MongoDB

Agenda

- What is MongoDB
- Why MongoDB
- MongoDB Concepts/Commands

1.

What is MongoDB

What is MongoDB

- Opensource
- Cross-platform
- Non-Relational
- NOSQL
- Document-oriented data store
- Distributed

2.

Why MongoDB

Why MongoDB

◎ **Fast, Iterative Development**

Scope creep and changing business requirements no longer stand between you and successful project delivery. A flexible data model coupled with dynamic schema and idiomatic drivers make it fast for developers to build and evolve Applications.

◎ **Flexible Data Model**

MongoDB's document data model makes it easy for you to store and combine data of any structure, without giving up sophisticated validation rules, data access and rich indexing functionality. You can dynamically modify the schema without downtime. You spend less time prepping your data for the database, and more time putting your data to work.

Why MongoDB

◎ **Multi-Datacenter Scalability**

MongoDB can be scaled within and across geographically distributed data centers, providing new levels of availability and scalability. As your deployments grow in terms of data volume and throughput, MongoDB scales easily with no downtime, and without changing your application. And as your availability and recovery goals evolve, MongoDB lets you adapt flexibly, across data centers, with tunable consistency

◎ **Integrated Feature Set**

Analytics and data visualization, text search, graph processing, geospatial, in-memory performance and global replication allow you to deliver a wide variety of real-time applications on one technology, reliably and securely. RDBMS systems require additional, complex technologies demanding separate integration overhead and expense to do this well.

Why MongoDB

◎ Lower TCO

Application development teams are more productive when they use MongoDB. Single click management means operations teams are as well. MongoDB runs on commodity hardware, dramatically lowering costs. Finally, MongoDB offers affordable annual subscriptions, including 24x7x365 global support. Your applications can be one tenth the cost to deliver compared to using a relational Database.

