GraphDB

Bootcamp II - 28 Mei 2021

NoSQL Database

- Bisa diartikan sebagai Not Only SQL.
- DBMS yang memiliki kecenderungan tanpa relasi dan mempunyai skalabilitas tinggi untuk dapat berkembang dan mengolah *Big Data* yang selalu berubah-ubah sekalipun.
- Istilah NoSQL pertama kali dikenalkan oleh Carl Strozzi tahun 1998 yang kemudian didiskusikan kembali sebagai *open source distributed database*

Perbandingan RDBMS dengan NoSQL

RDBMS	NoSQL
High-value, high-density, complex data	Low-value, low-density, simple data
Complex data relationships	Very simple relationships
Joins	Avoids joins
Schema-centric, structured data	Unstructured or semi-structured data
Designed to scale up	Distributed storage and processing
Well-defined standards	Standards not yet evolved
Database-centric	Application- and developer-centric
High security	Minimal or no security

Perbandingan HDFS dengan NoSQL

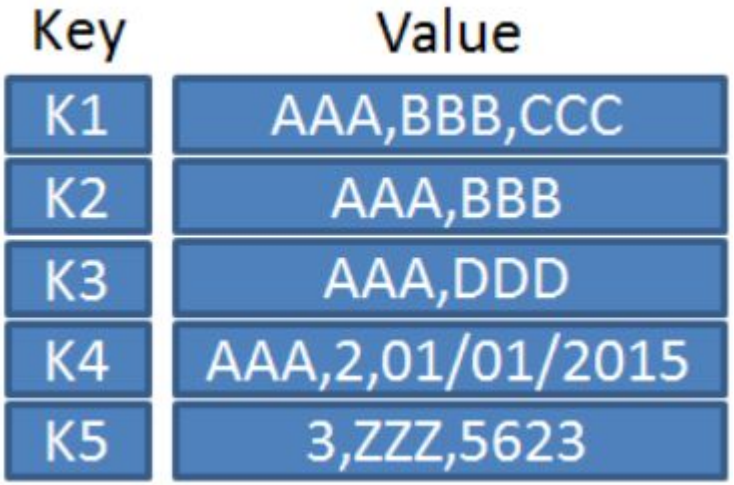
HDFS	NoSQL
File system	Database
No inherent structure	Simple data structure
Bulk storage	Fast access to specific records
Write once, read many	Read, write, delete, update

Jenis-Jenis Database NoSQL

- Key Value (Redis, DynamoDB, Riak)
- Column Based / Wide Column (Hbase, Cassandra, HyperTable)
- Document Oriented (Amazon SimpleDB, CouchDB, MongoDB)
- Graph Based (Neo4J, JanusGraph, InfiniteGraph)

