ADBMS

Chapter 04 NOSQL DISTRIBUTION MODEL

Difference NOSQL AND SQL

Fast Parameter	SQL Database	NoSQL Database
Type of Database	Relational Databases	Non-relational or distributed databases
Query Language	Structured Query Language (SQL)	No declarative query language
Schema	Schema of database is fixed.	Schema of database is not fixed and is dynamic in nature.
Ability to scale	Vertically Scalable	Horizontally Scalable
Model	Uses ACID model	Uses BASE model
Best suited for	Ideal choice for complex query intensive environment.	Suitable for the hierarchical data store as it supports key-value pairs.
Importance	It should be used when data validity is super important.	It should be used when fast data is more important than correct data.
Best option	When you need dynamic query support.	When you need scaling abilities for future requirements.
Examples	Oracle, Postgres, and MS-SQL.	MongoDB, Redis, Neo4j, Cassandra, Hbase.

SQL vs. NoSQL: Comparison

ml4devs.com/datastores

SQL		NoSQL
Relational	Model	Non-relational
Structured tables	Data	Semi-structured
Strict schema	Flexibility	Dynamic schema
ACID	Transactions	Mostly BASE, few ACID
Strong	Consistency	Eventual to Strong
Consistency prioritized	Availability	Basic Availability
Vertically by upgrading hardware	Scale	Horizontally by data partitioning

Key-Value
Dictionary or Hash Table

Wide Column
2-D Versioned Key-Value

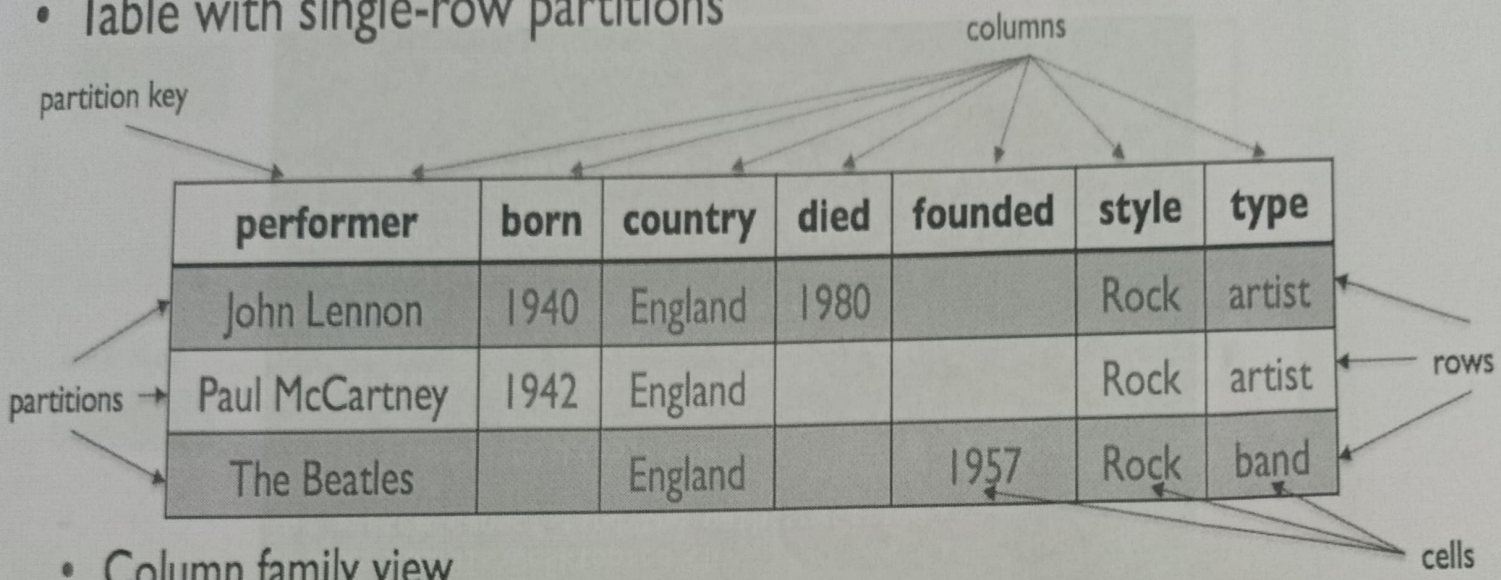
Document
Nested Objects (XML, JSON, YAML)

Graph
Entity-Relationships

© Satish Chandra Gupta
CC BY-NC-ND 4.0 International License
creativecommons.org/licenses/by-nc-nd/4.0/

