

# Engine trong Database

QuanLX

## Mục lục

1. Bối cảnh.....	3
2. Hệ quản trị cơ sở dữ liệu .....	3
3. So sánh MYSQL và MariaDB .....	5
4. So sánh MYSQL vs POSTGRESQL.....	8
5. Storage Engine phổ biến.....	13
5.1 InnoDB .....	14
5.2 MyISAM.....	15
5.3 Memory engine.....	15
5.4 Lựa chọn storage engine nào?.....	16

# 1. Bối cảnh

Hiện tại trong dự án công ty, hầu hết mọi người chỉ dùng và làm việc đến các cơ sở dữ liệu như: MySQL, Mongo, Redis. Nên đối với các hệ quản trị cơ sở dữ liệu (DBMS) khác thì mọi người chưa có nhiều khái niệm, chưa biết đến những ưu nhược điểm của từng loại. Sau bài này, hy vọng mọi người biết thêm một vài DBMS khác và có cái nhìn tổng quan, và sẽ phần nào giúp mọi người đưa ra quyết định nên chọn DBMS nào để phù hợp vs bài toán và business logic khi làm việc.

## 2. Hệ quản trị cơ sở dữ liệu

Hệ quản trị cơ sở dữ liệu (DBMS) là một phần mềm cho phép tạo lập các CSDL cho các ứng dụng khác nhau và điều khiển mọi truy cập tới các CSDL đó. Nghĩa là, hệ quản trị CSDL cho phép định nghĩa (xác định kiểu, cấu trúc, ràng buộc dữ liệu), tạo lập (lưu trữ dữ liệu trên các thiết bị nhớ) và thao tác (truy vấn, cập nhật, kết xuất, ...) các CSDL cho các ứng dụng khác nhau.

Có rất nhiều DBMS, phổ biến nhất hiện nay như: oracle, mysql, mysql server, postgresql, mongoDB,...

Cùng theo dõi xem top 10 DBMS được sử dụng phổ biến trên thế giới cho các ứng dụng công nghệ thông tin theo DB-Engines Ranking: <https://db-engines.com/en/ranking>

Rank			DBMS	Database Model	Score		
Mar 2021	Feb 2021	Mar 2020			Mar 2021	Feb 2021	Mar 2020
1.	1.	1.	Oracle	Relational, Multi-model	1321.73	+5.06	-18.91
2.	2.	2.	MySQL	Relational, Multi-model	1254.83	+11.46	-4.90
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1015.30	-7.63	-82.55
4.	4.	4.	PostgreSQL	Relational, Multi-model	549.29	-1.67	+35.37
5.	5.	5.	MongoDB	Document, Multi-model	462.39	+3.44	+24.78
6.	6.	6.	IBM Db2	Relational, Multi-model	156.01	-1.60	-6.55
7.	7.	8.	Redis	Key-value, Multi-model	154.15	+1.58	+6.57
8.	8.	7.	Elasticsearch	Search engine, Multi-model	152.34	+1.34	+3.17
9.	9.	10.	SQLite	Relational	122.64	-0.53	+0.69
10.	11.	9.	Microsoft Access	Relational	118.14	+3.97	-7.00
11.	10.	11.	Cassandra	Wide column	113.63	-0.99	-7.32
12.	12.	13.	MariaDB	Relational, Multi-model	94.45	+0.56	+6.10
13.	13.	12.	Splunk	Search engine	86.93	-1.61	-1.59
14.	14.	14.	Hive	Relational	76.04	+3.72	-9.34
15.	16.	15.	Teradata	Relational, Multi-model	71.43	+0.53	-6.41

Trong khuôn khổ bài viết này, chúng ta cùng đi tìm hiểu về 3 DBMS: **MySQL**, **PostgreSQL**, **MariaDB**.

→ Điểm chung của 3 DBMS này đầu tiên là: cả 3 đều là hệ cơ sở dữ liệu theo mô hình quan hệ (RDBMS) mã nguồn mở dựa trên ngôn ngữ truy vấn SQL

Hiện tại, hệ thống của chúng ta đang sử dụng RDBMS là MYSQL, chính vì vậy trong bài viết này chúng ta sẽ đặt MYSQL là tiêu điểm để đi so sánh về ưu - nhược điểm so với các RDBMS còn lại. Qua đó chúng ta sẽ hiểu cơ bản về các engine DB và đánh giá xem chúng ta đã tận dụng hết sức mạnh của MYSQL chưa, có nên chuyển sang các RDBMS khác không?