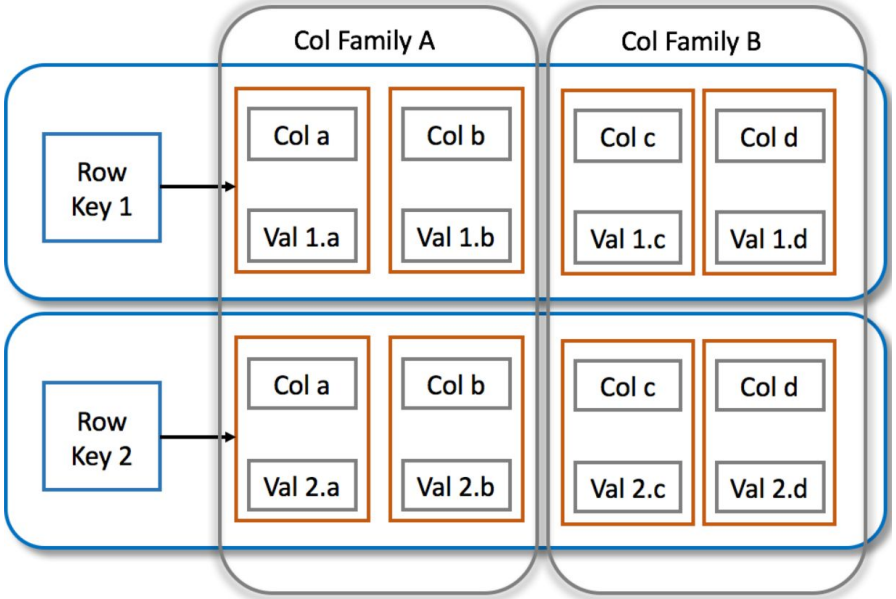
Key Value

Diperkenalkan sebagai *storage data* yang di desain untuk menyimpan, membaca, dan mengelola *associative array*.



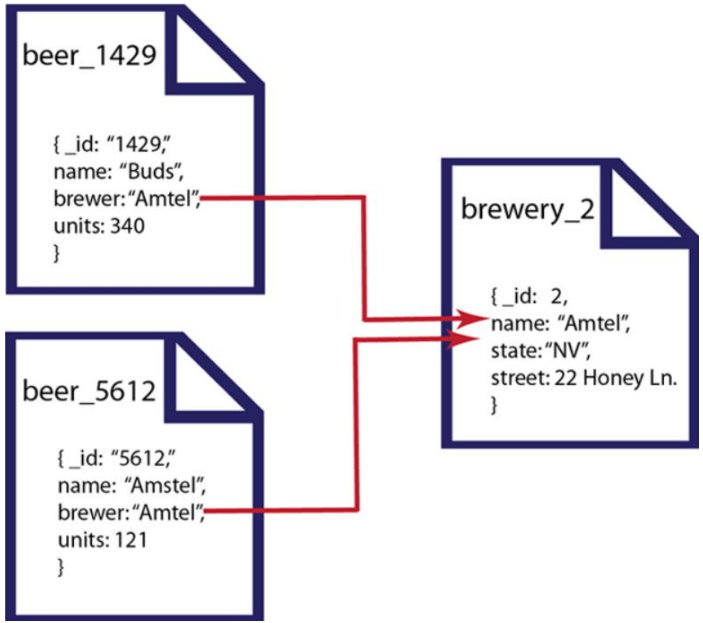
Column Based / Wide Column

Mengikuti ide Google’s Big Table, dimana setiap data di distribusi sesuai kelompoknya.



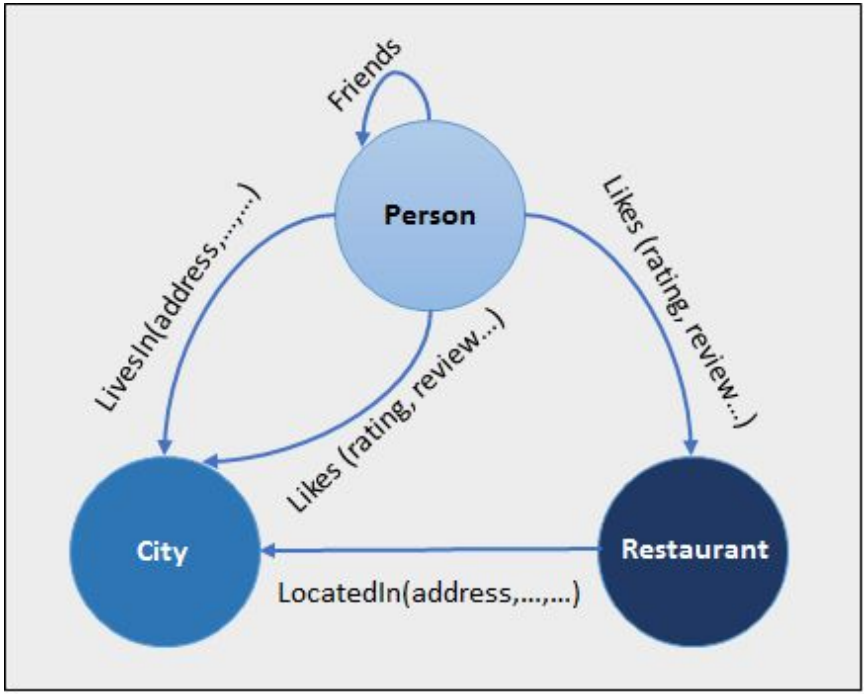
Document Oriented

Secara sederhana, jenis ini dapat diartikan sebagai mekanisme penyimpanan data yang formatnya berupa dokumen seperti XML atau JSON.

