Large-Scale and Multi-Structured Databases

Introduction to MongoDB

Prof Pietro Ducange

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

UNIVERSITÀ DI PISA

CROSSLAB Innovation for industry 4.0

Let's Start!

- mongoDB stands for "Hu**mongo**us **DB**"
  - Open-source
  - Document-based
  - "High performance, high scalability"
  - Different configurations on CAP triangle (CP and AP mainly)

Data Model

- Document-Based (max 16 MB)
- Documents are in ***BSON format***, consisting of field-value pairs
- Each document stored in a ***collection***
- Collections
  - are like ***tables*** of relational DBs.
  - their documents do ***not*** need to have ***uniform structures***
  - have ***index set*** in common

JSON

- “JavaScript Object Notation”
- ***Easy for humans*** to write/read, ***easy for computers*** to parse/generate
- JSON document is an unordered collection of “field: **value**” pairs
- ***6 main data types*** (string, number, object, array, boolean, null)
- Objects can be ***nested***
- JSON objects fields are not ordered, but array elements are.

BSON

- BSON stands for **Binary JSON**
- ***Binary-encoded serialization*** of JSON-like docs
- BSON extends the JSON model to provide ***additional data types***, ***ordered fields***, and to be ***efficient*** for encoding and decoding within different languages
- The MongoDB BSON implementation is ***lightweight***, ***fast*** and ***highly traversable***

BSON Types

Type	Alias	Notes
Double	"double"	
String	"string"	
Object	"object"	
Array	"array"	
Binary data	"binData"	
Undefined	"undefined"	Deprecated.
ObjectId	"objectId"	
Boolean	"bool"	
Date	"date"	
Null	"null"	

Type	Alias	Notes
Regular Expression	"regex"	
DBPointer	"dbPointer"	Deprecated.
JavaScript	"javascript"	
Symbol	"symbol"	Deprecated.
JavaScript code with scope	"javascriptWithScope"	Deprecated in MongoDB 4.4.
32-bit integer	"int"	
Timestamp	"timestamp"	
64-bit integer	"long"	
Decimal128	"decimal"	New in version 3.4.

JSON types

Check the doc for more details.

Documents in Mongo

The advantages of using documents are:

- Documents (i.e. objects) correspond to ***native data types*** in many programming languages.
- ***Embedded documents*** and ***arrays*** reduce need for expensive joins.
- ***Dynamic schema*** supports fluent ***polymorphism***.

The _id Field

By default, each document contains an **_id** field. This field has several special characteristics:

- The value serves as ***primary key*** for collection.
- The value is ***unique***, ***immutable***, and may be any non-array type.
- Default data type is ***ObjectId***, which is "small, likely unique, fast to generate, and ordered."
- ***Sorting on an ObjectId*** value is roughly equivalent to ***sorting on creation time.***

Key Features (I)

High Performance

- MongoDB provides high performance data persistence.
- Support for ***embedded data models*** reduces I/O activity on database system.
- ***Indexes*** support ***faster queries*** and can include keys from embedded documents and arrays.

Rich Query Language

MongoDB supports a ***rich query language*** to support read and write operations (CRUD) as well as:

- Data Aggregation
- Text Search and Geospatial Queries.

Key Features (II)

High Availability

MongoDB's replication facility, called ***replica set***, provides:

- automatic failover
- data redundancy.

Horizontal Scalability

MongoDB provides horizontal scalability as part of its core functionality:

- ***Sharding*** distributes data across a ***cluster of machines***.
- MongoDB supports creating ***zones*** of data based on the ***shard key***.
- In a balanced cluster, MongoDB directs reads and writes covered by a zone only to those shards inside the zone.