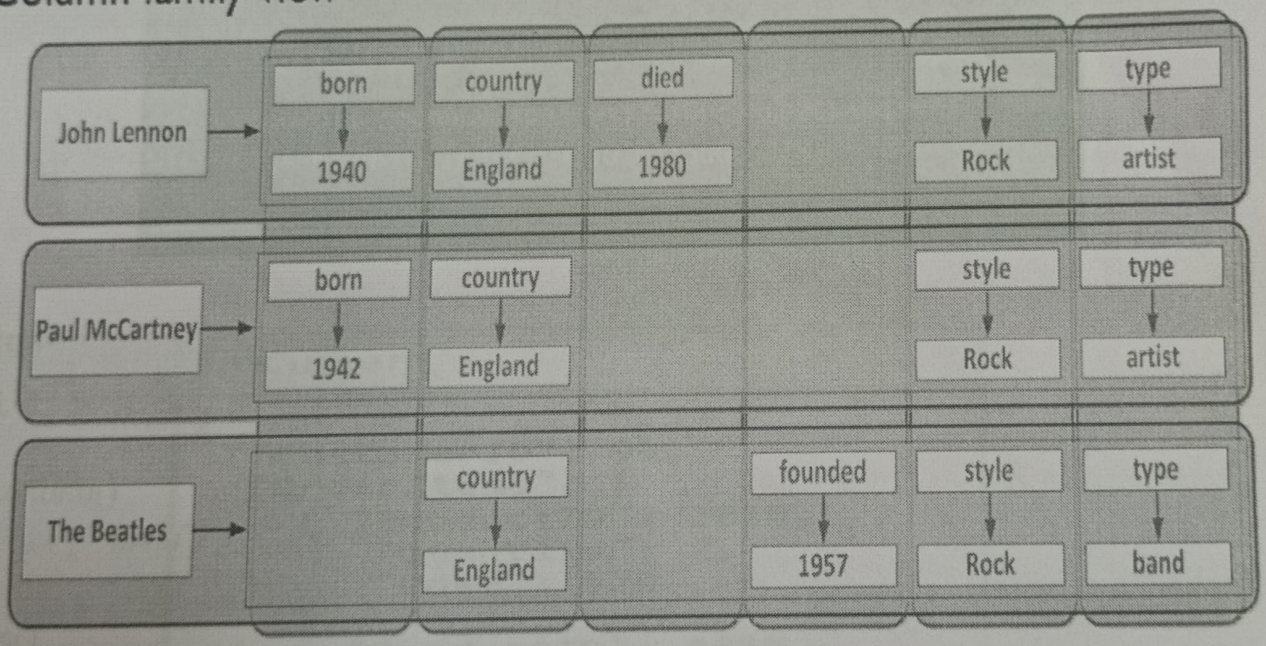
scgupta.me
twitter.com/scgupta
linkedin.com/in/scgupta

