

1. Giới Thiệu mongoDB


2. Ứng dụng MongoDB cho CSDL đa phương tiện

- ❖ Tổng Quan MongoDB
- ❖ Hoạt động của MongoDB
- ❖ Ưu và Nhược điểm của MongoDB
- ❖ MongoDB với hệ quản trị CSDL có cấu trúc.



- ❖ Giới thiệu MongoDB
- ❖ Ứng dụng mongoDB cho cơ sở dữ liệu đa phương tiện





1. Giới Thiệu mongoDB

2. Ứng dụng MongoDB cho CSDL đa phương tiện

Tổng Quan MongoDB

- ❖ MongoDB là một cơ sở dữ liệu đa nền tảng
- ❖ Hoạt động trên các khái niệm Collection và Document
- ❖ Cung cấp hiệu suất xử lý cao, tính khả dụng cao và khả năng mở rộng dễ dàng.
- ❖ Hỗ trợ lưu trữ Multimedia file.

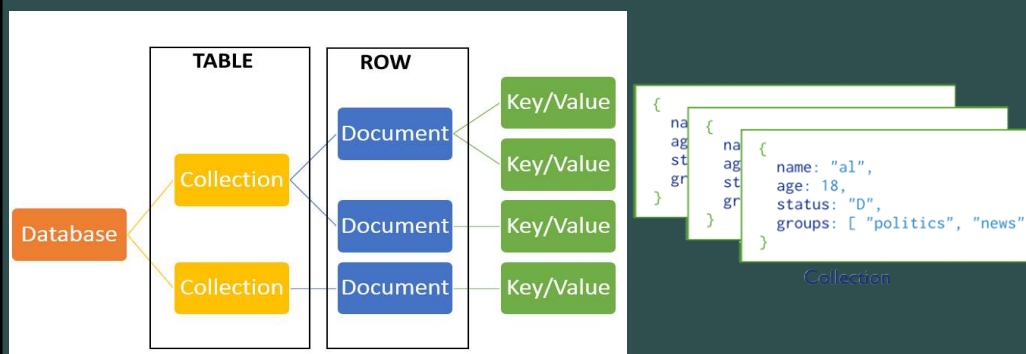


1. Giới Thiệu mongoDB
2. Ứng dụng MongoDB cho CSDL đa phương tiện

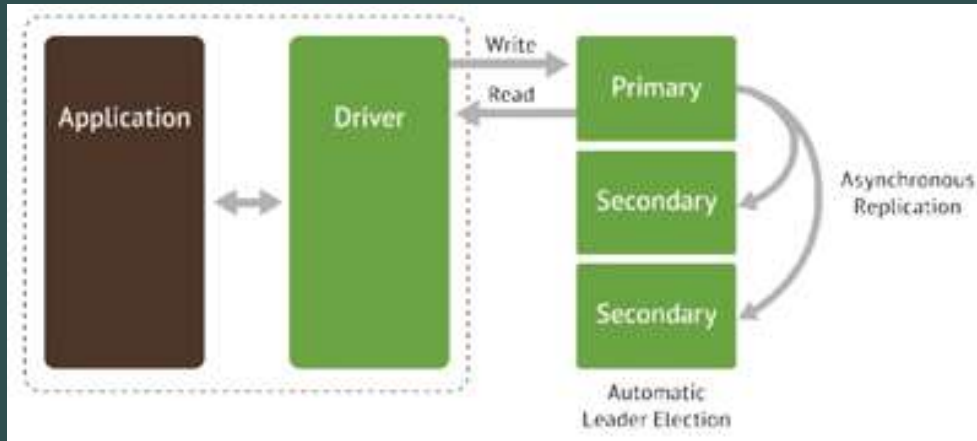
Tổng Quan MongoDB

- ❖ MongoDB không hỗ trợ Joins
- ❖ MongoDB không hỗ trợ Transactions
- ❖ MongoDB lưu dữ liệu theo định dạng JSON(hay gọi là BSON(Binary JSON))

MongoDB database structure



Hoạt động của MongoDB



Hoạt động của MongoDB

- ❖ MongoDB hoạt động dưới một tiến trình ngầm service luôn mở một cổng (Cổng mặc định là 27017) để lắng nghe các yêu cầu truy vấn, thao tác từ các ứng dụng gửi vào sau đó mới tiến hành xử lý.
- ❖ Mỗi một bản ghi của MongoDB được tự động gắn thêm một field có tên “_id” thuộc kiểu dữ liệu ObjectId mà nó quy định để xác định được tính duy nhất.

