Contents

- *What is BIG DATA*
- *Handling Steps of Big Data*
- *Dimensions (V's) of Big Data*
- *Cons of RDBMS*
- *Need of Unstructured Data*
- *NoSQL*
- *CAP Theorem*
- *NoSQL Data Models and Processing Tools*
- *MongoDB Vs RDBMS*
- *Practical Examples of NoSQL*

What is BIG DATA

- *“A massive volume of both structured and unstructured data that is so large that it's difficult to process with traditional database and software techniques.”* [1]
- *Web sites with **300+** million unique visitors/month.*
- *Criteria for considering data as big data*

  Size

  Type of data

  Latency

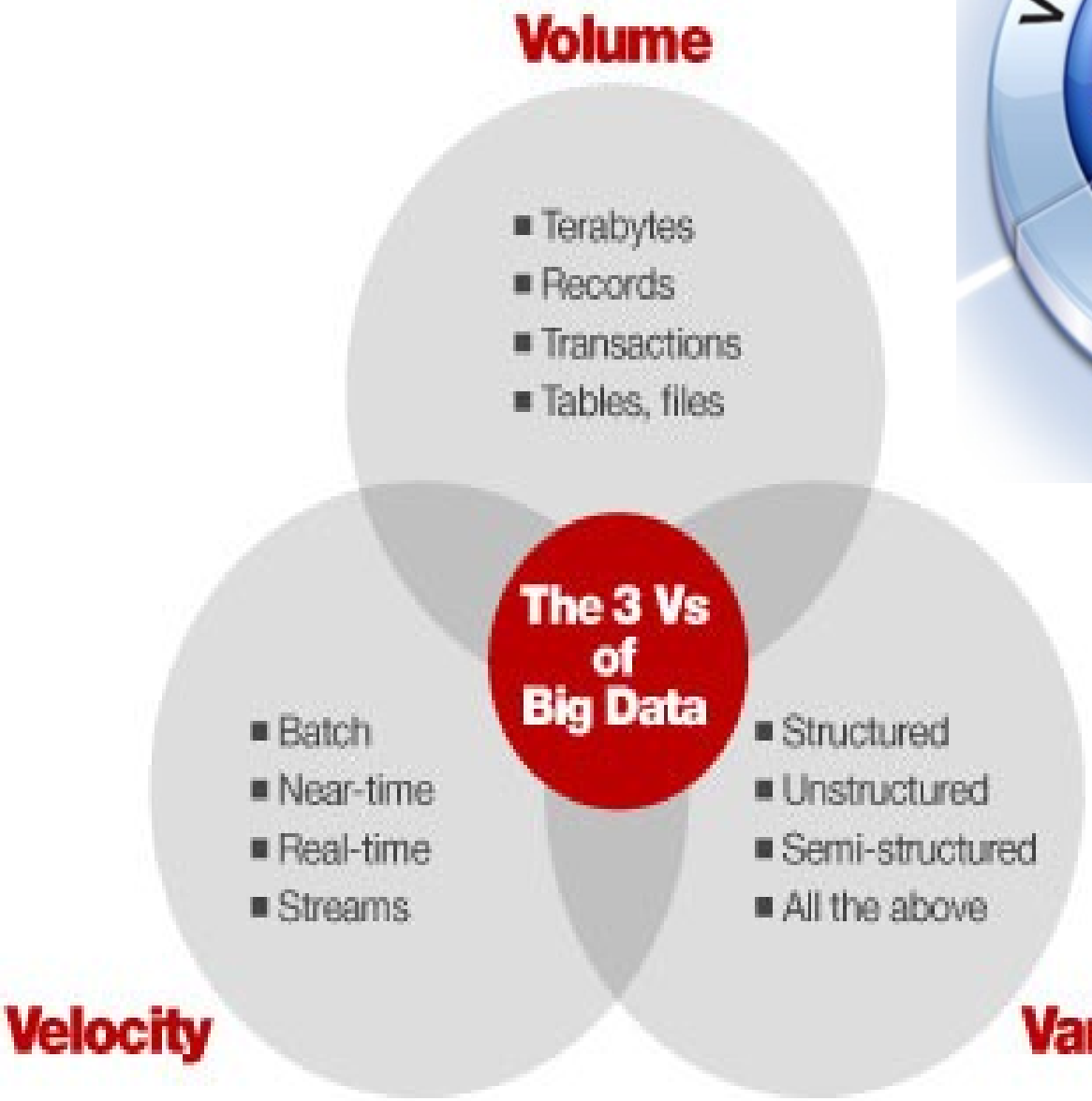
  Data complexity
- *Digital data from sensors used to gather climate information*
- *cell phone GPS signals*
- *Posts to social networking sites*