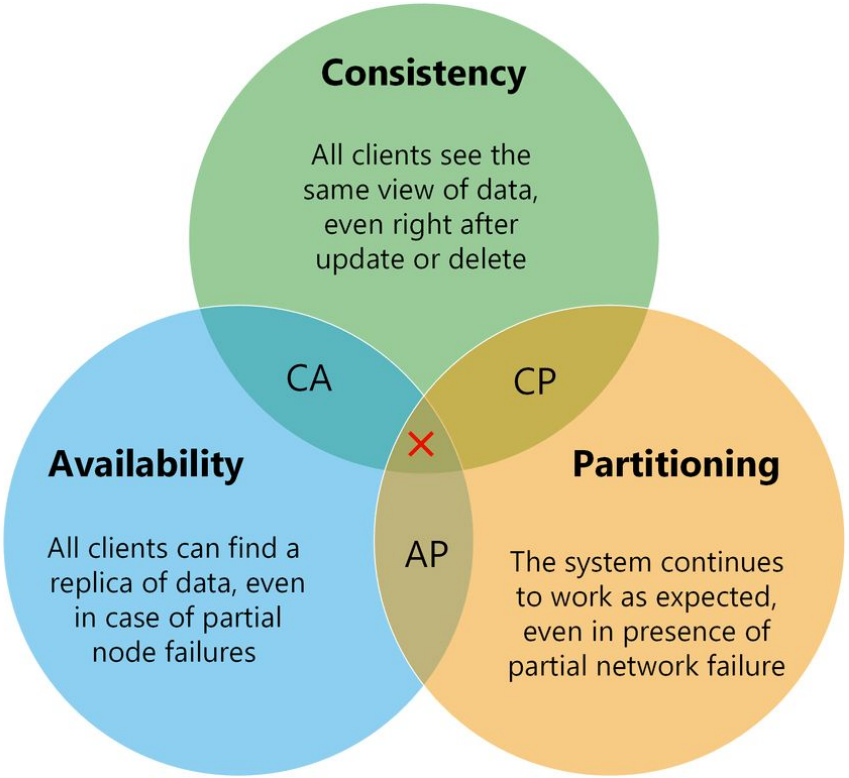


Graph Based

Fokus pada penyimpanan hubungan antar entitas.