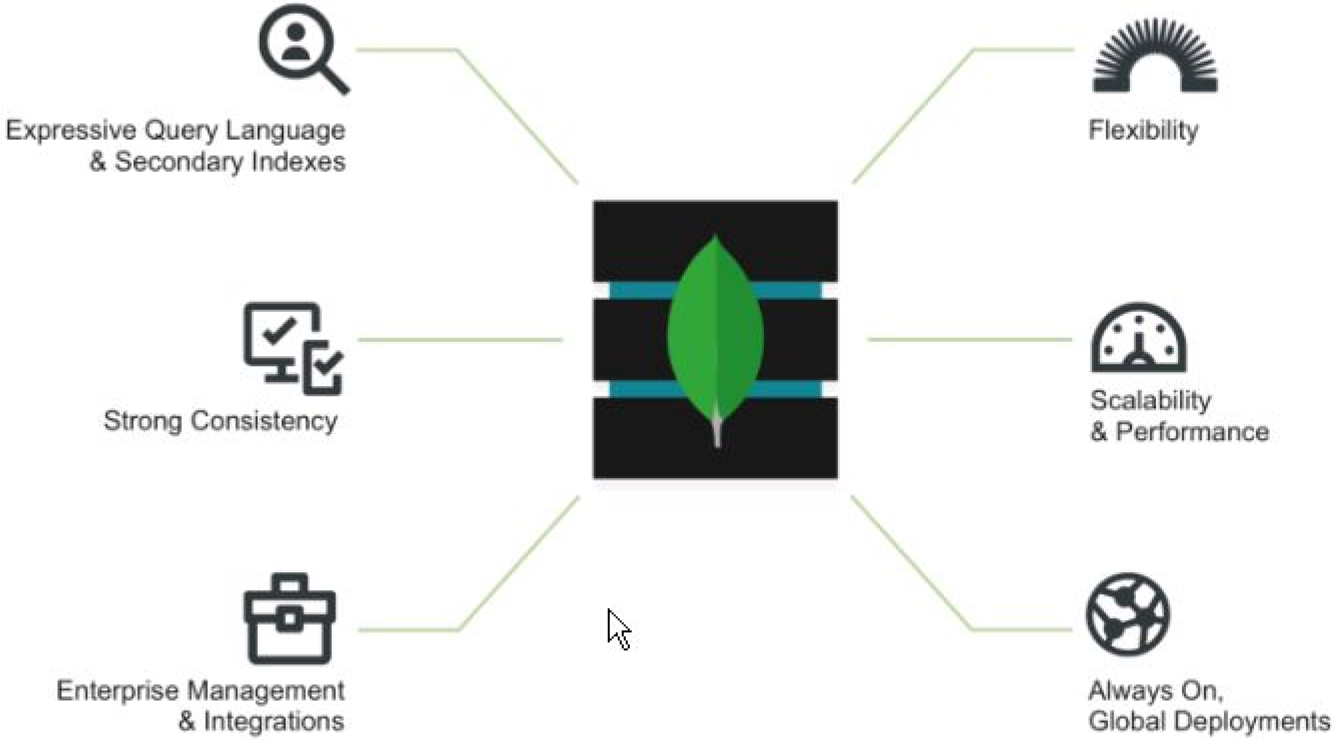
◎ Long-Term Commitment

MongoDB Inc and the MongoDB ecosystem stand behind the world's fastest-growing database. 20M+ downloads and 2,000+ customers including more than 50% of the Fortune 100. Over 1,000 partners and greater investor funding than any other database in history. You can be sure your investment is protected.

3.