Hoạt động của MongoDB

- ❖ Trường dữ liệu “_id” luôn được tự động đánh index (chỉ mục) để tốc độ truy vấn thông tin đạt hiệu suất cao nhất.
- ❖ Mỗi khi có một truy vấn dữ liệu, bản ghi được cache (ghi đệm) lên bộ nhớ Ram, để phục vụ lượt truy vấn sau diễn ra nhanh hơn mà không cần phải đọc từ ổ cứng.

Hoạt động của MongoDB

- ❖ Khi có yêu cầu thêm/sửa/xóa bản ghi, để đảm bảo hiệu suất của ứng dụng mặc định MongoDB sẽ chưa cập nhật xuống ổ cứng ngay, mà sau 60 giây MongoDB mới thực hiện ghi toàn bộ dữ liệu thay đổi từ RAM xuống ổ cứng.



Tạo Database

- ❖ Database là một nơi chứa vật lý cho các Collection.
- ❖ Mỗi Database lấy tập hợp các file riêng của nó trên hệ thống file.
- ❖ Mỗi MongoDB Server có thể có 0 hoặc nhiều cơ sở dữ liệu.

Tạo Database

`use DATABASE_NAME`

VD: Tạo database tên test_mogodb

`Use test_mogodb`

Khái Niệm Collection

- ❖ Collection là một nhóm các Document và nó tương đương như một bảng trong RDBMS.
- ❖ Một Collection tồn tại bên trong một cơ sở dữ liệu duy nhất. Các Collection không có ràng buộc Relationship nên việc truy xuất rất nhanh.
- ❖ Mỗi collection có thể chứa nhiều thể loại khác nhau và trong một Collection có thể có nhiều trường khác nhau.

Tạo Collection

`db.createCollection(name, options)`

Name: tên collection

Options: là một Document và được sử dụng để xác định cấu hình cho Collection.

VD: **mycollection**

`db.createCollection("mycollection")`

```
>show collections
mycollection
system.indexes
```

Khái Niệm Document

❖ Một Document trong MongoDB, có cấu trúc tương tự như kiểu dữ liệu JSON, là một tập hợp các cặp key-value.

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

← field: value
← field: value
← field: value
← field: value

Khái Niệm Document

❖ Các Document có schema động, Document trong cùng một Collection không cần thiết phải có cùng một tập hợp các trường hoặc cấu trúc giống nhau, và các trường chung trong Document của một Collection có thể giữ các kiểu dữ liệu khác nhau..

Thêm Document

>**db.COLLECTION_NAME.insert(document)**

Ví dụ


```
>db.mycol.insert({  
  _id: ObjectId(7df78ad8902c),  
  title: 'MongoDB Overview',  
  description: 'MongoDB is no sql database',  
  by: 'tutorials point',  
  url: 'http://www.tutorialspoint.com',  
  tags: ['mongodb', 'database', 'NoSQL'],  
  likes: 100  
})
```

Truy vấn

db.COLLECTION_NAME.find()

- ❖ Phương thức find() sẽ hiển thị tất cả Document ở dạng không có cấu trúc .
- ❖ Để hiển thị các kết quả theo một cách đã được định dạng, bạn có thể sử dụng phương thức **pretty()**.

db.COLLECTION_NAME.find(). pretty()



1. Giới Thiệu mongoDB

Truy vấn

Ví dụ

```
>db.mycol.find().pretty()
{
  "_id": ObjectId(7df78ad8902c),
  "title": "MongoDB Overview",
  "description": "MongoDB is no sql database",
  "by": "tutorials point",
  "url": "http://www.tutorialspoint.com",
  "tags": ["mongodb", "database", "NoSQL"],
  "likes": "100"
}
```