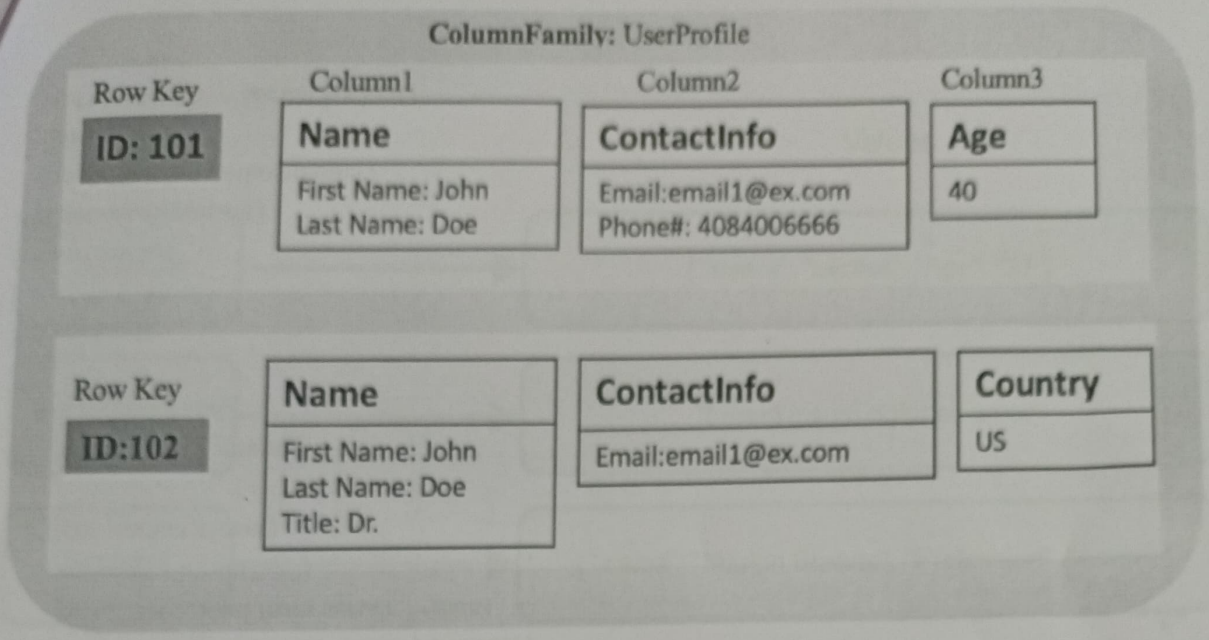
- Table with single-row partitions



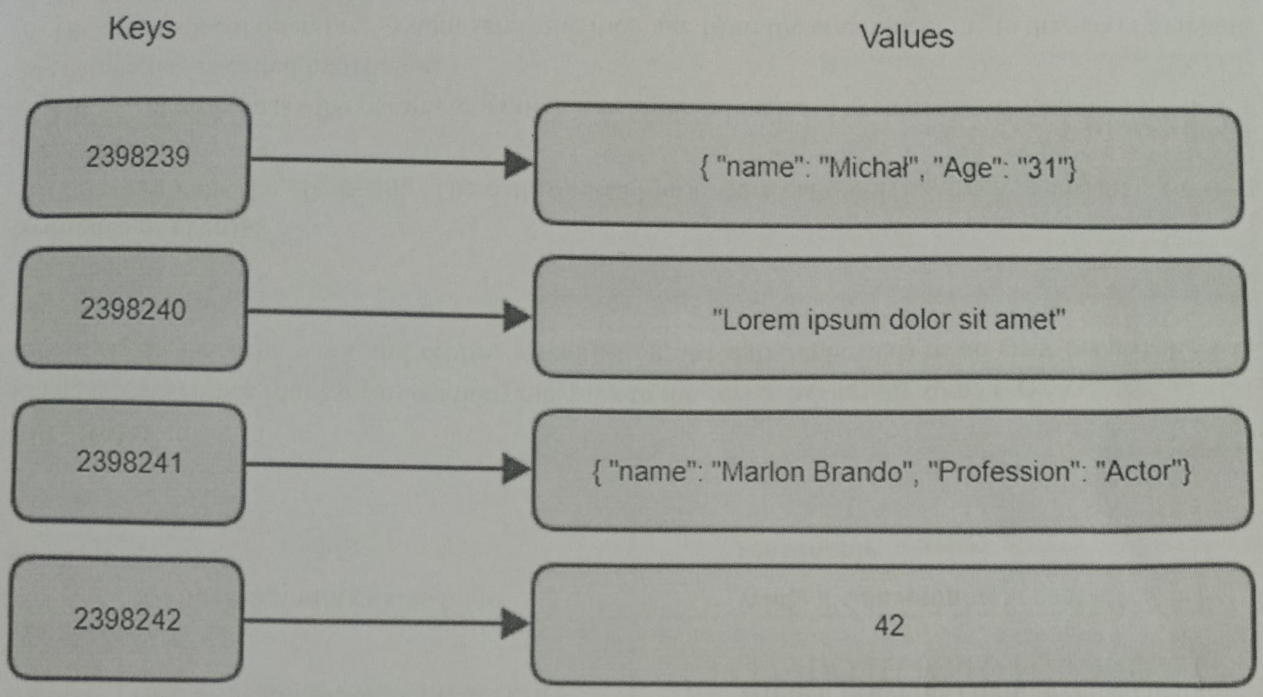
performer	born	country	died	founded	style	type
John Lennon	1940	England	1980		Rock	artist
Paul McCartney	1942	England			Rock	artist
The Beatles		England		1957	Rock	band

- Column family view





Key	Document
1001	{ "CustomerID": 99, "OrderItems": [{ "ProductID": 2010, "Quantity": 2, "Cost": 520 }, { "ProductID": 4365, "Quantity": 1, "Cost": 18 }], "OrderDate": "04/01/2017" }
1002	{ "CustomerID": 220, "OrderItems": [{ "ProductID": 1285, "Quantity": 1, "Cost": 120 }], "OrderDate": "05/08/2017" }