MongoDB Concepts
&
Commands

The Nexus Architecture



Expressive Query Language
& Secondary Indexes

Strong Consistency

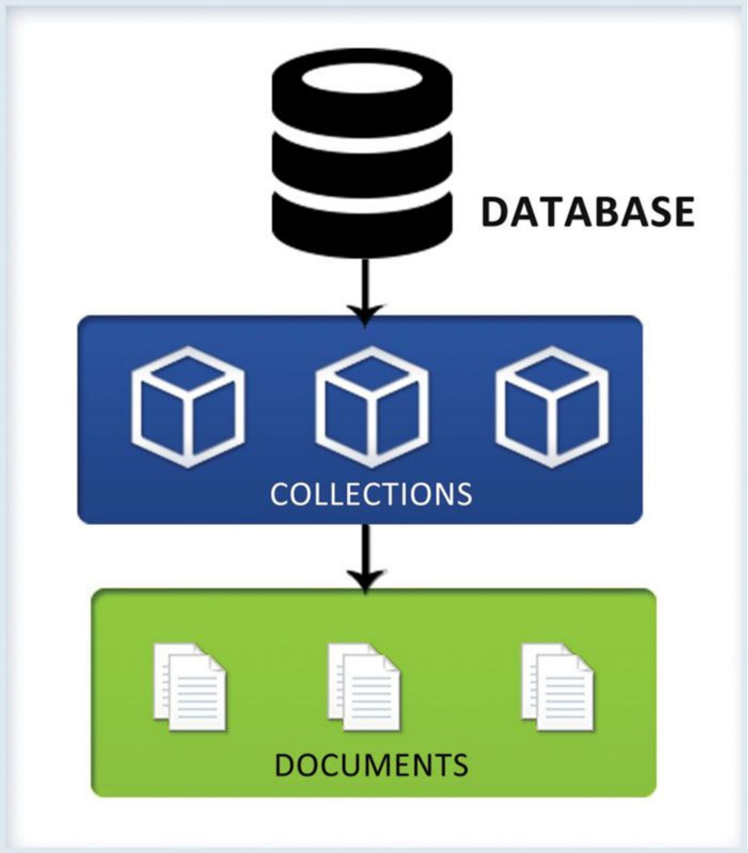
Enterprise Management
& Integrations

Flexibility

Scalability
& Performance

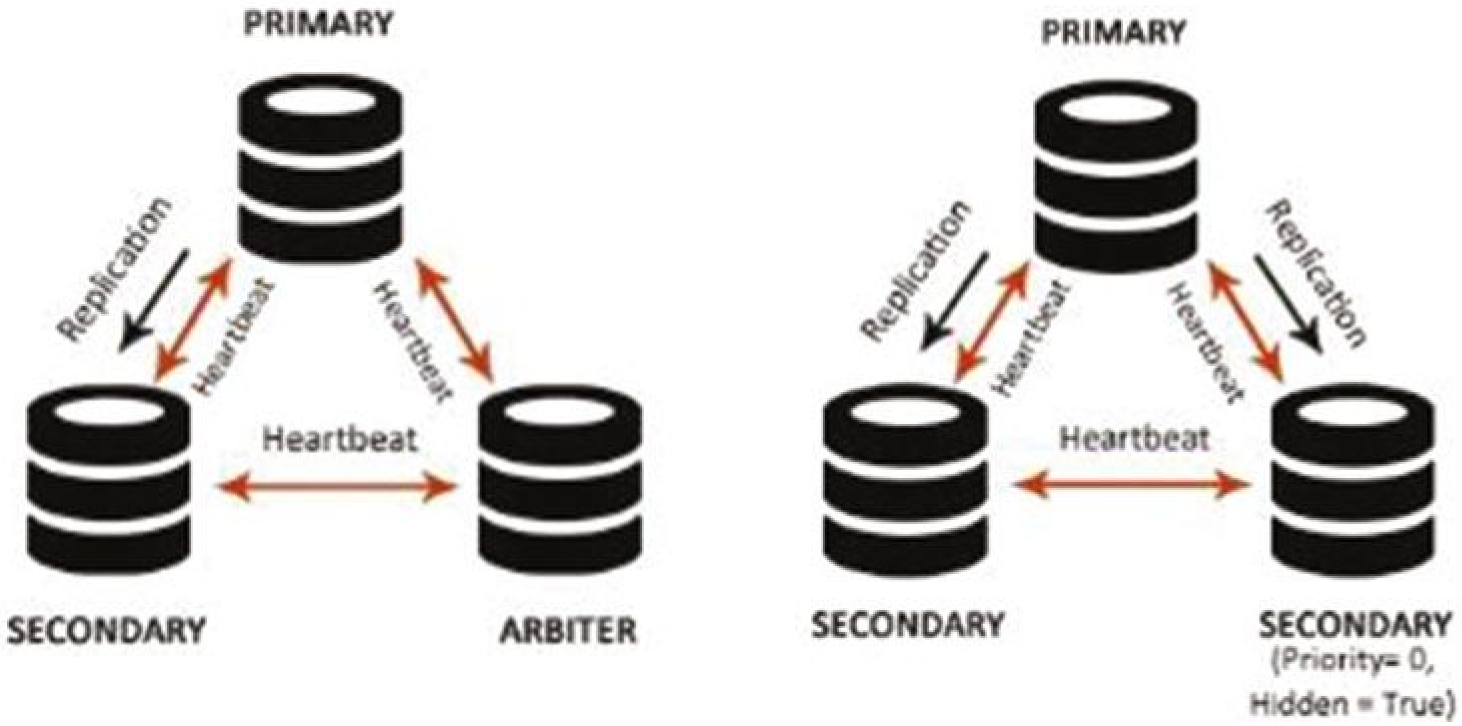
Always On,
Global Deployments

MongoDB Data Model

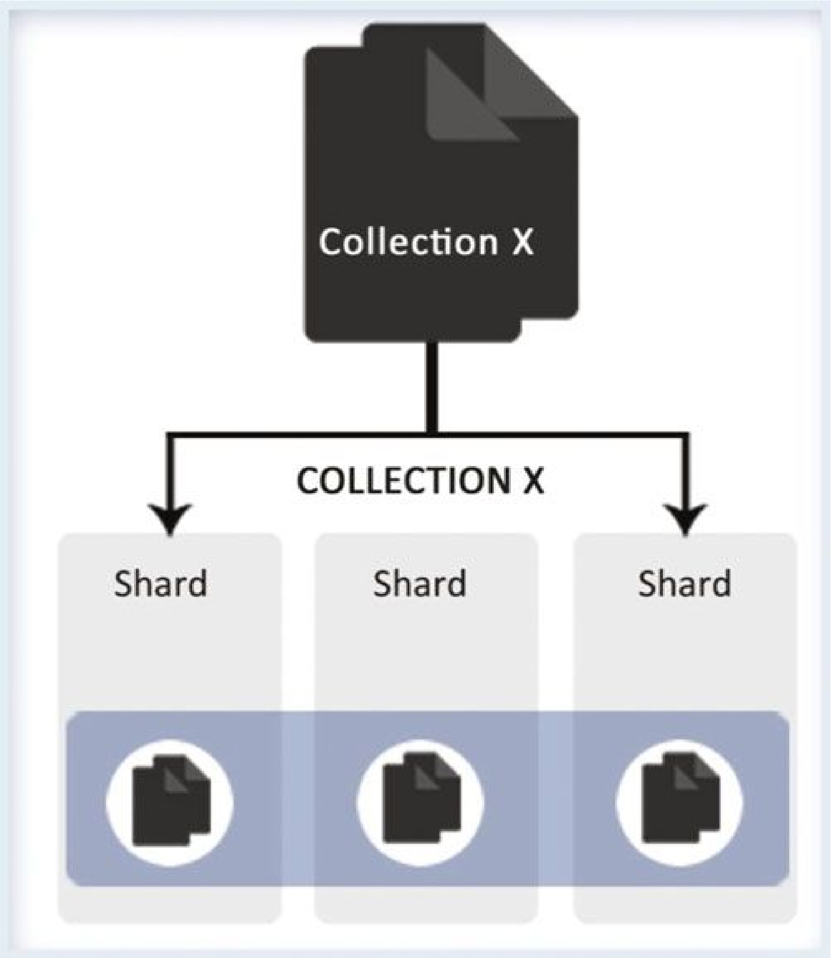


RDBMS	MongoDB
Table	Collection
Row	Document
Column	Field
Relationships	Linking and Embedding documents

MongoDB Replication



MongoDB Sharding



MongoDB Commands

	RDBMS	MongoDB
Create	CREATE TABLE `posts` (`id` int(11) NOT NULL AUTO_INCREMENT,`post_text` varchar(500) NOT NULL,`user_name` varchar(20) NOT NULL,`post_privacy` varchar(10) NOT NULL,`post_likes_count` int(11) NOT NULL,PRIMARY KEY (`id`))	db.createCollection("posts")
Insert	INSERT INTO `posts` (`id` ,`post_text` ,`user_name` ,`post_privacy` ,`post_likes_count`)VALUES (NULL , 'This is a sample post', 'mark', 'public', '0');	db.posts.insert({user_name:"mark", post_text:"This is a sample post", post_privacy:"public", post_likes_count:0})
Read	SELECT * FROM `posts	db.posts.find()

MongoDB Commands

	RDBMS	MongoDB
Conditional Search	SELECT * FROM `posts` WHERE `user_name` = 'mark'	db.posts.find({user_name:"mark"},{post_text:1,post_likes_count:1})
