
The
Center of
Applied
Data Science
NoSQL
Not Only SQL

The Center of Applied Data Science

Introduction

CAP Theorem

ACID Principle for SQL need to be relaxed for distributed systems.

https://medium.baqend.com/nosql-databases-a-survey-and-decision-guidance-ea7823a822d

The Center of Applied Data Science

Different data models

The way data is stored and accessed

- Key-value store
- Document store
- Wide-column
- Graph database

The Center of Applied Data Science

KEY-VALUE STORE

The Center of Applied Data Science

Key-value Store

- A set of key-value pairs with unique keys
- It only supports get and put operations
- Key-value stores are therefore often referred to as **schemaless**: Any assumptions about the structure of stored data are implicitly encoded in the application logic (*schema-on-read*) and not explicitly defined through a data definition language (*schema-on-write*).

The Center of Applied Data Science

Key-value store

- Very simple data model
- Very simple abstraction

$\Rightarrow$ easy to partition and query the data

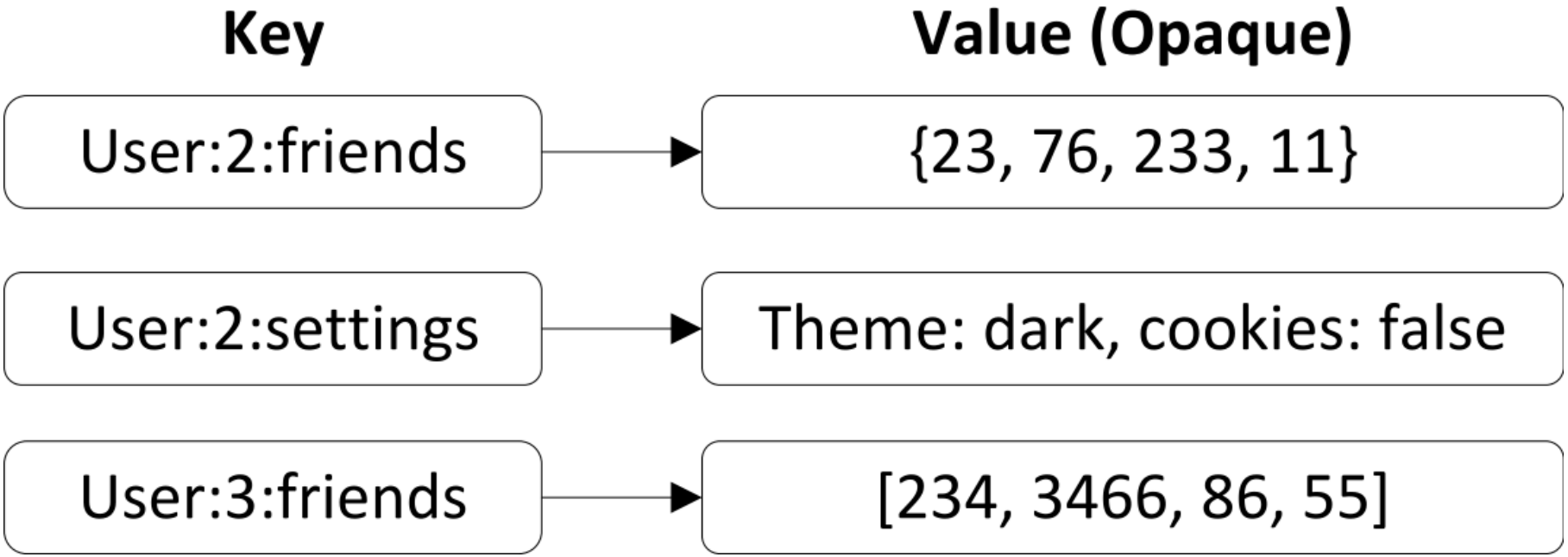
$\Rightarrow$ the database system can achieve **low latency** as well as **high throughput**.