KEY VALUE PAIR EXAMPLE

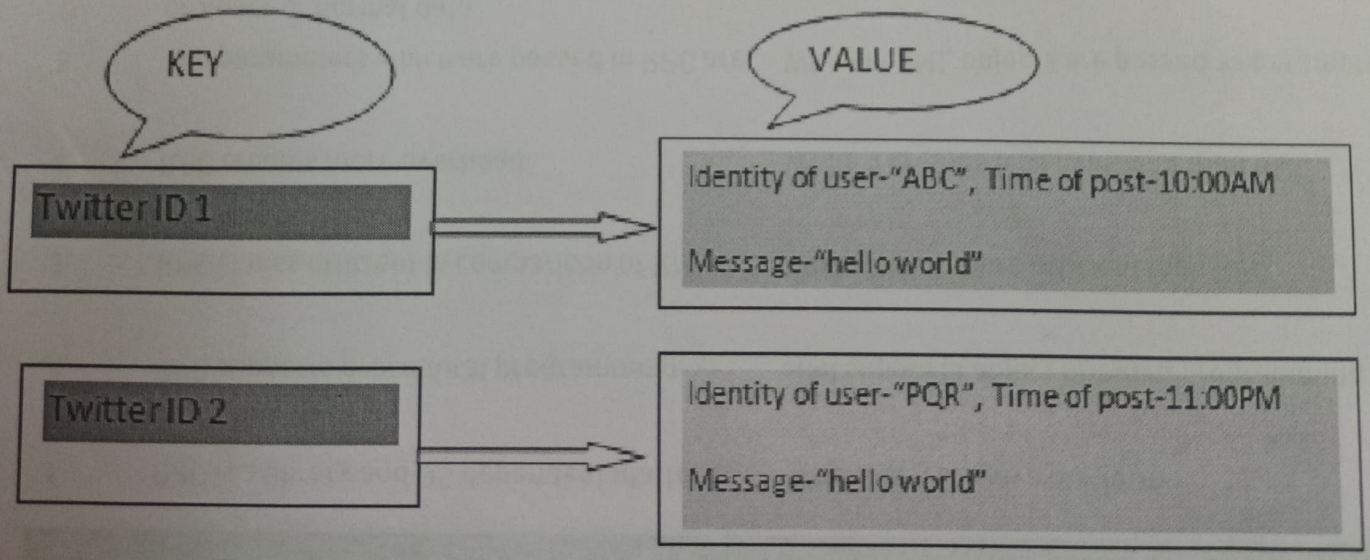
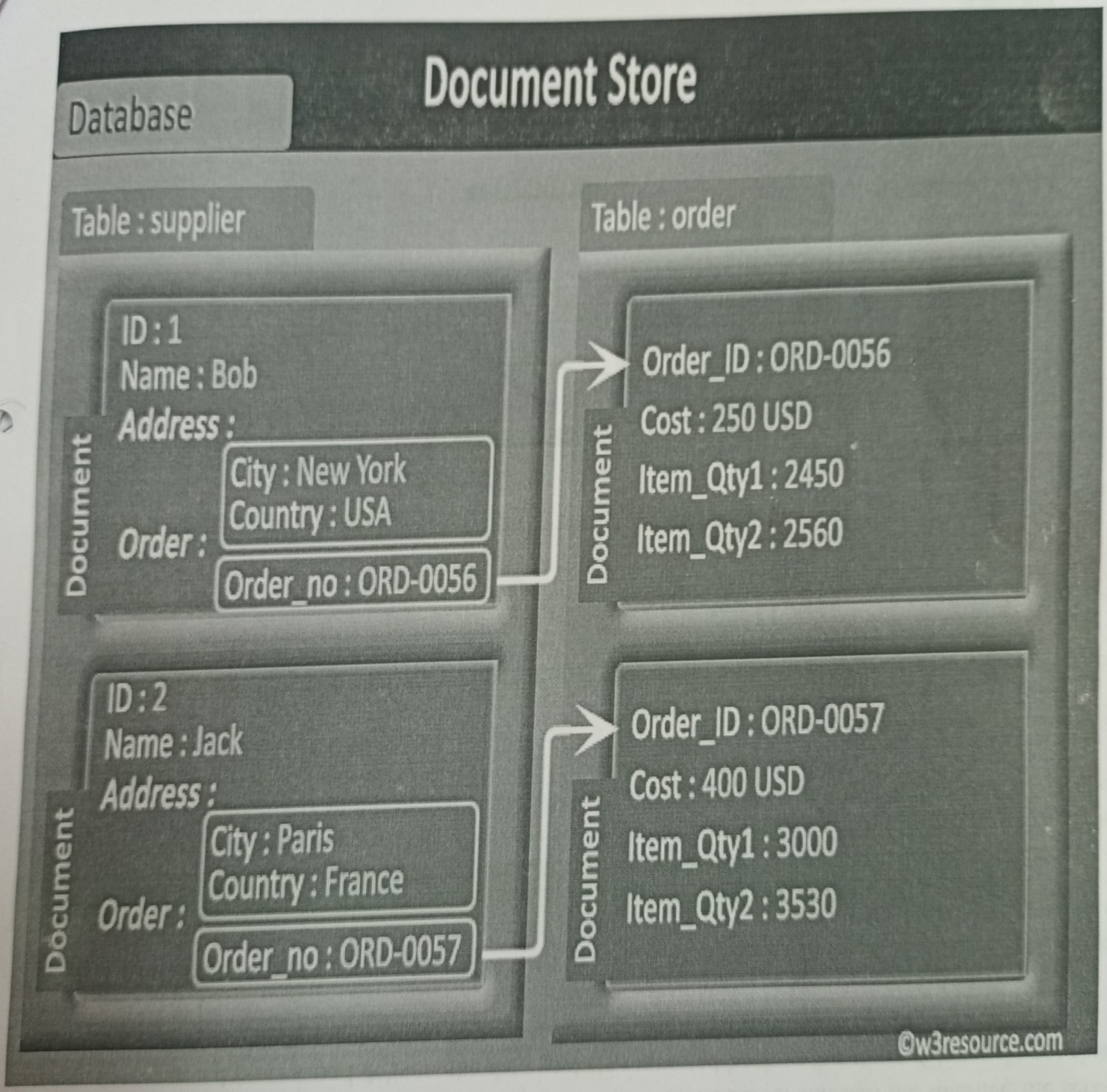


Figure2- Example of Key-value NoSQL database

Key	Value
Name	{ "FIrstName": "Bruce", "LastName": "Wayne" }
Email	bruce@wayneenterprises.com
Type	Gold
Image	1011110010101010101000101 1001010101010001011 00...

EXAMPLE OF DOCUMENT DATABASE



Documents: Structure References

contact document

```
{
  _id: <ObjectId2>,
  user_id: <ObjectId1>,
  phone: "123-456-7890",
  email: "xyz@example.com"
}
```