Conditional Search AND	SELECT `post_text` , `post_likes_count` FROM `posts` WHERE `user_name` = 'mark' AND `post_privacy` = 'public'	db.posts.find({user_name:"mark",post_privacy:"public"},{post_text:1,post_likes_count:1}
Conditional Search OR	SELECT `post_text` , `post_likes_count` FROM `posts` WHERE `user_name` = 'mark' OR `post_privacy` = 'public'	db.posts.find({$or:[{user_name:"mark"},{post_privacy:"public"}]},{post_text:1,post_likes_count:1})

MongoDB Commands

	RDBMS	MongoDB
Sort ASC	SELECT * FROM `posts` WHERE `user_name` = 'mark' order by post_likes_count ASC	db.posts.find({user_name:"mark"}).sort({post_likes_count:1})
Sort DESC	SELECT * FROM `posts` WHERE `user_name` = 'mark' order by post_likes_count DESC	db.posts.find({user_name:"mark"}).sort({post_likes_count:-1})
Limit	SELECT * FROM `posts` LIMIT 10	db.posts.find().limit(10)
Skip	SELECT * FROM `posts` LIMIT 10 OFFSET 5	db.posts.find().limit(10).skip(5)

MongoDB Commands

	RDBMS	MongoDB
Update	UPDATE posts SET post_privacy = "private" WHERE user_name='mark'	db.posts.update({user_name:"mark"},{$set:{post_privacy:"private"}},{multi:true})
Delete	DELETE FROM posts WHERE user_name='mark'	db.posts.remove({user_name:"mark"})
Index	CREATE INDEX index_posts ON posts(user_name,post_likes_count DESC)	db.posts.ensureIndex({user_name:1,post_likes_count:-1})
Show Index	SHOW INDEX FROM posts	db.posts.getIndexes()

Thanks!

Any questions?

You can find me at:
@geekduo
gkedar@gmail.com