Handling Big Da

- *Storage*
- *Processing*
- *Analysis*
- *Security*

4.6 billion camera phones world wide

3 Dimensions of Big Da



Source: Celent

Cons of RDBMS

- *Rigid schema design.*
- *Harder to scale.*
- *Replication.*
- *Join across multiple nodes is hard*
- *Handling data growth using RDBMS is difficult*
- *Need for a DBA.*
  - *Object Relational Mapping doesn't work quite well.*
  - *Only structured database like table form is handled*
  - ***ACID** transaction*
  - *Hence slower processing*

Need of unstructured data

- *Need of databases which are able to store and process big data effectively.*
- *demand for high performance when reading and writing.*
- *high concurrency applications.*
- *Easy to expand*
- *Big data analysis*
- *High scalability*
- *Data format.*
- *Manageability.*

NoSQL (continued..) [2]

- *Stands for **N**ot **O**nly **SQL***
- *Class of non-relational data storage systems*
- *Usually do not require a fixed table*
  - *Scales well for both reads and writes*
- *BASE property*
- *Auto - Sharding*
- *Supporting mass storage.*
- *Flexible schema and data types.*
- *Fast key value look ups.*
- *Easy maintenance.*
- *Large scalability.*

CAP Theorem

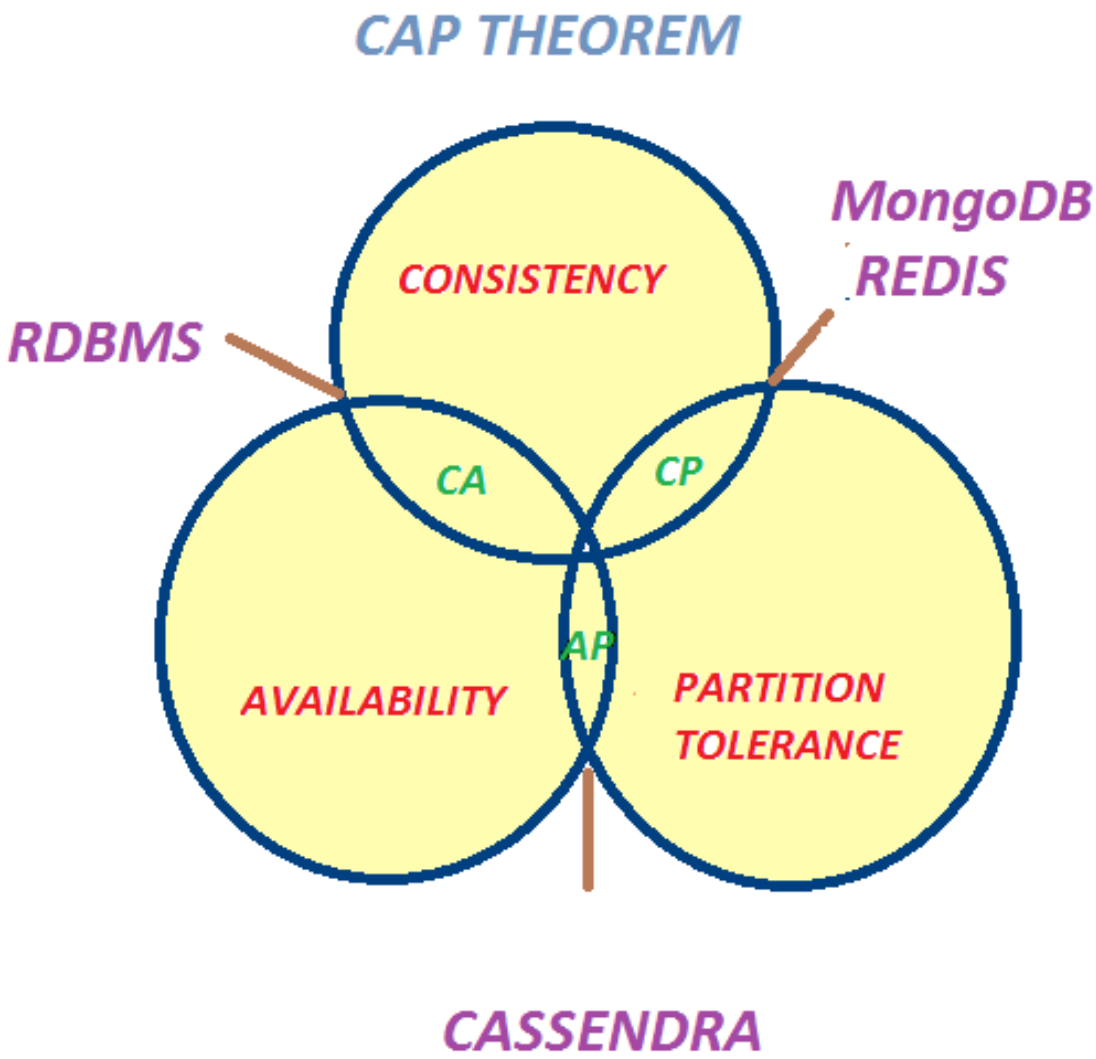
- *Also known as Brewer's Theorem by Prof. Eric Brewer, published in 2000 at University of Berkeley.* [2]