	Truy vấn			1. Giới Thiệu mongoDB
Phép toán	Cú pháp	Ví dụ	Mệnh đề WHERE tương đương	
Equality	{<key>: <value>}	db.mycol.find({"by":"tutorials point"}).pretty()	where by = 'tutorials point'	
Less Than	{<key>: {\$lt: <value>}}	db.mycol.find({"likes": {\$lt:50}}).pretty()	where likes < 50	
Less Than Equals	{<key>: {\$lte: <value>}}	db.mycol.find({"likes": {\$lte:50}}).pretty()	where likes <= 50	
Greater Than	{<key>: {\$gt: <value>}}	db.mycol.find({"likes": {\$gt:50}}).pretty()	where likes > 50	
Greater Than Equals	{<key>: {\$gte: <value>}}	db.mycol.find({"likes": {\$gte:50}}).pretty()	where likes >= 50	
Not Equals	{<key>: {\$ne: <value>}}	db.mycol.find({"likes": {\$ne:50}}).pretty()	where likes != 50	

Phép toán AND trong Mongoddb

Cú pháp:

```
db.mycol.find({key1:value1, key2:value2}).pretty()
```

Phép AND

Ví dụ

Ví dụ hiển thị tất cả loạt bài hướng dẫn (tutorials) được viết bởi 'tutorials point' có title là 'MongoDB Overview'

```
>db.mycol.find({"by":"tutorials point","title": "MongoDB Overview"}).pretty()
{
  "_id": ObjectId("7df78ad8902c"),
  "title": "MongoDB Overview",
  "description": "MongoDB is no sql database",
  "by": "tutorials point",
  "url": "http://www.tutorialspoint.com",
  "tags": ["mongodb", "database", "NoSQL"],
  "likes": "100"
}
>
```

Phép toán OR trong Mongoddb

Cú pháp:

```
db.mycol.find(  
  { $or: [ {key1: value1}, {key2:value2} ] }  
).pretty()
```

Phép OR

Ví dụ sau sẽ hiển thị tất cả loạt bài hướng dẫn (tutorials) được viết bởi 'tutorials point' hoặc có title là 'MongoDB Overview'

```
>db.mycol.find({$or:[{"by":"tutorials point"},"title": "MongoDB Overview"}]).pretty()  
{  
  "_id": ObjectId(7df78ad8902c),  
  "title": "MongoDB Overview",  
  "description": "MongoDB is no sql database",  
  "by": "tutorials point",  
  "url": "http://www.tutorialspoint.com",  
  "tags": ["mongodb", "database", "NoSQL"],  
  "likes": "100"  
}
```

Ví dụ sau hiển thị các Document mà có các like lớn hơn 10 và có title là hoặc 'MongoDB Overview' hoặc bởi là 'tutorials point'.

```
>db.mycol.find({"likes": {$gt:10}, $or: [{"by": "tutorials point"}, {"title": "MongoDB Overview"}]}).pretty()
{
  "_id": ObjectId("7df78ad8902c"),
  "title": "MongoDB Overview",
  "description": "MongoDB is no sql database",
  "by": "tutorials point",
  "url": "http://www.tutorialspoint.com",
  "tags": ["mongodb", "database", "NoSQL"],
  "likes": "100"
}
```

Quy tắc thiết kế schema MongoDB

- ❖ One – few
- ❖ One - many
- ❖ One Squillions

One – few

Ưu điểm của việc nhúng dữ liệu là bạn chỉ cần 1 câu lệnh query để có thể lấy đầy đủ thông tin chi tiết bên trong tuy nhiên dữ liệu của bạn phụ thuộc vào Collection.

VD:One – few

Một ví dụ của "One to few" là việc bạn mô tả địa chỉ của một người.

```
> db.person.findOne()
{
  name: 'Kate Monster',
  ssn: '123-456-7890',
  addresses : [
    { street: '123 Sesame St', city: 'Anytown', cc: 'USA' },
    { street: '123 Avenue Q', city: 'New York', cc: 'USA' }
  ]
}
```



One – Many

1. Giới Thiệu mongoDB

Ví dụ : Bạn có 1 sản phẩm như Smartphone chẳng hạn. Nó bao gồm rất nhiều các linh kiện bên trong và các linh kiện này có thể thay thế được. Số linh kiện ở đây bị giới hạn cỡ vài trăm

```
> db.parts.findOne()
{
  _id : ObjectId('AAAA'),
  partno : '123-aff-456',
  name : '#4 grommet',
  qty: 94,
  cost: 0.94,
  price: 3.99
}
```

```
> db.products.findOne()
{
  name : 'left-handed smoke shifter',
  manufacturer : 'Acme Corp',
  catalog_number: 1234,
  parts : [ // array of references
    ObjectId('AAAA'),
    ObjectId('F17C'),
    ObjectId('D2AA'),
    // etc
  ]
}
```



