Database Management System – 49 (NoSQL Databases)

Ajay James
Asst. Prof in CSE
Government Engineering College Thrissur

AJ-GEC THRISSUR

Outline

- NoSQL Databases
- Features of NoSQL
- Key-value Pair Based
- Column-oriented Graph
- Graphs based
- Document-oriented

NoSQL databases

- Not only SQL
- Non-tabular databases
- Store data differently than relational tables

AP, GEC THRISSUR

Relational vs NoSQL

Relational Database	NoSql Database
Supports powerful query language	Supports very simple query language
It has a fixed schema	No fixed schema
Follows ACID (Atomicity, Consistency, Isolation, and Durability).	It is only “eventually consistent”
Supports transactions	Does not support transaction

Features of NoSQL

- **Non-relational**
  - NoSQL databases never follow the relational model
  - Never provide tables with flat fixed-column records
  - Doesn't require data normalization
  - No complex features like query languages, query planners, referential integrity joins, ACID
- **Schema-free**
  - NoSQL databases are either schema-free or have relaxed schemas
  - Do not require any sort of definition of the schema of the data
  - Offers heterogeneous structures of data in the same domain

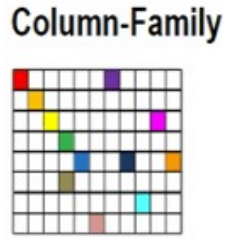
AJCEC THRISSUR

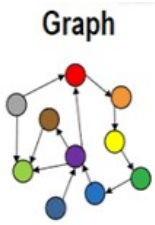
Features of NoSQL

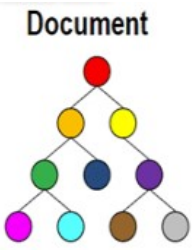
- **Simple API**
  - Offers easy to use interfaces for storage and querying data provided
- **Distributed**
  - Multiple NoSQL databases can be executed in a distributed fashion
  - Offers auto-scaling and fail-over capabilities
  - Often ACID concept can be sacrificed for scalability and throughput
  - Only providing eventual consistency

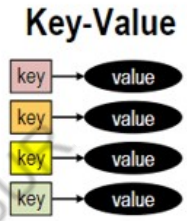
Types of NoSQL database

- Key-value Pair Based
- Column-oriented Graph
- Graphs based
- Document-oriented









AJ-GEC THRISSUR

Key-value Pair Based

- Uses the *associative array* (also called a *map* or *dictionary*) as their fundamental data model
- Data is represented as a collection of *key–value pairs*, such that each possible key appears at most once in the collection
- Examples - **Redis**, Dynamo, Riak

Column-based

- Similar at first appearance to traditional relational DBMS
- Concepts of rows and columns are still there
- Instead of storing data in a row for fast access, data is organized for fast column operations
- Ideal for running aggregate functions or for looking up records that match multiple columns
  - Counting the number of results, summing them, or calculating their mean average
- **Cassandra**, MariaDB, CrateDB, ClickHouse, Greenplum Database, Apache Hbase, Apache Kudu, Apache Parquet

AMGEC THRISSUR

Row oriented vs column oriented

Row-oriented

ID	Name	Grade	GPA
001	John	Senior	4.00
002	Karen	Freshman	3.67
003	Bill	Junior	3.33