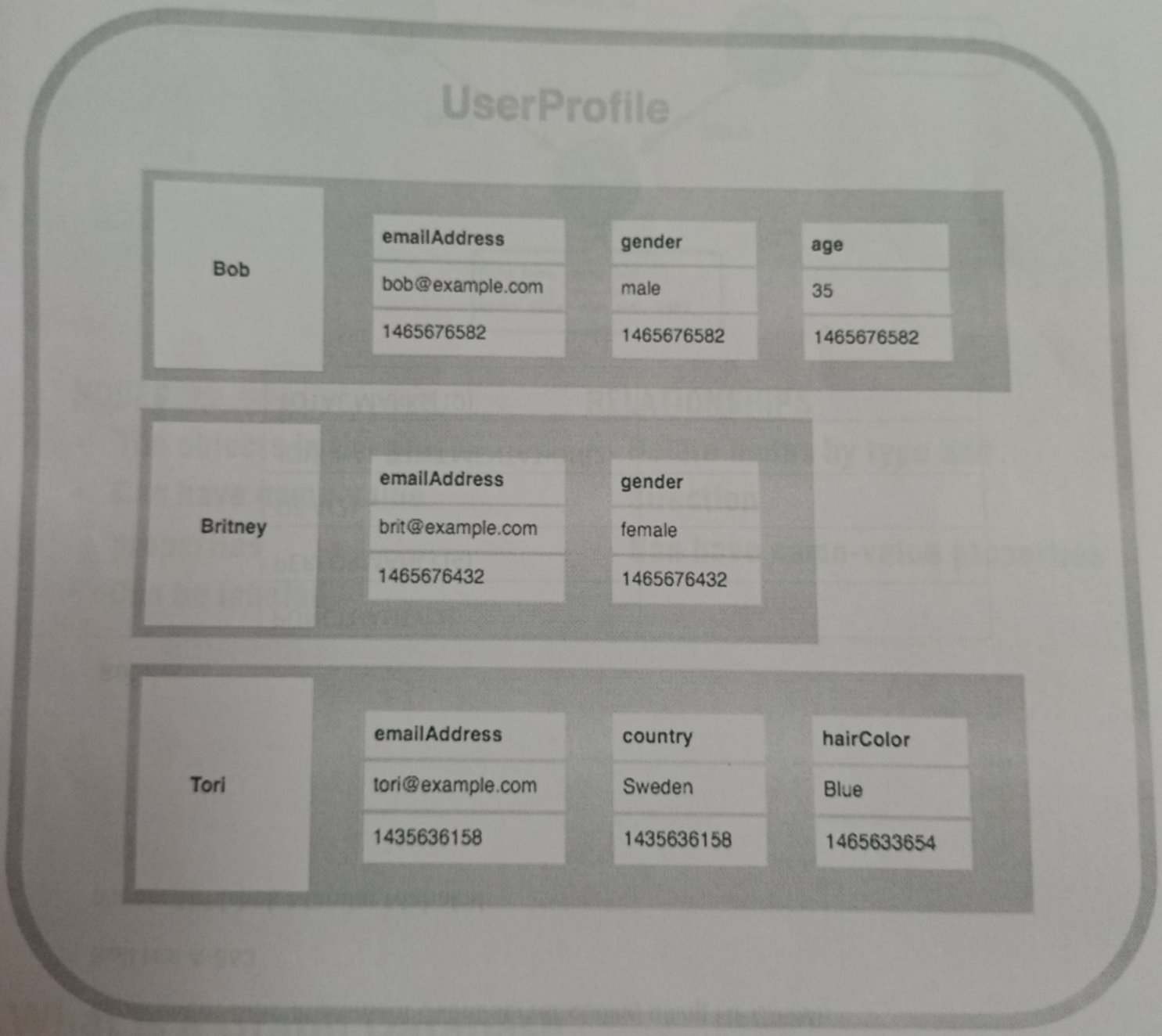
user document

```
{
  _id: <ObjectId1>,
  username: "123xyz"
}
```

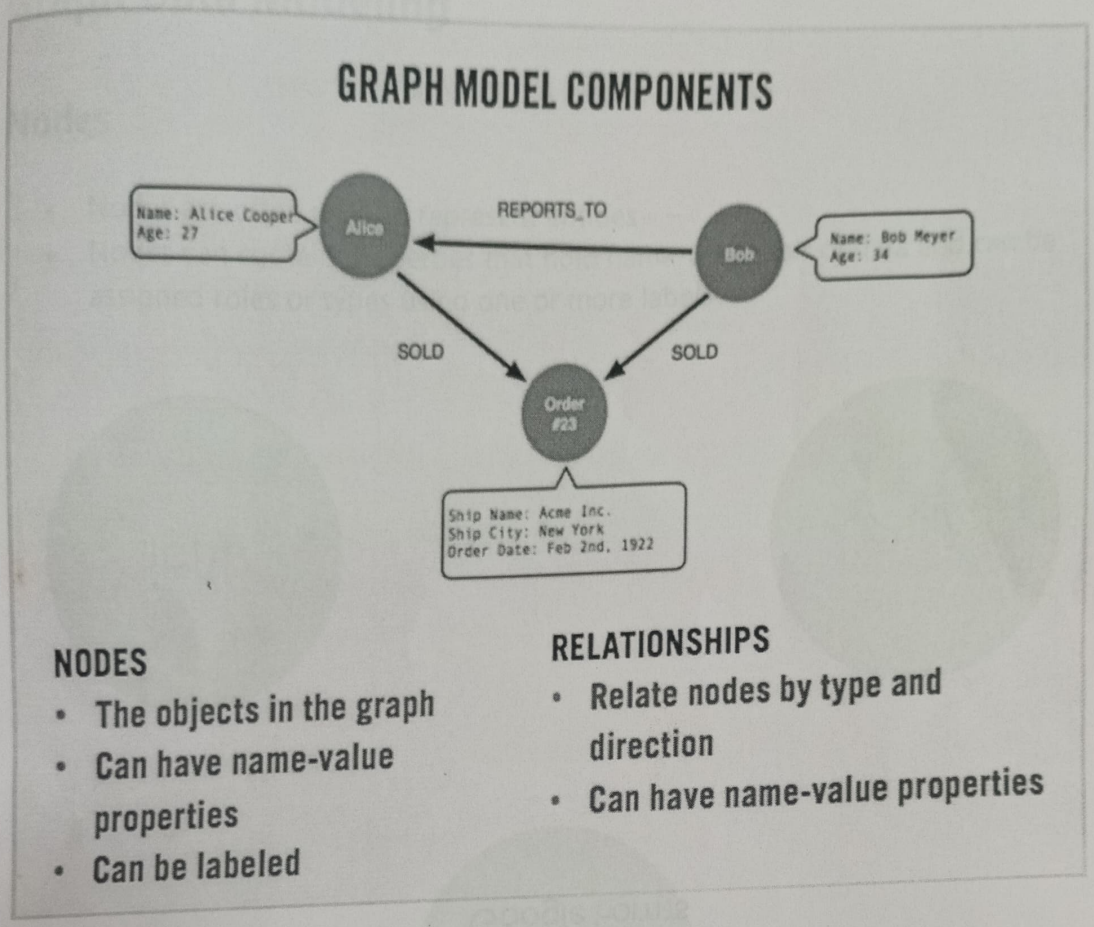
access document

```
{
  _id: <ObjectId1>,
  user_id: <ObjectId1>,
  level: 5,
  group: "dev"
}
```