1. Giới Thiệu mongoDB

VD:One – Many

Để có thể truy vấn dữ liệu từng linh kiện trong một sản phẩm xác định bạn có thể sử dụng cú pháp sau:

```
// Lấy thông tin sản phẩm qua catalog_number
> product = db.products.findOne({catalog_number: 1234});

// Lấy toàn bộ linh kiện trong sản phẩm
> product_parts = db.parts.find({_id: { $in : product.parts } }).toArray() ;
```

One – Squillions

Được sử dụng khi các mẫu ghi đủ lớn để tràn kích thước của dữ liệu là 16 MB kể cả khi sử dụng ObjectId

One – Squillions

Ví dụ hệ thống logging , nơi bạn lưu lại các message từ nhiều server khác nhau.

```
> db.hosts.findOne()
{
  _id : ObjectId('AAAB'),
  name : 'goofy.example.com',
  ipaddr : '127.66.66.66'
}

> db.logmsg.findOne()
{
  time : ISODate("2014-03-28T09:42:41.382Z"),
  message : 'cpu is on fire!',
  host: ObjectId('AAAB') // Reference to db.hosts['_id']
}
```



VD: One – Squillions

Để có thể truy vấn dữ liệu lấy được toàn bộ message từ một server có thể sử dụng cú pháp sau:

```
// tìm thằng server
> host = db.hosts.findOne({ipaddr : '127.66.66.66'}); // assumes unique index
// liệt kê 5000 message của thằng server
> last_5k_msg = db.logmsg.find({host: host._id}).sort({time : -1}).limit(5000).toArray()
```

Ưu điểm của MongoDB

- ❖ Ít Schema hơn
- ❖ Cấu trúc của một đối tượng là rõ ràng.
- ❖ Không có các Join phức tạp.
- ❖ Khả năng truy vấn sâu hơn. MongoDB hỗ trợ các truy vấn động trên các Document bởi sử dụng một ngôn ngữ truy vấn dựa trên Document mà mạnh mẽ như SQL.



MONGODB
Development & Administration

1. Giới Thiệu mongoDB

Ưu điểm của MongoDB

- ❖ MongoDB là dễ dàng để mở rộng.
- ❖ Sử dụng bộ nhớ nội tại để lưu giữ phần công việc, giúp truy cập dữ liệu nhanh hơn.
- ❖ Replication và tính hiệu năng cao.
- ❖ Cập nhật nhanh hơn.



MONGODB
Development & Administration

1. Giới Thiệu mongoDB
Nhược điểm của MongoDB

Nhược điểm của MongoDB

- ❖ Không ràng buộc, toàn vẹn nên không ứng dụng được cho các mô hình giao dịch yêu cầu độ chính xác cao.
- ❖ Không có cơ chế transaction (giao dịch) để phục vụ các ứng dụng ngân hàng.
- ❖ Dữ liệu được caching, lấy RAM làm trọng tâm hoạt động vì vậy khi hoạt động yêu cầu một bộ nhớ RAM lớn

MONGODB
Development & Administration

1. Giới Thiệu mongoDB
Nhược điểm của MongoDB

Nhược điểm của MongoDB

- ❖ thay đổi về dữ liệu mặc định đều chưa được ghi xuống ổ cứng ngay lập tức vì vậy khả năng bị mất dữ liệu từ nguyên nhân mất điện đột xuất là rất cao.

MONGODB
Development & Administration

mongoDB

1. Giới Thiệu mongoDB
MongoDB với các DBMS khác

MongoDB với các SQL Server 2008

- ❖ Sự tương quan các thuật ngữ được sử dụng trong MongoDB và SQL.
- ❖ Hiệu suất hoạt động của MongoDB và SQL Server 2008.

MONGODB
Development & Administration

Các thuật ngữ được sử dụng trong MongoDB và SQL

Operation	In SQL	In MongoDB
Create	Insert	Insert
Read	Select	Find
Update	Update	Update
Delete	Delete	Remove
Table	Table	Collection
Row	Row	Document

Hiệu suất hoạt động của MongoDB và SQL Server 2008.