- *"Of three properties of a shared data system: data consistency, system availability and tolerance to network partitions,* ***only two*** *can be achieved at any given moment." [2]*
- *NoSQL database provides* ***BASE*** *property.*
- ***C****onsistency* ***-*** *all nodes see the same data at the same time*
  - *Strict Consistency – RDBMS.*
  - *Tunable Consistency – Cassandra.*
  - *Eventual Consistency – Amazon Dynamo*
- ***A****vailability*
- ***P****artition Tolerance*
- *Weaker consistency (Eventual), Best effort, Simple and fast, Optimistic.*

BASE Properties of CAP theorem

Basically available:
Nodes in the a distributed environment can go down, but the whole system shouldn't be affected.

Soft State (scalable):
The state of the system and data changes over time.

Eventual Consistency:
Given enough time, data will be consistent across the distributed system.



NoSQL Data Models

- ***Key-value type (Redis)***

  value corresponds to a Key.

- ***Column-based (Cassandra)***

  database using Table. more suitable application on aggregation and data warehouse.

- ***Document-type(MongoDB)***

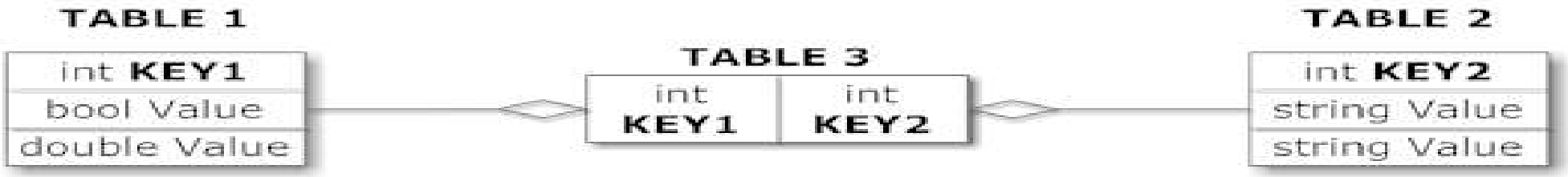
  No table structure is used.

- ***Graph-based (Neo4J)***

  store an information about networks.