### 3. So sánh MySQL và MariaDB



Cùng tìm hiểu một chút về lịch sử của 2 thằng này nhé. Khá là thú vị.

MySQL là một trong những cơ sở dữ liệu được sử dụng rộng rãi nhất trên thế giới được phát triển dựa trên C / C ++ được phát triển năm 1994. Sau đó được Sun Microsystems mua lại vào năm 2008. Tiếp sau đó vào năm 2010 Sun Microsystems đã được Oracle mua lại và kể từ đó MySQL được duy trì và quản lý bởi đội ngũ Oracle. MariaDB thì sao? Trong quá trình mua lại Sun Microsystems của Oracle thì một số kỹ sư cao cấp đang nghiên cứu phát triển MySQL cảm thấy có xung đột lợi ích giữa cơ sở dữ liệu thương mại của MySQL và Oracle. Kết quả là các kỹ sư đó đã tạo ra một nhánh của cơ sở mã MySQL và như thế MariaDB được tạo ra.

MariaDB được đặt với ý tưởng từ tên con gái (Maria) của nhà sáng lập - Michael "Monty" Widenius. Tương tự như cách MySQL được đặt tên theo con gái của ông là My.

MariaDB có một số tối ưu hóa có xu hướng cải thiện hiệu suất so với MySQL. Bảng sau minh họa sự khác biệt so sánh giữa MariaDB và MySQL

	MariaDB	MySql
--	---------	-------

Nhà phát triển	MariaDB Corporation AB (MariaDB Enterprise) và MariaDB Foundation (community MariaDB Server)	Oracle
Phát hành lần đầu	2009	1995
Phát hành hiện tại	10.3.10, October 2018	8.0.12, July 2018
Giấy phép	Mã nguồn mở	Mã nguồn mở + độc quyền
Phát triển	Mở	Đóng
Công cụ lưu trữ	Aria (được xem như một phiên bản cập nhật của MyISAM) XtraDB (thay thế cho InnoDB) BLACKHOLE CSV MEMORY ARCHIVE MERGE ColumnStore MyRocks SphinxSE TokuDB (xử lý dữ liệu lớn) CONNECT SEQUENCE Spider Cassandra (hỗ trợ NoSQL )	InnoDB MyISAM BLACKHOLE CSV MEMORY ARCHIVE MERGE
Ngôn ngữ triển khai	C và C++	C và C++
Hệ điều hành máy chủ	FreeBSD Linux Solaris Windows	FreeBSD Linux OS X Solaris Windows
Faster cache/indexes	Với công cụ lưu trữ bộ nhớ của MariaDB, một câu lệnh INSERT có thể được hoàn thành nhanh hơn so với MySQL tiêu chuẩn.	memory storage engine của MySQL chậm hơn so với

		MariaDB.
New Features/Extensions	MariaDB đi kèm với các tính năng và tiện ích mở rộng mới sớm hơn bao gồm các câu lệnh JSON, WITH và KILL, GIS(giúp lưu trữ dữ liệu định vị và truy vấn một cách dễ dàng hơn)	Các tính năng MariaDB mới không được cung cấp trong MySQL.

	MariaDB	MySql
tốc độ và sức chứa Connection Pool	MariaDB đi kèm với thread pool nâng cao có khả năng chạy nhanh hơn và hỗ trợ hơn 200.000 kết nối.	Thread pool được cung cấp bởi MySQL không thể hỗ trợ tới 200.000 kết nối mỗi lần.
Cải tiến Replication	Trong MariaDB, replication có thể được thực hiện an toàn hơn và nhanh hơn. replication cũng có thể được thực hiện nhanh hơn gấp 2 lần so với MySQL truyền thống.	Phiên bản cộng đồng của MySQL cho phép số lượng chủ đề tĩnh được kết nối. MySQL's enterprise mới đi kèm với khả năng mở rộng.