But not possible to make complex queries (range queries…)

The Center of Applied Data Science

Key-value store

- It is possible to encode information in the keys and/or in the values



The Center of Applied Data Science

Technologies

- Redis (http://redis.io) in-memory database (**Re**mote **di**ctionary **s**erver)
- Berkeley DB
- Kyoto cabinet (Tokyo cabinet)
- Riak
- Couchbase

Key-value databases usage

Generally useful for

- storing session information
- user profiles
- Preferences
- shopping cart data

Avoid using Key-value databases when we need

- to query by data
- to have relationships between the data being stored
- to operate on multiple keys at the same time

The Center of Applied Data Science

Or document-oriented database

DOCUMENT STORE

The Center of Applied Data Science

Document Store

- a key-value store that restricts values to semi-structured formats such as JSON documents.
- This restriction in comparison to key-value stores brings great flexibility in accessing the data.
- Can get a **document** as well as a **specific field** of a document
- Like a Python dictionary

The Center of Applied Data Science

Document Store

It is possible

- to fetch an entire document by its ID,
- but also to retrieve only parts of a document:
  - e.g. the age of a customer,
  - and to execute queries like aggregation, query-by-example or even full-text search.

The Center of Applied Data Science

Document Store

- Each record and its associated data is thought of as a “document”
- In a document database, everything related to a database object is encapsulated together

https://www.mongodb.com/document-databases

Documents are independent units

- Better performance : related data is read contiguously off disk
- Easier to distribute data across multiple servers while preserving its locality

Application logic is easier to write

- Unnecessary to translate between objects in your application and SQL queries
- The object model can be translated directly into a document

Unstructured data can be stored easily

- A document contains whatever keys and values the application logic requires
- Costly migrations are avoided since the database does not need to know its information schema in advance.

https://www.mongodb.com/document-databases

The Center of Applied Data Science

Some technologies

- MongoDB
- CouchDB
- Cloud Datastore (https://cloud.google.com/datastore/)
- Amazon DynamoDB (https://aws.amazon.com/dynamodb/)
- Azure DocumentDB

Document databases usage

- Content management systems
- Blogging platforms
- Web analytics
- Real-time analytics
- Ecommerce-applications

Avoid using document databases for systems that need complex transactions spanning multiple operations or queries against varying aggregate structures

The Center of Applied Data Science

Column family store

WIDE-COLUMN DATA STORE

Row Oriented
(RDBMS Model)

id	Name	Age	Interests
1	Ricky		Soccer, Movies, Baseball
2	Ankur	20	
3	Sam	25	Music

Multi-valued

null

Column Oriented
(Multi-value sorted map)

id	Name
1	Ricky
2	Ankur
3	Sam

id	Age
2	20
3	25

id	Interests
1	Soccer
1	Movies
1	Baseball
3	Music

The Center of Applied Data Science

Column family store

- Each column family can be compared to a container of rows in an RDBMS table where
  - the key identifies the row
  - and the row consists of multiple columns.
- The difference is that various rows **do not have to have the same columns**, and **columns can be added** to any row at any time **without having to add it to other rows**.