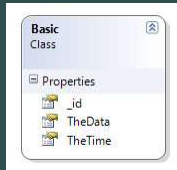
- ❖ So sánh tính năng insert
- ❖ So sánh tính năng truy xuất dữ liệu



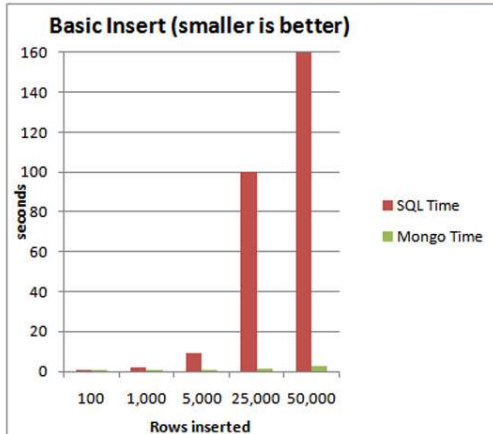
1. Giới Thiệu mongoDB MongoDB với các DBMS khác

So sánh insert.

- ❖ So sánh thời gian thực thi của hai hệ quản trị CSDL MongoDB và SQL Server 2008 khi thêm 50.000 dòng dữ liệu



		Number of Parallel Clients		Time in seconds			
Basic Insert	Total Rows	Rows / client	SQL Time	Mongo Time	Sql Ops/sec	Mongo Ops/sec	
several columns	100	20	0.19	0.011	526	9,091	
600 bytes per row.	1,000	200	1.8	0.02	556	50,000	
	5,000	1,000	9	0.25	556	20,000	
	25,000	5,000	100	1.5	250	16,667	
	50,000	10,000	270	2.5	185	20,000	



1. Giới Thiệu mongoDB MongoDB với các DBMS khác

Thời gian thêm 10.000 dòng của MongoDB.

```

MongoDB Client
Warming up ...
Building insert data...
Waiting on mutex
Running!
Finished with 10000 MongoDB inserts in 2.032 sec.
Done
  
```



1. Giới Thiệu mongoDB MongoDB với các DBMS khác

Thời gian thêm 10.000 dòng của SQL.

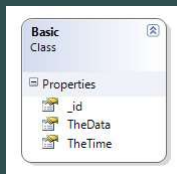
```
SQL Server Client
Warming up ...
Building insert data...
Waiting on mutex
Running!
Finished with 10000 SQL inserts in 204.215 sec.
Done
```



1. Giới Thiệu mongoDB MongoDB với các DBMS khác

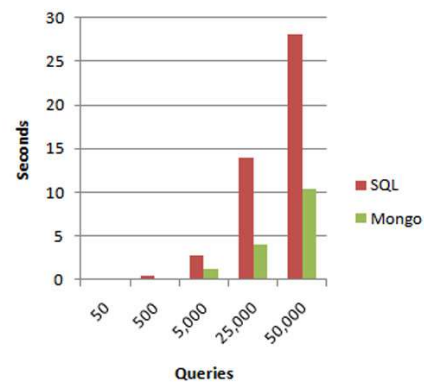
So sánh truy xuất.

- ❖ So sánh thời gian truy xuất của hai hệ quản trị CSDL MongoDB và SQL Server 2008