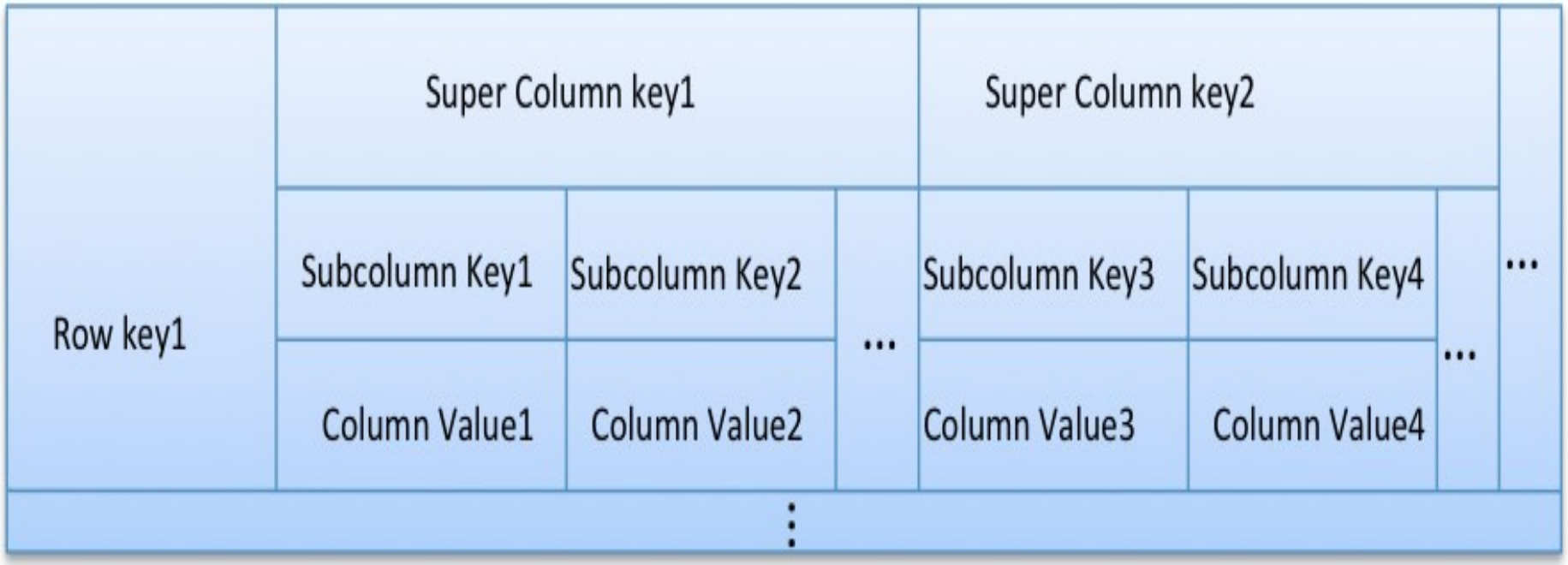
NoSQL Data Processing Tools

- **Key-value databases- Redis (CP)**
  - *The maximum of value limit to 1 GB.*
  - *suitable for providing high performance computing to small amount of data.*
  - *main drawback is that capacity of the database is limited by physical memory.*
  - *Support sql queries.*
  - *Simple values or data structures by keys*



Column-oriented database-Cassandra

- *Multi-datacenter replication*
- *Support for map/reduce, good for analytics, data warehousing*
- *Tunable consistency & strong availability and partition tolerance (AP)*
- *No single point of failure*
- *Probably the easiest of this list to manage in big/growing clusters*
- *Fact reading from database*



Document database- MongoDB

General Purpose	Rich data model	Support complex data types	Sophisticated query language reduceable to SQL
Easy to Use	Easy mapping to object oriented code	High-speed	Simple to setup and manage
Fast & Scalable	open source and no cost to use download	Auto-sharding built in	Dynamically add / remove capacity with no downtime

MongoDB is easy to use

MySQL

Select *from emp;

Create table log(<col1> size,<col2> size);

Insert into products values(“book”,40);