The Center of Applied Data Science

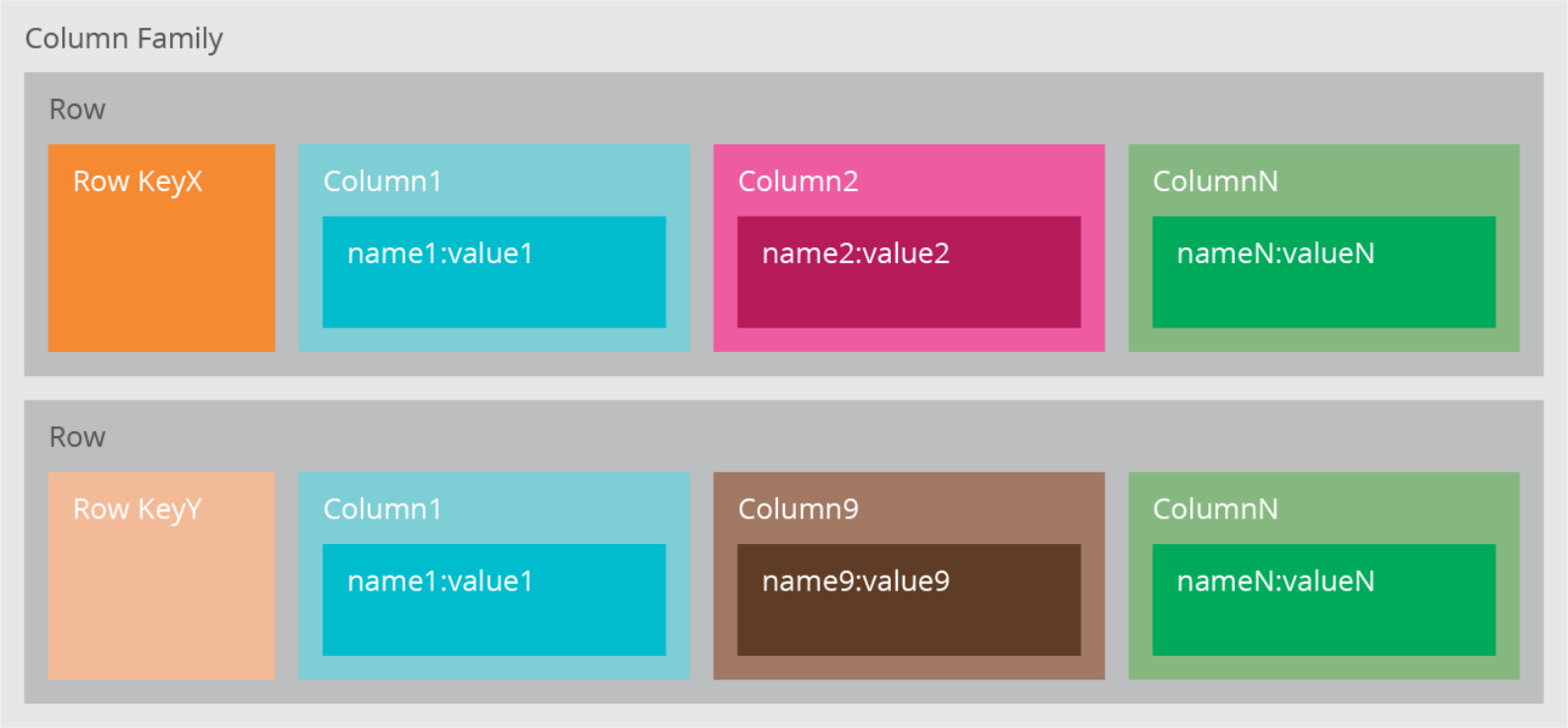
Column family store

- Column-family databases store data in column families as rows that have many columns associated with a row key
- Column families are groups of related data that is often accessed together
- Ex: For a Customer, we would often access their Profile information at the same time, but not their Orders

https://www.thoughtworks.com/insights/blog/nosql-databases-overview

The Center of Applied Data Science

Column family store



The Center of Applied Data Science

Column family store

- When a column consists of a map of columns, then we have a super column. A super column consists of a name and a value which is a map of columns. Think of a super column as a container of columns.

The Center of Applied Data Science

Column Stores

- Effective to store sparse data: Null cells are not stored
- Effective to store multi-value cells