Tóm lại với nhiều tính năng hay và ra đời sớm hơn, cùng với đó là việc hỗ trợ nhiều stored Engine hơn, MariaDB đang ngày càng chứng tỏ sức mạnh. Được bảo trì bằng chính người đã tạo ra Mysql, bên cạnh đó MariaDB còn hoàn toàn tương thích với Mysql ở các phiên bản 5.1 -> 5.5 (bạn có thể switch qua lại 1 cách ngon lành). Đặc biệt ở MariaDB, có thể kết hợp cả SQL và NoSQL được cung cấp bởi stored engine Cassandra.



Mặc dù vẫn có một vài thí nghiệm benchmark chỉ ra rằng MariaDB nhanh hơn MySQL từ 3-5%, tuy nhiên để khẳng định là tốc độ nhanh hơn thì rất khó để nói. Mặc dù vậy, với sự phát triển của cộng đồng do cộng đồng thì MariaDB sẽ được tối ưu ngày càng tốt hơn.

Một trong những website lớn nhất thế giới là Wikipedia đã chuyển từ MySQL qua MariaDB. Ngay cả Fullstack-Station cũng đang dùng MariaDB.

## 4. So sánh MYSQL vs POSTGRESQL

Đầu tiên, thì đi hiểu qua trước PostgreSql là gì?

PostgreSQL là một hệ quản trị cơ sở dữ liệu quan hệ – đối tượng, được phát triển bởi Khoa Điện toán, Đại học California – Hoa Kỳ dựa trên Postgres bản 4.2.

PostgreSQL là mã nguồn mở miễn phí, được xây dựng theo chuẩn SQL99. Người dùng có thể tự do sử dụng, chỉnh sửa và phân bổ PostgreSQL theo nhiều hình thức khác nhau., có tính ổn định cao, có thể phát triển nhiều ứng dụng khác nhau với chi phí tương đối thấp.

Vậy so với Mysql thì PostgreSQL có gì khác biệt, Cùng theo dõi một số tiêu chí quan trọng sau đây được đánh giá giữa hai nhà chuyên môn là Sawada chuyên PostgreSQL với Tanaka chuyên MySQL





	MySQL	PostgreSQL
<b>Non-blocking trong DDL (Định nghĩa dữ liệu)</b>	<ul style="list-style-type: none"> <li>– Thực thi thao tác nhiều DDL bằng Non – blocking. (Tức là, dù trong transaction nhưng không có block đến bảng.) → tốc độ xử lý nhanh, kéo thiểu việc giảm tải cho máy chủ server.</li> <li>– Chức năng Non – Blocking có từ phiên bản MySQL 5.6.</li> </ul>	<ul style="list-style-type: none"> <li>– PostgreSQL sẽ tùy thuộc vào những câu lệnh thực thi thao tác DDL nào.</li> <li>- Thao tác write, chẳng hạn như thêm cột, nó sẽ phát sinh block đến bảng, khiến bạn không thể tham chiếu thêm nữa</li> <li>– Có thể thấy, PostgreSQL thao tác DDL là Alter table không phải non – blocking. Để sử dụng được, bạn cần sự hỗ trợ của các tool. Ví dụ như pg_repack, tool này chuyên dùng cho maintain, giúp thực thi các thao tác alter table một phần hoặc tiến hành reindex bằng cách block tối thiểu.</li> </ul>