Core Processes

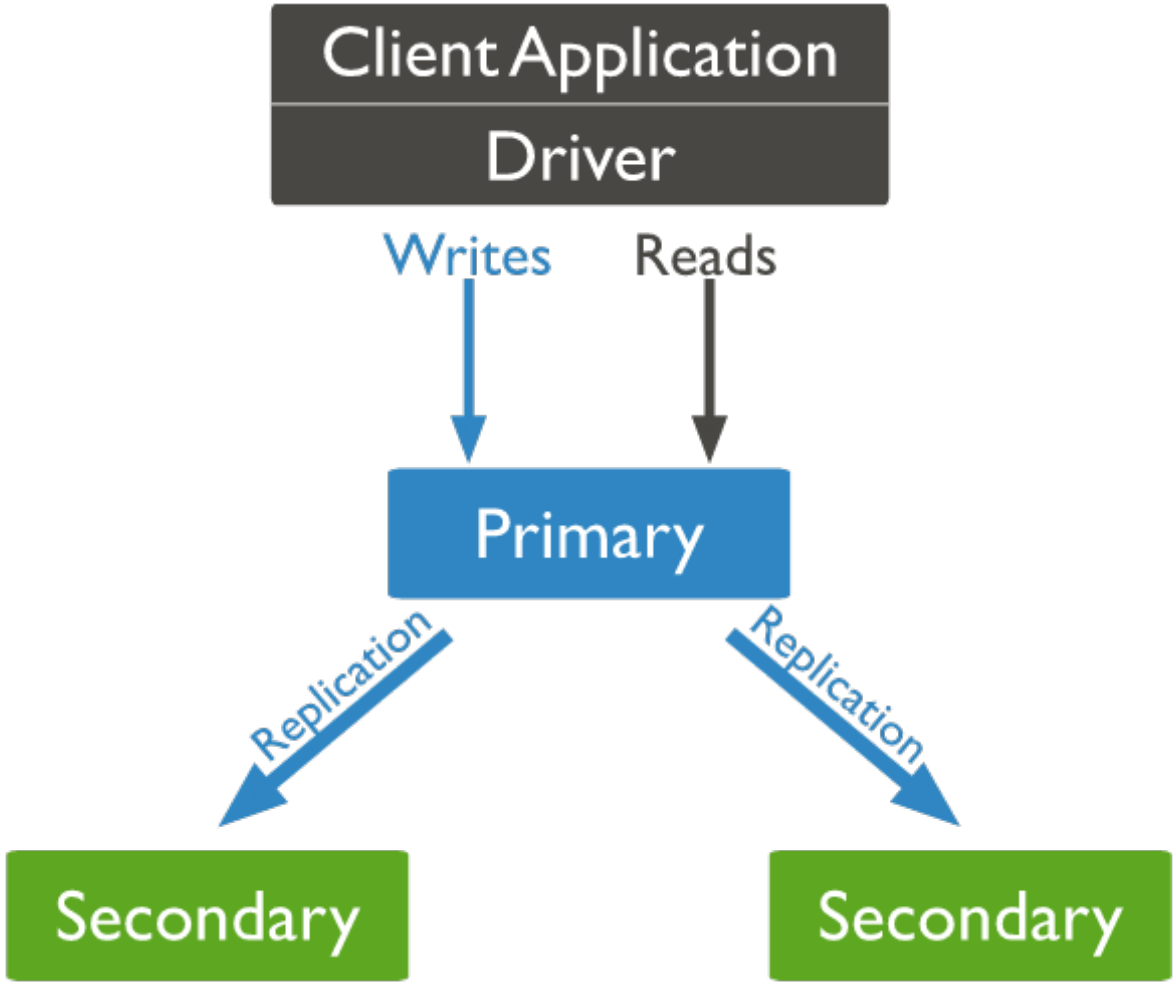
The core components in the MongoDB package are:

- ***mongod:*** the main daemon process for the MongoDB system. It handles data requests, manages data access, and database process (doc) ;
- ***mongos:*** the controller and query router interfaced between client applications and the sharded cluster (doc);

Old Mongo Versions:

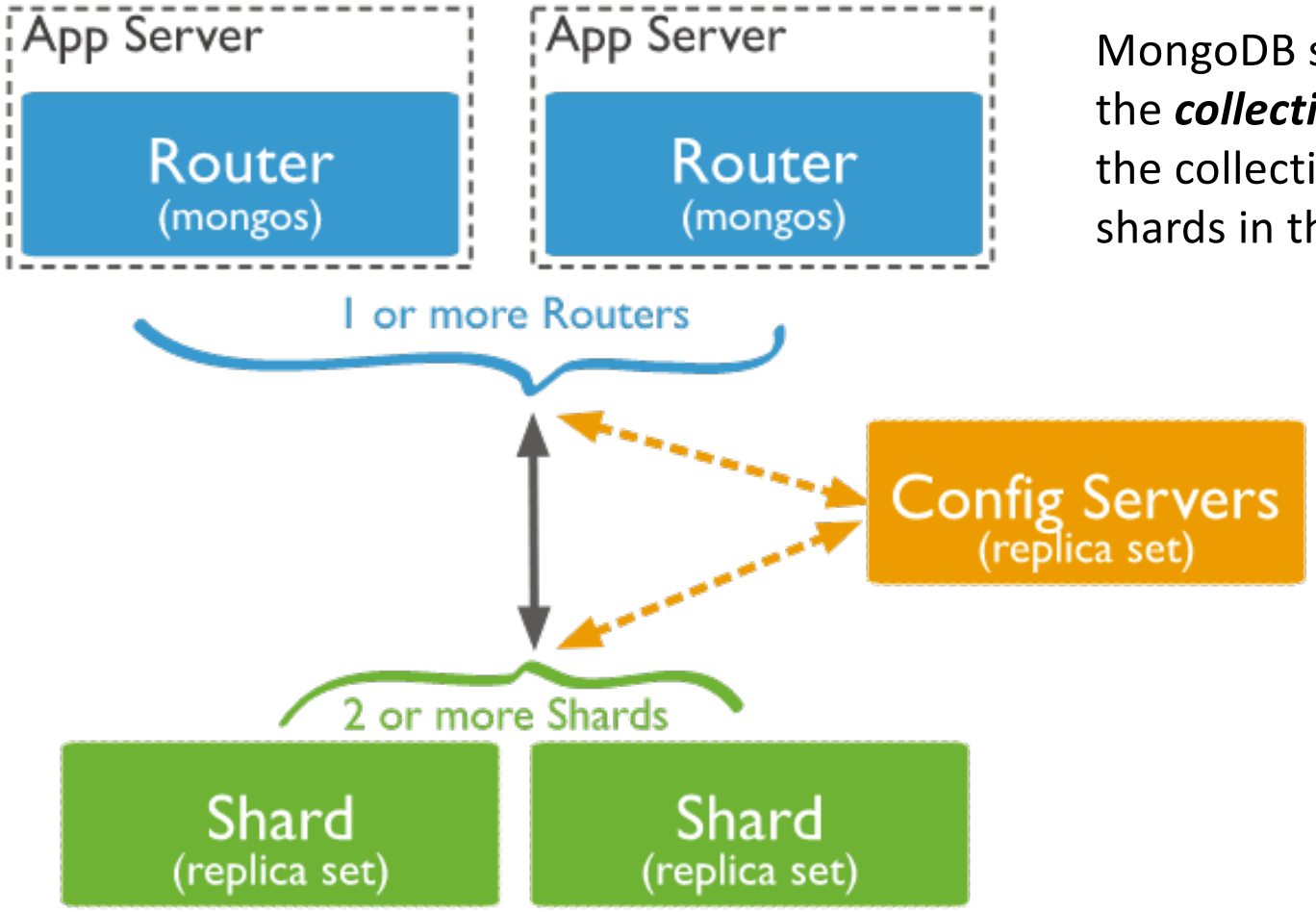
- ***mongo:*** the interactive MongoDB Shell. The MongoDB Shell is transitioning to **mongosh** which is shipped with MongoDB Compass.