MongoDB

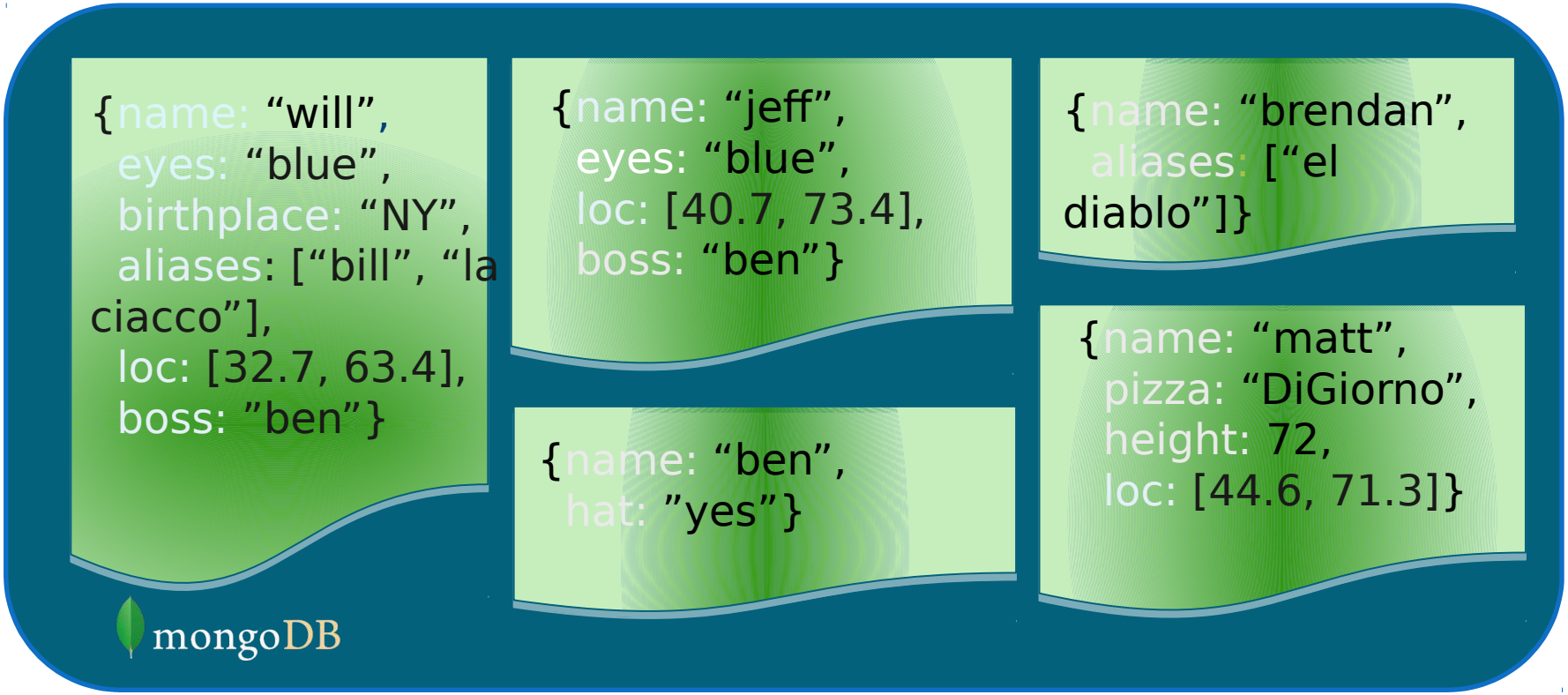
db.emp.find({});

db.createCollection("log", { capped : **true, size : 5242880, max : 5000 });**

db.products.save({ item: "book", qty: 40 });

Schema Free

- MongoDB does not need any pre-defined data schema *[5]*
- Every document could have different data!



NoSQL is popular for development & deployment of distributed system applications .