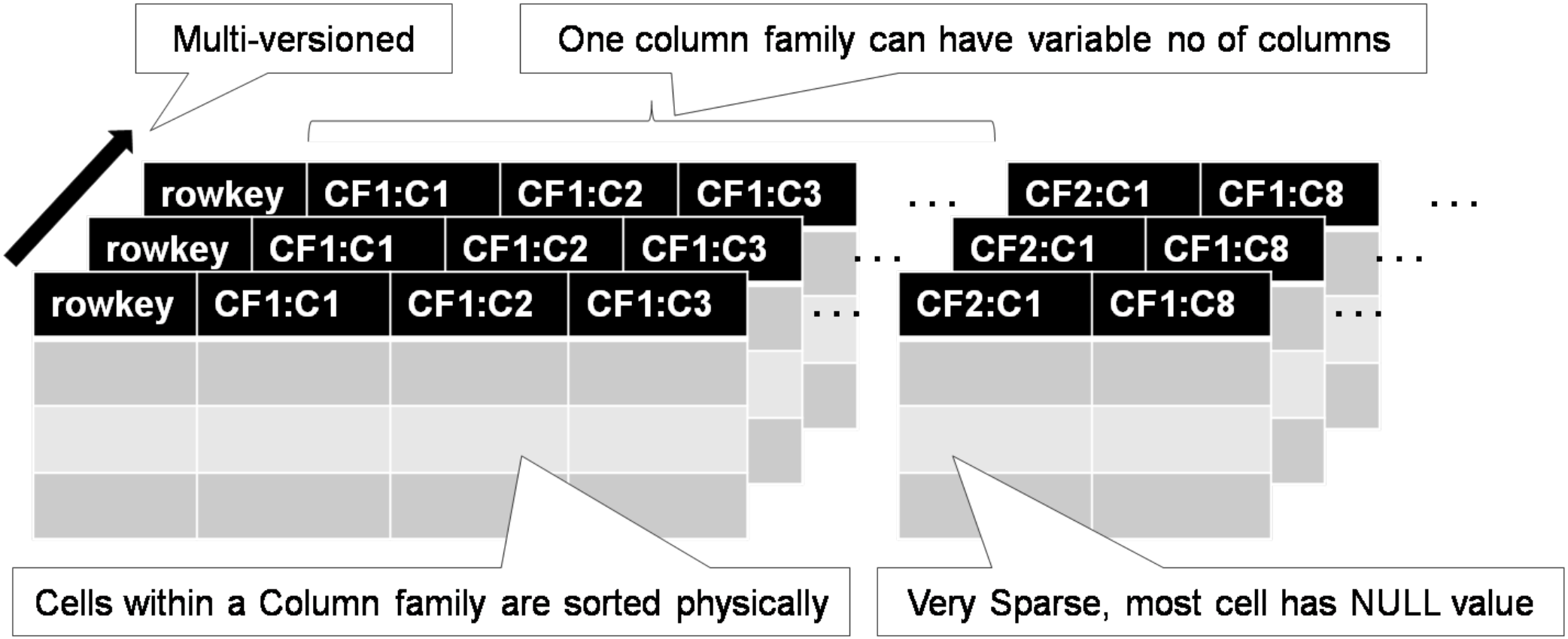
https://dzone.com/articles/bigtable-model-cassandra-and

Wide-Columns store: The column family

- In RDBMS, each row have a fixed set of columns, => difficult to support columns with multi-value attributes.
- This model introduces the "Column Family": a row has a fixed number of "column family" but within the "column family", a row can have a variable number of columns that can be different in each row.

The Center of Applied Data Science

The column family



The Center of Applied Data Science

Technologies

- Cassandra (by datasax), opensource

- BigTable (by Google)
- Hbase
- Hypertable
- Amazon DynamoDB

The Center of Applied Data Science

Resources

- https://www.datastax.com/dev/blog/basic-rules-of-cassandra-data-modeling
- https://academy.datastax.com/resources/getting-started-time-series-data-modeling

Column family databases usage

Useful for

- maintaining counters
- expiring usage
- heavy write volume such as log aggregation

Should be avoid for systems that are in early development, changing query patterns

The Center of Applied Data Science

GRAPH DATABASE

The Center of Applied Data Science

Graph database

- This kind of database is designed for data whose relations are well represented as a graph consisting of elements interconnected with a finite number of relations between them.
- Examples ?