Replica Architecture



https://www.mongodb.com/docs/manual/replication/

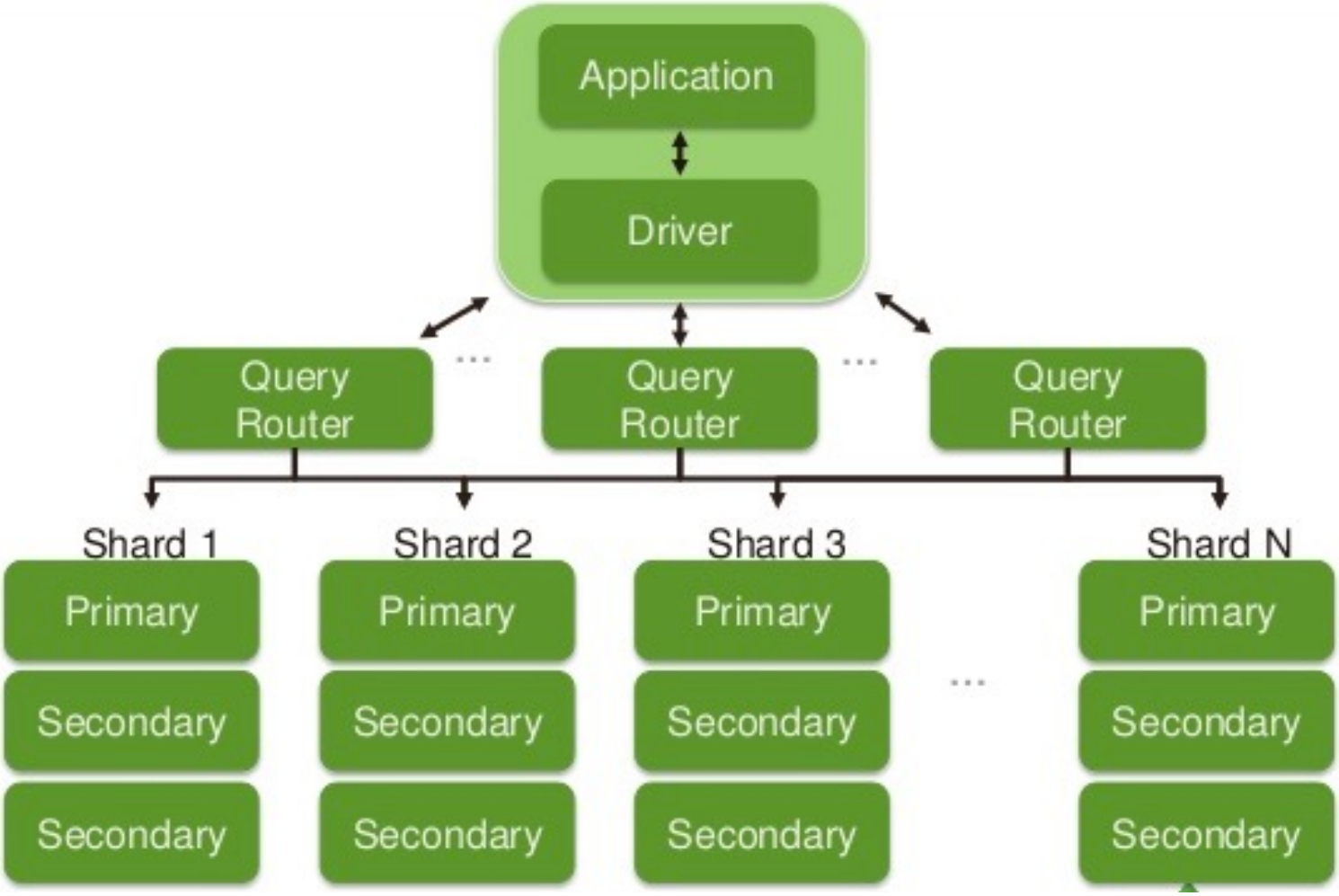
Sharding Architecture



MongoDB shards data at the ***collection level***, distributing the collection data across the shards in the cluster.

https://www.mongodb.com/basics/sharding

Overall MongoDB Architecture



MongoDB vs. SQL

mongoDB	SQL
Document	Tuple
Collection	Table/View
PK: _id Field	PK: Any Attribute(s)
Uniformity not Required	Uniform Relation Schema
Index	Index
Embedded Structure	Joins
Shard	Partition

Installing MongoDB

- Install ***Community MongoDB*** selecting the distribution which suits with your operating systems. Check official documentation (link)

- Install the *latest version* (as of today 6.0)

Configuration File and Other Files

DEFAULT CONFIGURATION FILE

- On Linux, a default ***/etc/mongod.conf*** configuration file is included when using a package manager to install MongoDB.
- On Windows, a default ***<install directory>/bin/mongod.cfg*** configuration file is included during the installation.
- On macOS, a default ***/usr/local/etc/mongod.conf (/opt/homebrew/etc/mongod.conf (on Apple M1 processors) c***onfiguration file is included when installing from MongoDB's official Homebrew tap.

Please, refer to the documentation section about configuration files and log and data directories.

Mongd DB Server

- We consider a ***localhost installation*** where ***27017*** is the default port the standalone mongod listens on.

- Check the commands for running and stopping mongod server on different Operating Systems.