<p><b>Performance của DML (thao tác dữ liệu)</b></p>	<p>– MySQL có thuật toán sort không tốt bằng PostgreSQL. Bởi vậy MySQL sẽ bị chậm hơn khi dùng những câu SELECT cần sort</p> <p>– MySQL chuyên về use case. Ví dụ, lấy ra 10 hay 100 dữ liệu đầu tiên (như cách của Twitter) sẽ nhanh hơn so với PostgreSQL</p> <p>– UPDATE thì performance của MySQL tốt hơn so với PostgreSQL. Bởi MySQL ghi đè đối tượng được update, đúng nghĩa cập nhật.</p> <p>– MySQL delete chậm do sau khi xóa dữ liệu nó sẽ thực hiện đánh lại secondary index (tất cả index ngoại trừ cluster index) xử lý đồng bộ nên tốn thời gian hơn. Từ phiên bản 5.5 trở lên sẽ cải thiện tình trạng này tốt hơn.</p>	<p>– Câu SELECT cần phải sort lượng dữ liệu lớn sau khi ORDER BY thì PostgreSQL sẽ làm việc nhanh hơn.</p> <p>– Chậm hơn</p> <p>– PostgreSQL khi update sẽ xử lý tương tự như insert. Tức là nó sẽ đánh dấu flag như delete vào dòng trước khi thay đổi, rồi thêm dòng mới có dữ liệu sau khi thay đổi</p> <p>– Delete nhanh hơn.</p>
<p><b>Thuật toán JOIN</b></p>	<p>– MySQL chủ yếu sử dụng thuật toán Support Nested Loop Join, bởi hệ quản trị cơ sở dữ liệu này thiết kế đúng theo tiêu chí thuật toán đơn giản.</p>	<p>– PostgreSQL support cả 3 loại Hash Join, Nested Loop Join, Sort Merge Join. Trong đó:</p> <ul style="list-style-type: none"> <li>• Sử dụng dữ liệu cần join nhiều thì dùng Hash Join và Sort Merge join.</li> <li>• Dữ liệu đã được sort thì sử dụng Sort Merge Join.</li> <li>• Dữ liệu của các bảng được Join ít hoặc bảng ít bảng nhiều, bạn có thể dùng Nested Loop Join.</li> </ul>

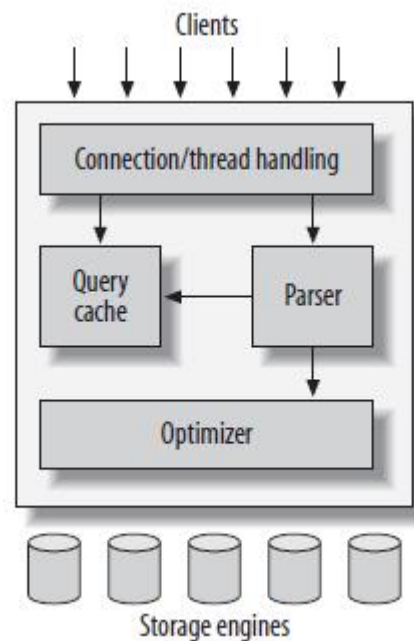
<p><b>Xử lý transaction</b></p>	<p>– MySQL mặc định xử lý transaction là repeatable-read. Phương thức này giúp giữ nguyên dữ liệu đọc, không lo thay đổi, mất ở một transaction khác.</p> <p>– Trong trường hợp thêm/update dữ liệu bởi transaction khác thì không tránh được phantom-read. Để cải thiện tình trạng này, khi sử dụng MySQL cần dùng Next Key Locking</p> <p>→ nhưng lại gặp vấ khác: .Ví dụ như là sử dụng SELECT FOR UPDATE chẳng hạn, mà trong mệnh đề WHERE sử dụng dấu nhỏ hơn &lt; để tìm kiếm records mà dưới 10 thì tất cả các key trên 10 đều bị locking. Khi này thì không thể phát sinh thêm key mới nữa khiến không thể INSERT được dữ liệu.</p>	<p>– <b>PostgreSQL</b> có tính năng Read – committed, hình thức này có khả năng xảy ra vấn đề phantom-read hoặc non – repeatable – read (hiện tượng cùng một transaction đọc cùng dữ liệu mà giá trị thay đổi).</p> <p>– Trường hợp đổi sang Repeatable – read sẽ không có next key locking nên tránh được tình trạng phantom – read. Bởi vậy PostgreSQL để tránh conflict ngon hơn MySQL.</p>
<p><b>Store procedure, Trigger</b></p>	<p>– MySQL chỉ sử dụng được SQL. Trên mỗi SQL không thực thi step của store procedure được.</p> <p>– Bản MySQL 5.6 có điểm yếu là mỗi một bảng chỉ gán được tối đa 6 trigger. Nếu không thì chúng chỉ có for each row.</p>	<p>– Ngoài SQL, PostgreSQL còn sử dụng được cả procedure bên ngoài như python.</p>
<p><b>Dạng logic và vật lý của replication</b></p>	<p>– MySQL có replication dạng logic và vật lý. Dạng logic được setting mặc định. Từ phiên bản 5.7 về sau thì dạng vật lý là mặc định.</p>	<p>– PostgreSQL chỉ có dạng vật lý ( copy image dòng sau khi thay đổi). Nhưng từ phiên bản release beta version 10 có thể sử dụng được cả dạng logic (copy câu</p>