COLUMN ORIENTED DATABASE



EXAMPLE OF GRAPH DATABASE



What is a Graph Database

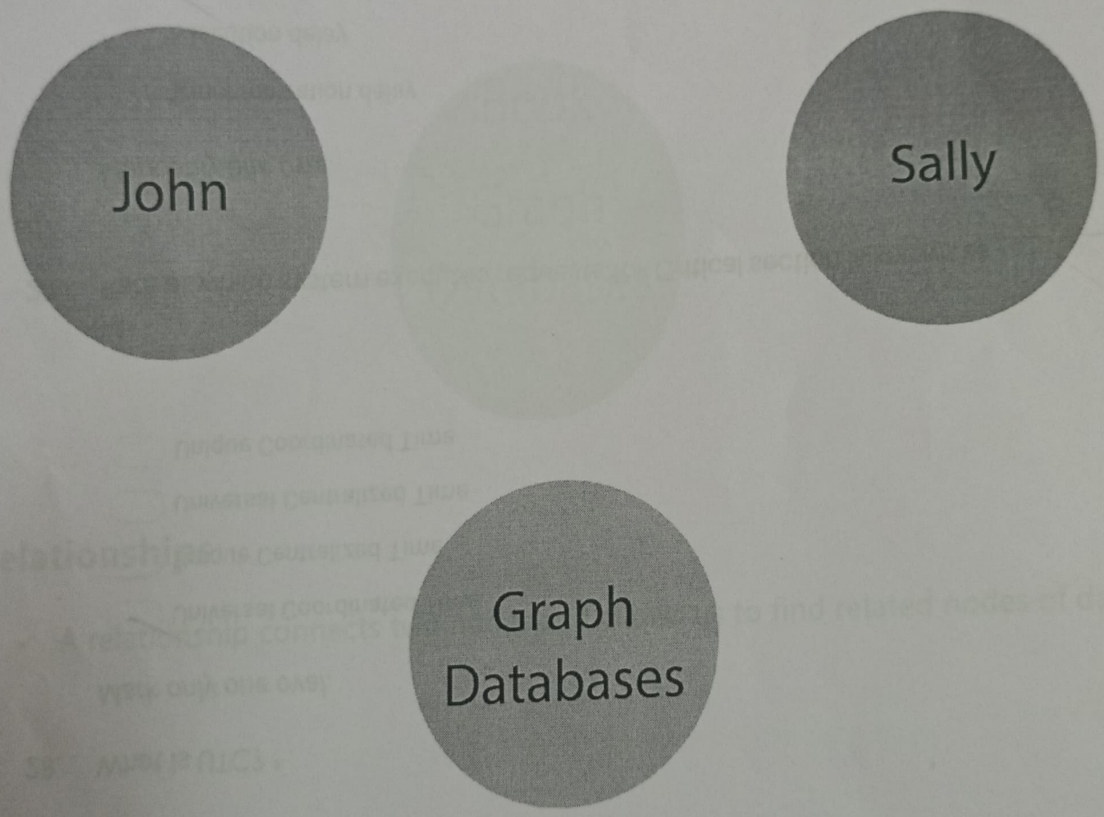
Comment

Graph database is a database that uses graph strutures to represent and store data
It uses nodes to store entities, edges to store relationship