- Use `--port` to set a specific port

Mongo Shell

- The mongo shell is an ***interactive JavaScript*** interface to MongoDB.
- We can use the mongo shell to ***query and update data*** as well as perform ***administrative operations***.
- To connect to a MongoDB instance:

```
mongosh "mongodb://localhost:27017"

mongosh "mongodb://mongodb0.example.com:28015" --
username alice --authenticationDatabase admin
```

- To shutdown the server (from the shell)

```
db.shutdownServer()
```

Basics of Mongo Shell

- Check the list of databases:

  show databases (or show dbs)

- Display the database currently used:

  db

- Switch databases (*<database> is the name of a specific database*):

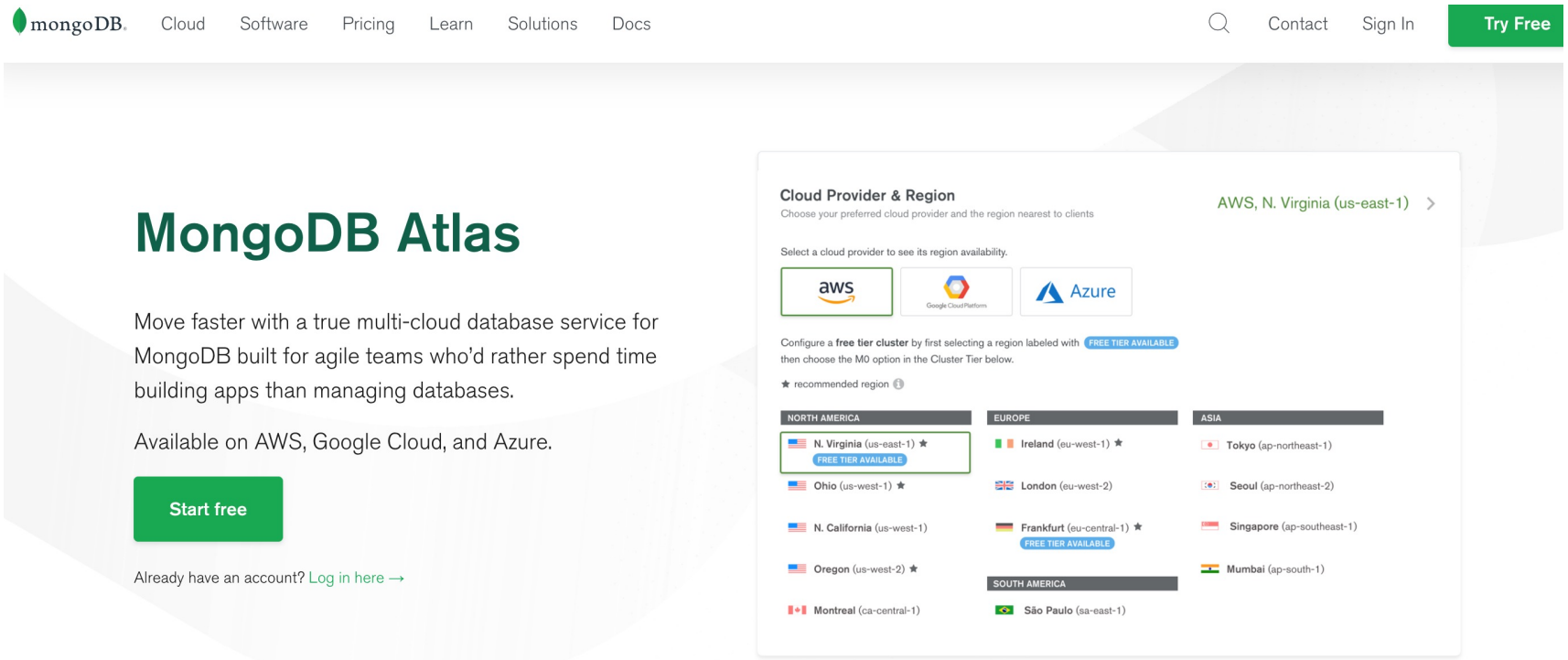
  use <database>

- Create both the database *myNewDatabase* and the collection *myCollection* during the insertOne() operation:

  use myNewDatabase
  db.myCollection.insertOne({ x: 1 });

Mongo ATLAS: Cloud Service

Available at: https://www.mongodb.com/cloud/atlas



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

UNIVERSITÀ DI PISA

CROSSLAB Innovation for industry 4.0

Suggested Readings

Most of the material used in this class has been extracted from the MongoDB official documentation.

Student should read the links highlighted along the previous slides and the following documents:
https://docs.mongodb.com/manual/introduction/

https://www.mongodb.com/docs/manual/core/document/

https://www.mongodb.com/collateral/mongodb-architecture-guide

Reference book for MongoDB Classes:
MongoDB: The Definitive Guide, 3rd Edition, Powerful and Scalable Data Storage