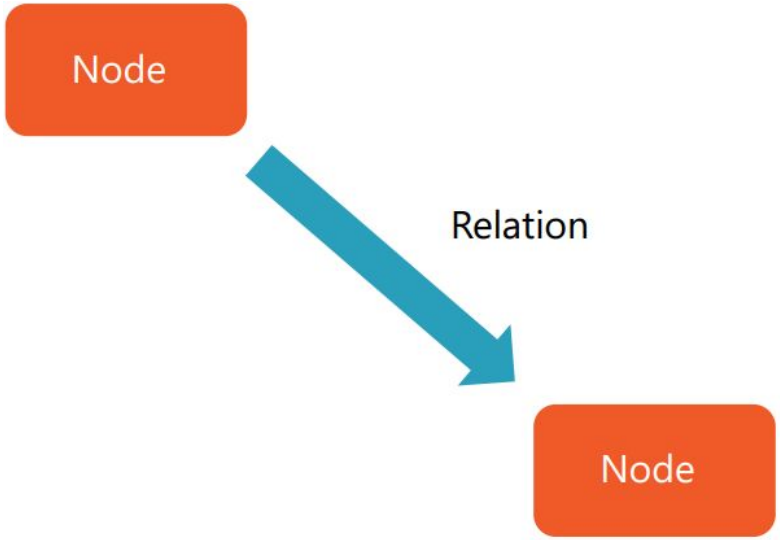


CAP Theorem

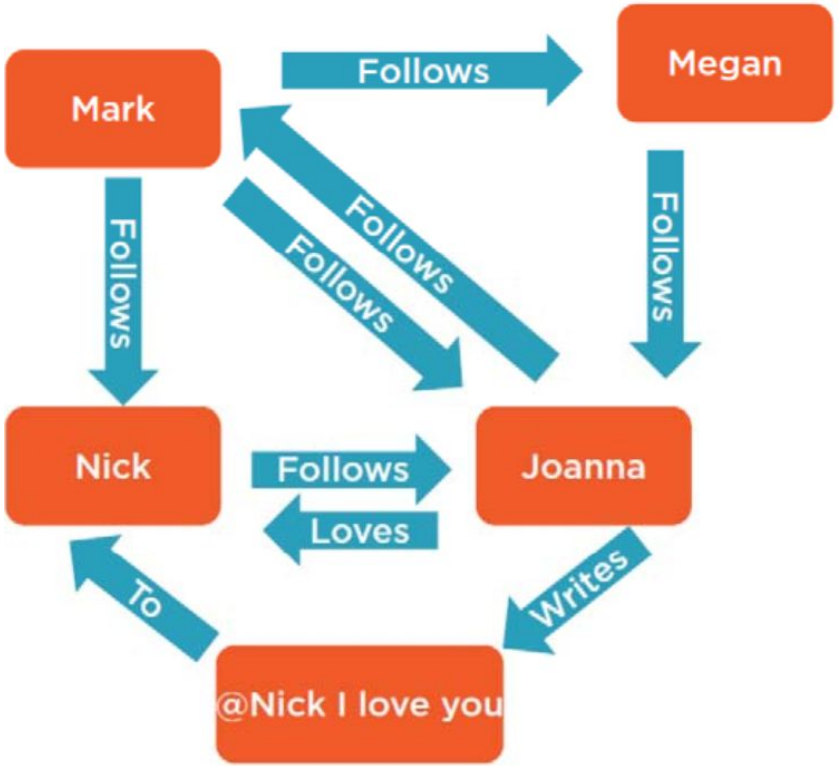

Consistency
All clients see the same view of data, even right after update or delete
CA
CP
Availability
All clients can find a replica of data, even in case of partial node failures
AP
Partitioning
The system continues to work as expected, even in presence of partial network failure

Graph

Pada matematika, graph merupakan sebuah struktur yang dapat memodelkan hubungan antar object.



Graph


Mark
Follows
Megan
Follows
Follows
Follows
Follows
Nick
Follows
Joanna
Loves
To
Writes
@Nick I love you

Graph

- Mudah digunakan untuk menampilkan informasi yang sifatnya *extendable* dan *expandable*
- Tampilan informasi sangat memudahkan bagi *user*
- *Whiteboard Compatible*

Property Graph Model

Contains nodes and relationships

Nodes and relationships contain properties

Relationships are named and directed with a start and end node

Mengapa Harus Graph?

- *Highly related data*
- *Flexible schema*
- Struktur data yang digunakan mudah diterima user

"Use a relational database for all applications"

"Consider the type of database for every application you're writing"

Perbandingan Relational DB dengan Graph DB

Relational	Graph
Tables	Nodes
Schema with nullables	No schema
Relations with foreign keys	Relation is first class citizen
Related data fetched with joins	Related data fetched with a pattern

Partner and Vukotic's Experiment

- Social Network
- Friends of Friends Structure
- MySQL and Neo4J
- 1,000,000 people
- Each with an average of 50 friends
- Depth 2: Find all friends of a user's friends
- Depth 3: Find all friends of friends of a user's friends
- Etcetera

Hasil Experiment

Depth	Rel.Db(s)	Neo4j(s)	#records
2	0,016	0,01	~2500
3	30,267	0,168	~110000
4	1543,505	1,359	~600000
5	Unfinished	2,132	~8000000

GraphQL