Graph Data Modeling

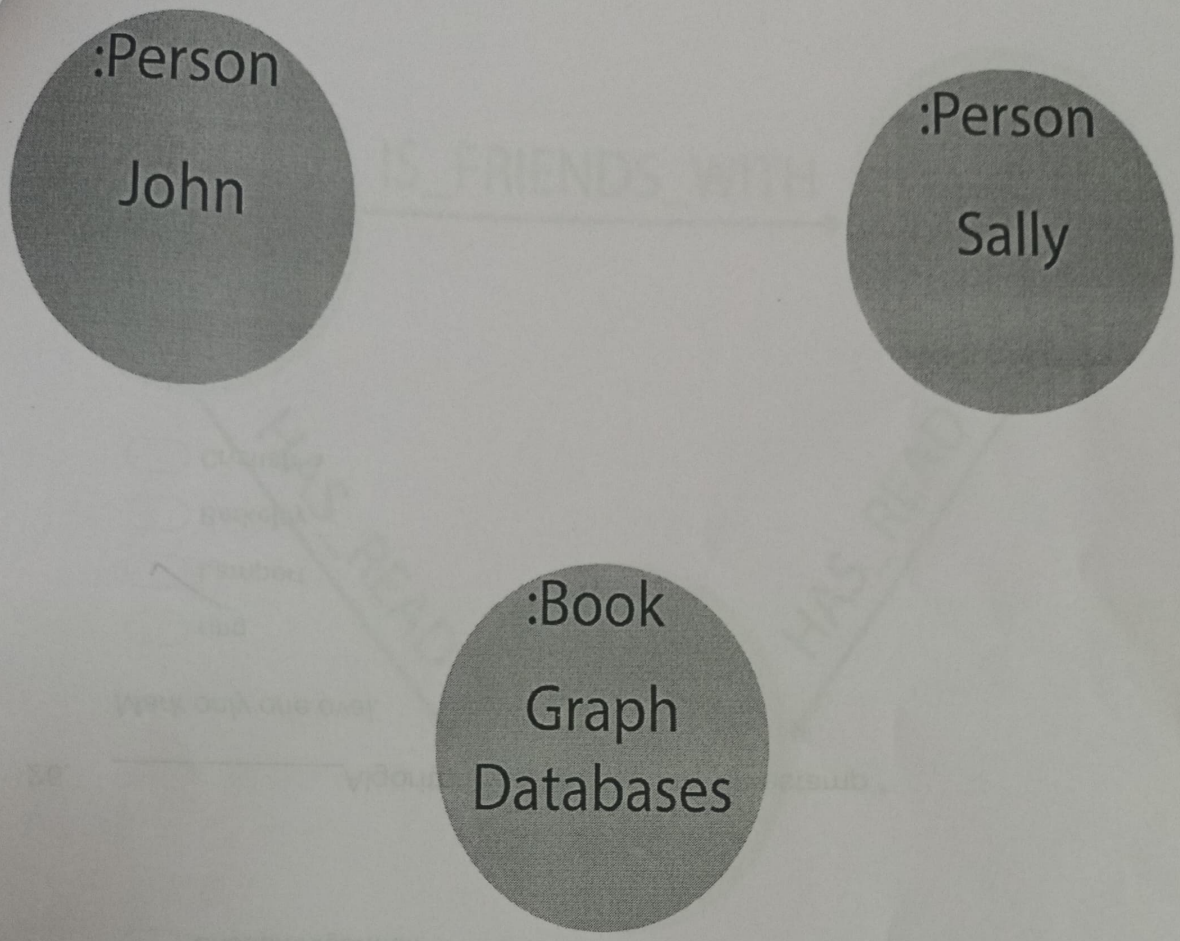
Nodes

- Nodes are often used to represent entities
- Nodes can contain properties that hold name-value pair of data and can be assigned roles or types using one or more labels