		SQL).
<b>Chức năng tiện lợi</b>	<p>– MySQL các version thấp hơn chưa có chức năng này, nhưng kể từ 8.0 thì đã đc cập nhật.</p> <p>– MySQL thì có backup vật online còn Xtrabackup hoặc enterprise backup không thực hiện được base backup online hay trên remote.</p>	<p>– PostgreSQL có hàm window, có thể apply hàm tổng, hợp thành set kết quả và phân chia thành từng phần.</p> <p>- Mệnh đề With có thể subquery trước khi select. Bởi vậy, chức năng tiện lợi của postgresQL mạnh hơn.</p> <p>– Chức năng Query song song giúp tăng tốc độ xử lý bằng cách sử dụng nhiều CPU để chạy query. PostgreSQL có đa dạng các tool từ OSS, PostGIS,...</p> <p>– Chức năng Pg_basebackup có thể sử dụng online hoặc dùng trên remote đều được.</p>
<b>Về sự lỏng lẻo loại dữ liệu, thay đổi loại, so sánh chuỗi</b>	<Làm câu hỏi quiz trên lớp...> <update sau>	

### Tổng quan lại:

- PostgreSQL có điểm mạnh lớn nhất là nhiều chức năng nên hệ thống nào cần nhiều chức năng hỗ trợ thì nó có lẽ sẽ phù hợp hơn. Ví dụ như khi chuyển từ Oracle Database sang, hay những doanh nghiệp hệ Sler (System Integration-er). Sau nữa là hệ phân tích cũng dùng nhiều.
- MySQL phù hợp đối với những dịch vụ web đơn giản. tức hệ thống mà lấy ra lượng dữ liệu nhất định rồi hiển thị ra. Nhưng về sau khi nâng version thì có lẽ cả PostgreSQL lẫn MySQL cũng sẽ có những tính năng được nâng cấp tốt hơn. Và lúc đó ai thích cái nào thì dùng do không còn nhiều khác biệt.

## 5. Storage Engine phổ biến.



Storage engine là 1 thành phần phần mềm nằm bên dưới 1 hệ quản trị cơ sở dữ liệu (Database management system – DBMS), được dùng để tạo, đọc, cập nhật và xóa dữ liệu từ 1 cơ sở dữ liệu (CSDL). Đa số các DBMS cung cấp các API cho phép người dùng tương tác với những engine này mà không cần thông qua giao diện người dùng của DBMS.

Nhiều DBMS hiện đại hỗ trợ nhiều storage engine trong cùng 1 CSDL. Cùng tìm hiểu một số engine **phổ biến nhất hiện nay**.

## 5.1 InnoDB

Đây cũng là dạng storage engine cân bằng giữa các yếu tố về hiệu năng và độ tin cậy.

InnoDB lưu dữ liệu trên 1 file (thuật ngữ gọi là tablespace).

Đặc điểm:

- InnoDB sử dụng cơ chế **row-level locking**, tức là khi có hành động thêm-sửa-xóa lên một bản ghi thì chỉ có bản ghi đó bị khóa lại, các bản ghi khác vẫn thực hiện thêm-sửa-xóa bình thường.
- InnoDB có thể chứa tối đa 64TB dữ liệu.
- Hỗ trợ transaction.
- Hỗ trợ ràng buộc với các bảng khác (khóa ngoại).
- Hỗ trợ đánh index (chỉ mục).
- Hỗ trợ index full-text, cho phép tìm kiếm full-text.
- InnoDB hỗ trợ MVCC (Multiversion Concurrency Control) để cải thiện việc truy cập đồng thời và hỗ trợ chiến thuật next-key locking.
- InnoDB được xây dựng dựa trên clustered index, do đó việc tìm kiếm theo primary key có hiệu năng rất cao. InnoDB không hỗ trợ sắp xếp index do vậy việc thay đổi cấu trúc bảng liên quan đến column thuộc index sẽ dẫn tới toàn bộ dữ liệu phải được đánh chỉ mục từ đầu (CHẬM với những bảng lớn)

Đối với MySQL, **InnoDB** là storage engine mặc định (từ version 5.5 đến hiện tại - 8.0) khi bạn tạo một table mà không chỉ rõ storage engine.