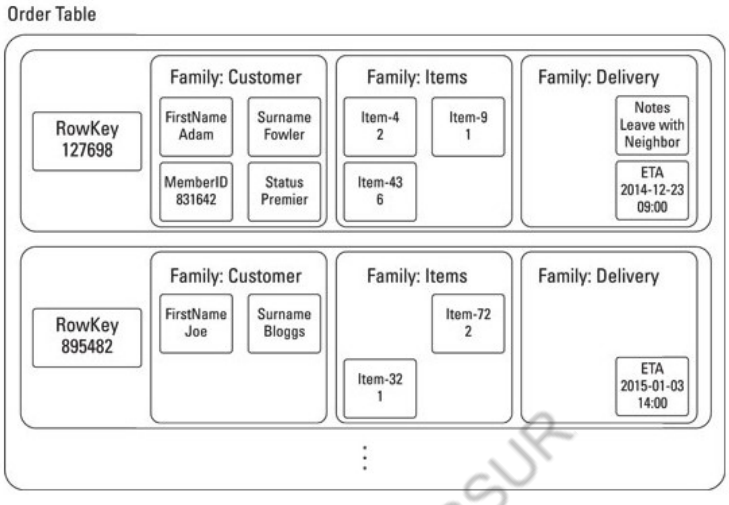
Column-oriented

Name	ID
John	001
Karen	002
Bill	003

Grade	ID
Senior	001
Freshman	002
Junior	003

GPA	ID
4.00	001
3.67	002
3.33	003

Column-based

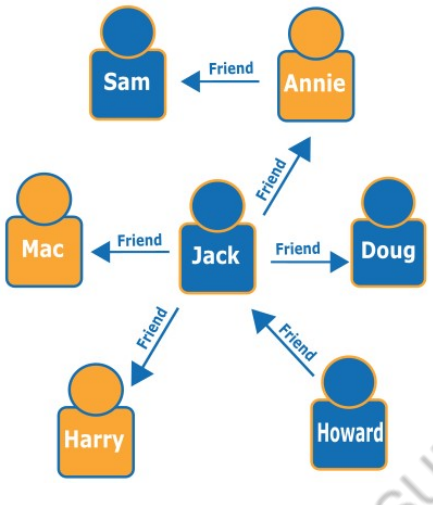


AMGEC THRISSUR

Graphs based

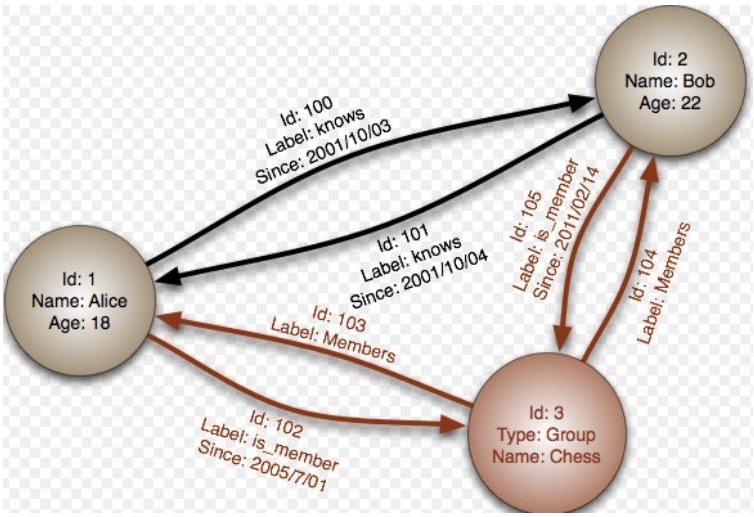
- To store and navigate relationships
- Relationships are first-class citizens in graph databases
- Most of the value of graph databases is derived from these relationships
- Graph databases use nodes to store data entities, and edges to store relationships between entities
- No limit to the number and kind of relationships a node can have
- Applications - Social networking, recommendation engines, and fraud detection
- **ArangoDB**, Neo4j, Amazon Neptune, Dgraph, JanusGraph

Graphs based



A-GEC THRISSUR

Graph based



Document-oriented

- Retrieve, store, and manage document oriented information
- *Semistructured* data
- Written as a JavaScript Object Notation (JSON) object
- JSON is a human-readable data format
- **MongoDB**, Cosmos DB, DocumentDB, SimpleDB, PostgreSQL, OrientDB, Elasticsearch, RavenDB

AJ-GEC THRISSUR

Document oriented

Tom's contact card document

```
{
    "_id": "tomjohnson",
    "firstName": "Tom",
    "middleName": "William",
    "lastName": "Johnson",
    "email": "tom.johnson@digitalocean.com",
    "department": ["Finance", "Accounting"]
}
```

Tom's contact card document with social media accounts information attached

```
{
    "_id": "tomjohnson",
    "firstName": "Tom",
    "middleName": "William",
    "lastName": "Johnson",
    "email": "tom.johnson@digitalocean.com",
    "department": ["Finance", "Accounting"],
    "socialMediaAccounts": [
        {
            "type": "facebook",
            "username": "tom_william_johnson_23"
        },
        {
            "type": "twitter",
            "username": "@tomwilliamjohnson23"
        }
    ]
}
```

References

- https://en.wikipedia.org/wiki/Graph_database
- https://www.digitalocean.com/community/conceptual_articles/an-introduction-to-document-oriented-databases
- https://en.wikipedia.org/wiki/Document-oriented_database
- https://www.geeksforgeeks.org/introduction-to-nosql/

AJ-GEC THRISSUR

Thank you