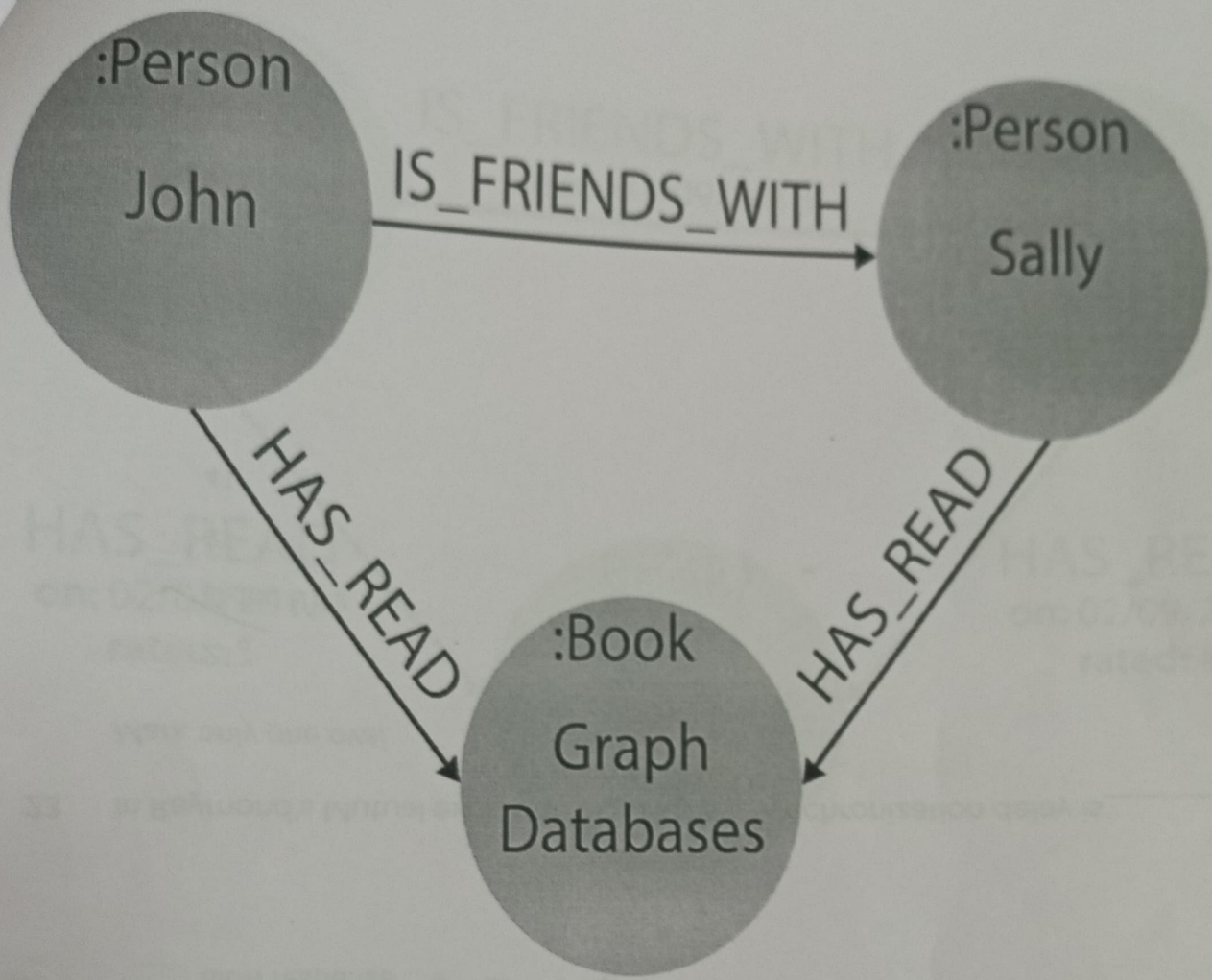
Labels

- Label is a named graph construct that is used to group nodes into sets
- A node can be labeled with any number of labels
- Labels make queries easier to write and more efficient



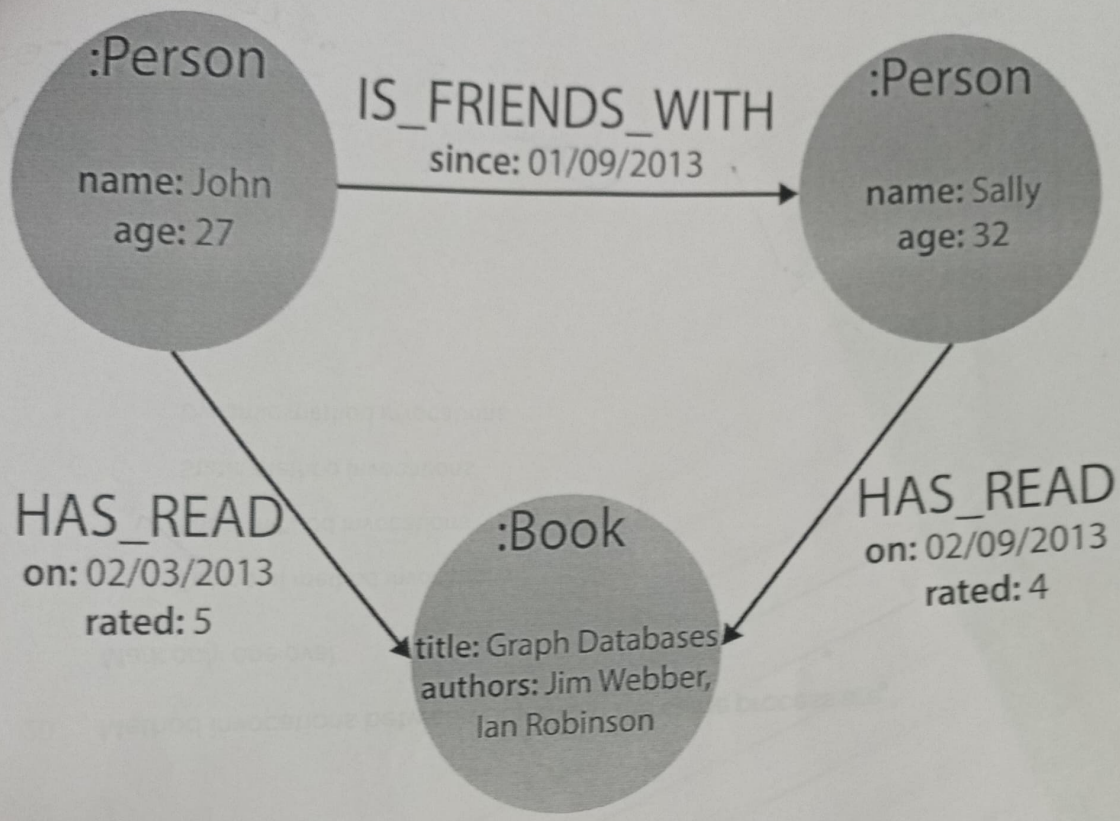
Relationships

- A relationship connects two nodes and allows us to find related nodes of data



Properties

- Properties are name-value pairs of data that you can store on nodes or relationships



When to use Graph Database

Is Data Highly-Connected?

If the connections within the data are not the primary focus or the graph is sparsely connected, then graph database is not the best fit.

Is Retrieving the Data more Important than Storing it?

Graph databases are optimized for data retrieval.
A good rule of thumb is, if you don't intend to use JOIN operations in your queries, then a graph is not a must-have.