Là engine phức tạp nhất trong các engine của MySQL

## 5.2 MyISAM

MyISAM được thiết kế để phù hợp với những table có thao tác đọc (select) nhiều hơn các thao tác thêm-sửa-xóa, vì:

- MyISAM sử dụng cơ chế table-level locking
- MyISAM có thể đánh chỉ mục full-text, hỗ trợ tìm kiếm full-text.
- MyISAM hỗ trợ nén dữ liệu, hỗ trợ tăng tốc độ đọc dữ liệu. Mặc dù vậy dữ liệu sau khi nén không thể cập nhật được.
- MyISAM không ghi dữ liệu ngay vào ổ đĩa cứng, mà ghi vào 1 buffer trên memory (và chỉ ghi vào đĩa cứng sau 1 khoảng thời gian), do đó tăng tốc độ ghi. Tuy vậy, sau khi server crash, ta cần phải phục hồi dữ liệu bị hư hỏng bằng myisamchk.

**Một số đặc điểm khác:**

- Không hỗ trợ transaction.
- Không hỗ trợ row-level lock.
- MyISAM có thể lưu trữ dữ liệu tối đa lên tới 256TB.
- Không hỗ trợ ràng buộc với các bảng khác (khóa ngoại).
- Hỗ trợ đánh index (chỉ mục).

Đối với MySQL, trước phiên bản 5.5, **MyISAM** là dạng storage engine mặc định trong lúc tạo table bạn không chỉ rõ.

## 5.3 Memory engine

Còn được gọi là HEAP tables. Memory engine là dạng storage mà toàn bộ dữ liệu sẽ được lưu trên memory (RAM). Do được lưu trên memory, nên dữ liệu trong bảng này rất dễ bị mất khi gặp các vấn đề như bị crash ứng dụng, lỗi phần cứng, server reset, hay... mất điện. Vì thế dạng storage engine chỉ nên được sử dụng khi một bảng “tạm”, hoặc cache kết quả truy vấn từ một bảng khác.

Ưu điểm duy nhất của của storage engine này là do tốc độ truy xuất cao.

Đặc điểm:

- Sau khi server restart, cấu trúc bảng được bảo toàn, dữ liệu bị mất hết.
- Memory engine sử dụng HASH index nên rất nhanh cho query lookup.
- Memory engine dùng table-level locking do vậy tính concurrency không cao

## 5.4 Lựa chọn storage engine nào?

Mặc dù được chia làm nhiều loại storage engine (3 cái trên là 3 cái tiêu biểu), nhưng trong thực tế InnoDB gần như là storage engine duy nhất được sử dụng ở thời điểm hiện tại. Do nó là engine cân bằng giữa mọi yếu tố.

Mặt khác, nó cũng là engine mà MySQL thể hiện tốt nhất (để thể hiện điều này, thì trong version 8.0, MySQL cũng ưu ái khi giành riêng một mục lớn chỉ để nói về InnoDB, còn các storage engine khác được nhóm vào mục “Các storage engine thay thế khác”)

Sau đây là một vài tiêu chí lựa chọn engine:

- Transactions: Nếu ứng dụng yêu cầu transactions, InnoDB là lựa chọn duy nhất. Nếu không yêu cầu transactions, MyISAM là lựa chọn tốt.
- Concurrency: Nếu yêu cầu chịu tải cao và không cần thiết transactions, MyISAM là lựa chọn số 1.
- Sao lưu: Các engine đều phần nào hỗ trợ sao lưu. Ngoài ra ta cần hỗ trợ sao lưu trên cả quan điểm thiết kế hệ thống. Ví dụ: bạn thiết kế database server gồm master và slave, master yêu cầu transaction nên dùng innodb, slave cần sao lưu và đọc nên có thể dùng MyISAM. Cơ chế đồng bộ master-slave sẽ giúp bạn quản lý sự khác nhau giữa các engine nhưng đảm bảo tính sao lưu. Tiêu chí này có trọng số nhỏ.
- Phục hồi sau crash: MyISAM có khả năng phục hồi sau crash kém hơn InnoDB.