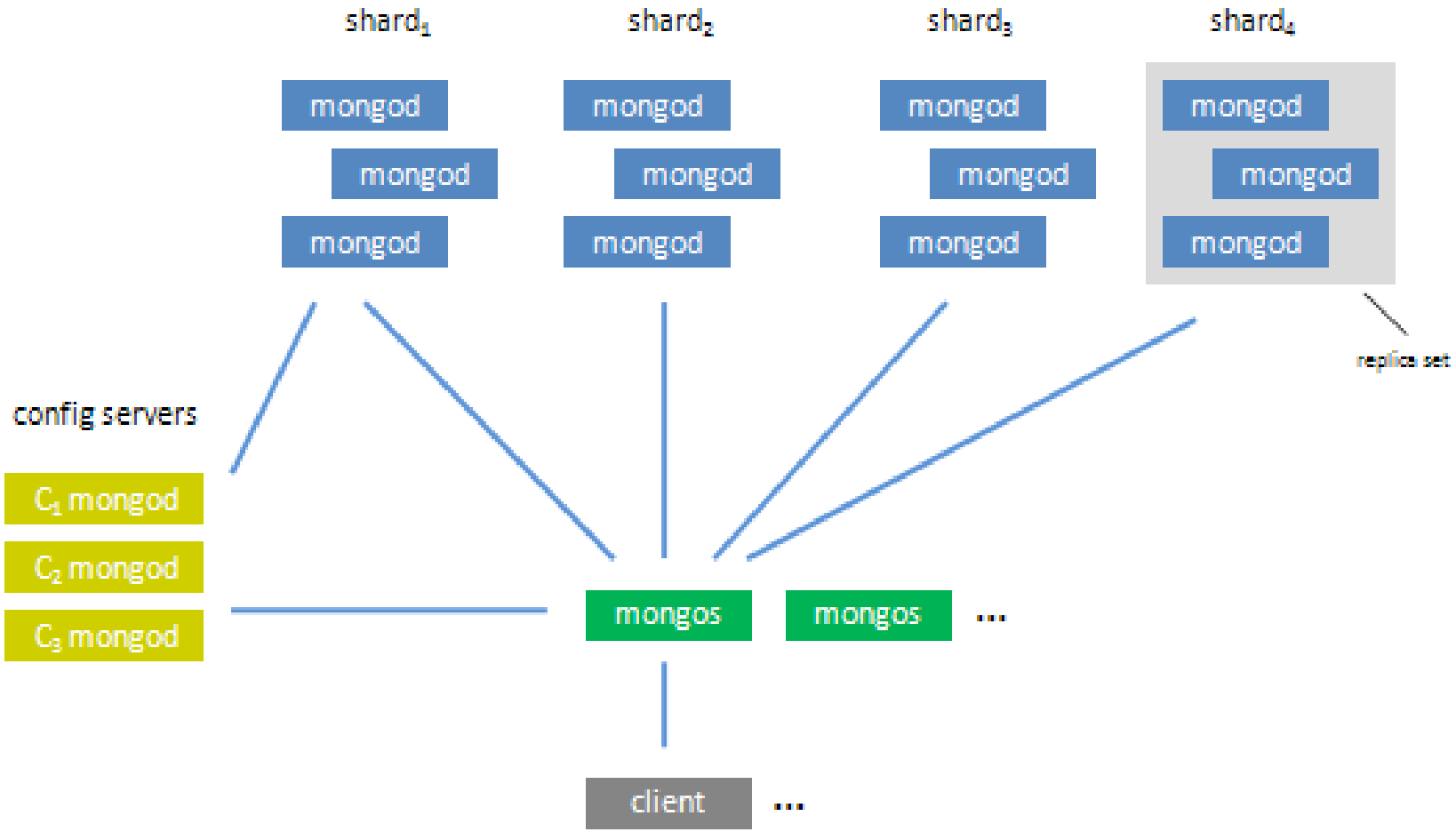
MongoDB makes it easy to code, scale, and operate NoSQL.

10gen is the company behind MongoDB

SQL Vs MongoDB

SQL	Mongodb
Database	Database
Table	Collection
Row	JSON document or BSON document
Column	Field
table joins	embedded documents and linking
primary key	Specify any unique column as primary key
Aggregation (e.g. group by)	aggregation framework

Sharding with mongodb



Practical examples of NoSQL

- Social networking sites
- Session Store
- User Profile Information
- Content and Metadata store
- Mobile Application
- Online shopping sites
- E-commerce
- Ad-targeting