDB |  
| flaggedrevs_statistics | InnoDB |  
| category             | InnoDB |  
| babel                | InnoDB |  
+-----+-----+  
4 rows in set (0.000 sec)
```



Display the storage engine used in the table

From the image above, it can be seen that all tables in the nodes database use InnoDB, so it can be said that the nodes database uses InnoDB. However, one database (schema) can have a table with a different engine, and the query below is the sample to create 2 tables that use different storage engines:

```
CREATE TABLE product (  
    id INT PRIMARY KEY,  
    name VARCHAR(100)  
) ENGINE=InnoDB;  
  
CREATE TABLE access_log (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    time DATETIME,  
    user VARCHAR(50)  
) ENGINE=MyISAM;
```

where the product table uses InnoDB and the table access_log uses MyISAM. If a database uses various storage engines and you want to know how many storage engines are used, you can use the query below:

```
SELECT ENGINE, COUNT(*) AS Number_of_tables  
FROM information_schema.TABLES  
WHERE TABLE_SCHEMA = 'db_name'  
GROUP BY ENGINE;
```

You can change the storage engine of a table by using the query below, for example, change it to MyISAM:

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

```
MariaDB [nodes]> SELECT TABLE_NAME, ENGINE
-> FROM information_schema.TABLES
-> WHERE TABLE_SCHEMA = 'nodes';
+-----+-----+
| TABLE_NAME | ENGINE |
+-----+-----+
| flaggedrevs | InnoDB |
| flaggedrevs_statistics | InnoDB |
| category | InnoDB |
| babel | InnoDB |
+-----+-----+
4 rows in set (0.000 sec)
```

```
MariaDB [nodes]> ALTER TABLE nodes.babel ENGINE=MyISAM;
Query OK, 19570 rows affected (0.076 sec)
Records: 19570 Duplicates: 0 Warnings: 0
```

```
MariaDB [nodes]> SELECT TABLE_NAME, ENGINE FROM information_schema.TABLES WHERE TABLE_SCHEMA = 'nodes';
+-----+-----+
| TABLE_NAME | ENGINE |
+-----+-----+
| flaggedrevs | InnoDB |
| flaggedrevs_statistics | InnoDB |
| category | InnoDB |
| babel | MyISAM |
+-----+-----+
4 rows in set (0.000 sec)
```



Change the storage engine in the table

Note

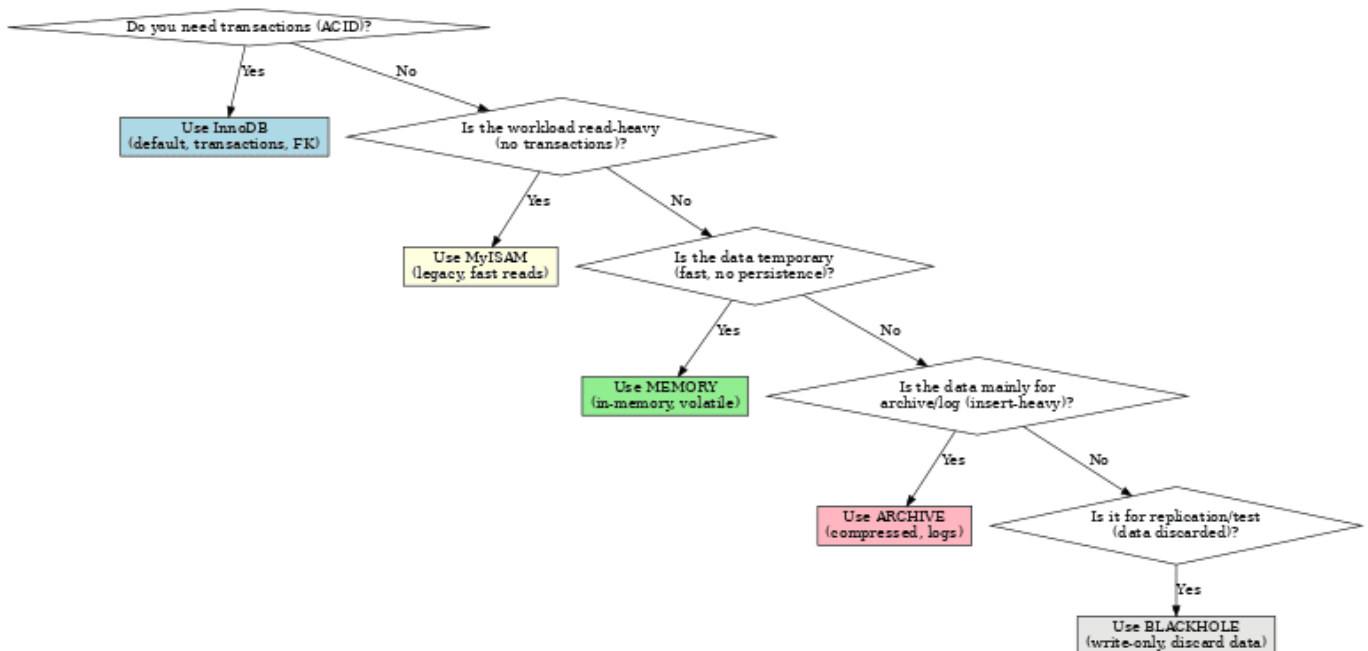
As explained above, there are many storage engines that you can use. But before you decide to use a storage engine, you should first find out the advantages and disadvantages of each storage engine, and to briefly see the differences between storage engines in the image below:

COMPARISON OF STORAGE ENGINES IN MARIADB/MYSQL

Feature / Engine	InnoDB (default)	MyISAM	MEMORY	ARCHIVE	BLACKHOLE
Transactions (ACID)	✓	✗	✗	✗	✗
Foreign Keys	✓	✗	✗	✗	✗
Locking	Row-level	Table-level	Table-level	Table-level	N/A
Crash recovery	Yes (redo /undo logs)	Limited (can lose data)	Data lost (in RAM)	Yes (insert only, compressed)	Accepts writes but discards data
Read/Write speed	Balanced	Very fast on reads, slower writes	Extremely fast (in-memory)	Optimized for inserts only	None
Storage usage	General purpose apps, OLTP, most modern apps	Read-footprint workloads, legacy apps	RAM only	Very small (compressed)	Replication (pass-through), testing

Comparison between the storage engines

And below is a flowchart to determine the storage engines you will use:



Flowchart to choose the storage engine

However, as far as I know, among the many choices of storage engines, generally people use 2 types of storage engines, namely InnoDB and MyISAM. Maybe you can see the difference between the two through [this site](#).

References

en.wikipedia.org

slideshare.net

medium.com

support.sages.com

mariadb.com

dev.mysql.com

managedserver.eu