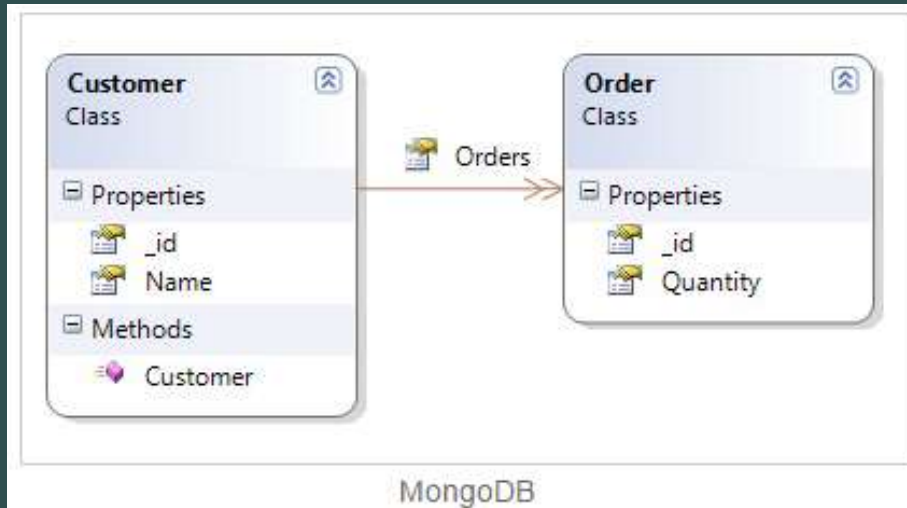


	Number of Parallel Clients		Time in seconds					
	Total Rows	Rows / client	SQL Time	Mongo Time	Sql Ops/sec	Mongo Ops/sec		
Basic Query with index	50	10	0.1	0.08	500	625		
	500	100	0.38	0.1	1,316	5,000		
	5,000	1,000	2.8	1.2	1,786	4,167		
	25,000	5,000	14	4	1,786	6,250		
	50,000	10,000	28	10.4	1,786	4,808		

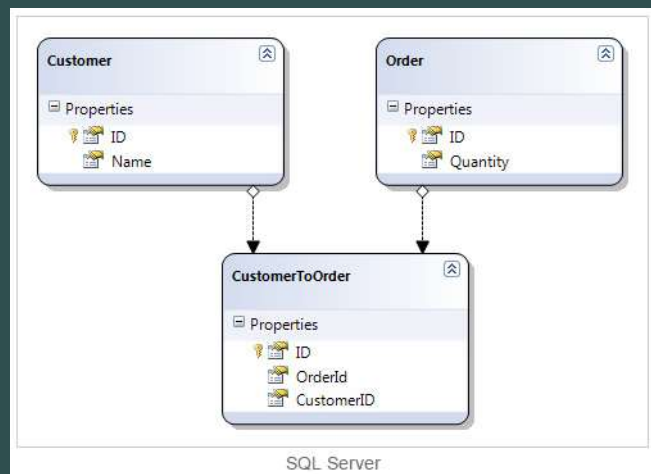
Basic Queries (smaller is better)



So sánh truy xuất có sử dụng khóa ngoại .



So sánh truy xuất có sử dụng khóa ngoại (TT) .



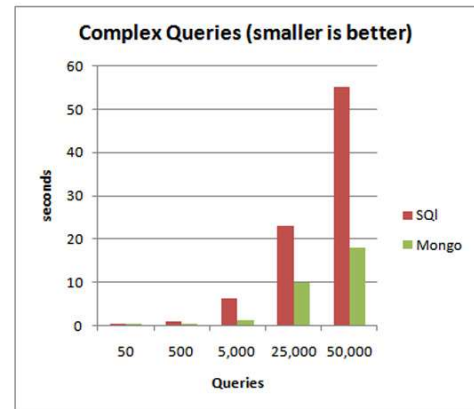


1. Giới Thiệu mongoDB MongoDB với các DBMS khác

So sánh truy xuất.

Nhận xét: kết quả không thay đổi so với truy vấn ban đầu.

Number of Parallel Clients		Time in seconds				
	Total Rows	Rows / client	SQL Time	Mongo Time	Sql Ops/sec	Mongo Ops/sec
Complex / Real w	50	10	0.1	0.066	500	758
query (2 joins in S	500	100	1	0.19	500	2,632
embedded doc in	5,000	1,000	6.4	1.2	781	4,167
MongoDB)	25,000	5,000	23	9.9	1,087	2,525
	50,000	10,000	55	18	909	2,778
	1,000,000	200,000	960	398	1,042	2,513



Tài Liệu Tham Khảo

1. <http://vietjack.com/mongodb/>
2. <https://www.mongodb.com/blog/post/storing-large-objects-and-files-in-mongodb>.
3. <http://kienthucweb.net/gioi-thieu-ve-co-so-du-lieu-mongodb.html>
4. <https://docs.mongodb.com/v3.2/reference/program/mongofiles/>
5. <http://expressmagazine.net/development/2324/so-sanh-hieu-suat-hoat-dong-cua-mongodb-va-sql-server-2008>