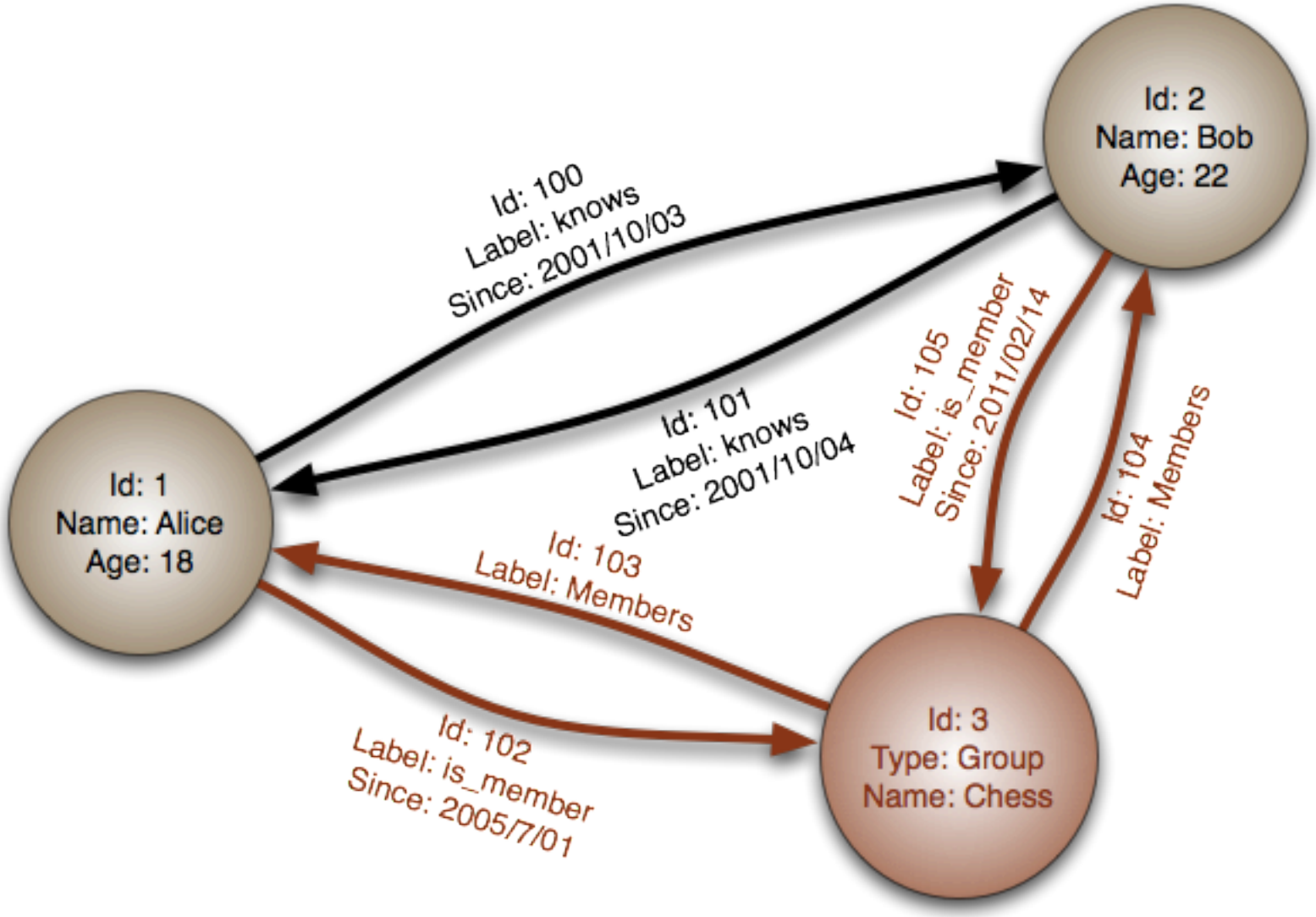
The Center of Applied Data Science

Graph database

- This kind of database is designed for data whose relations are well represented as a graph consisting of elements interconnected with a finite number of relations between them.
- Examples: social relations, public transport links, road maps or network topologies.

The Center of Applied Data Science

Graph database



The Center of Applied Data Science

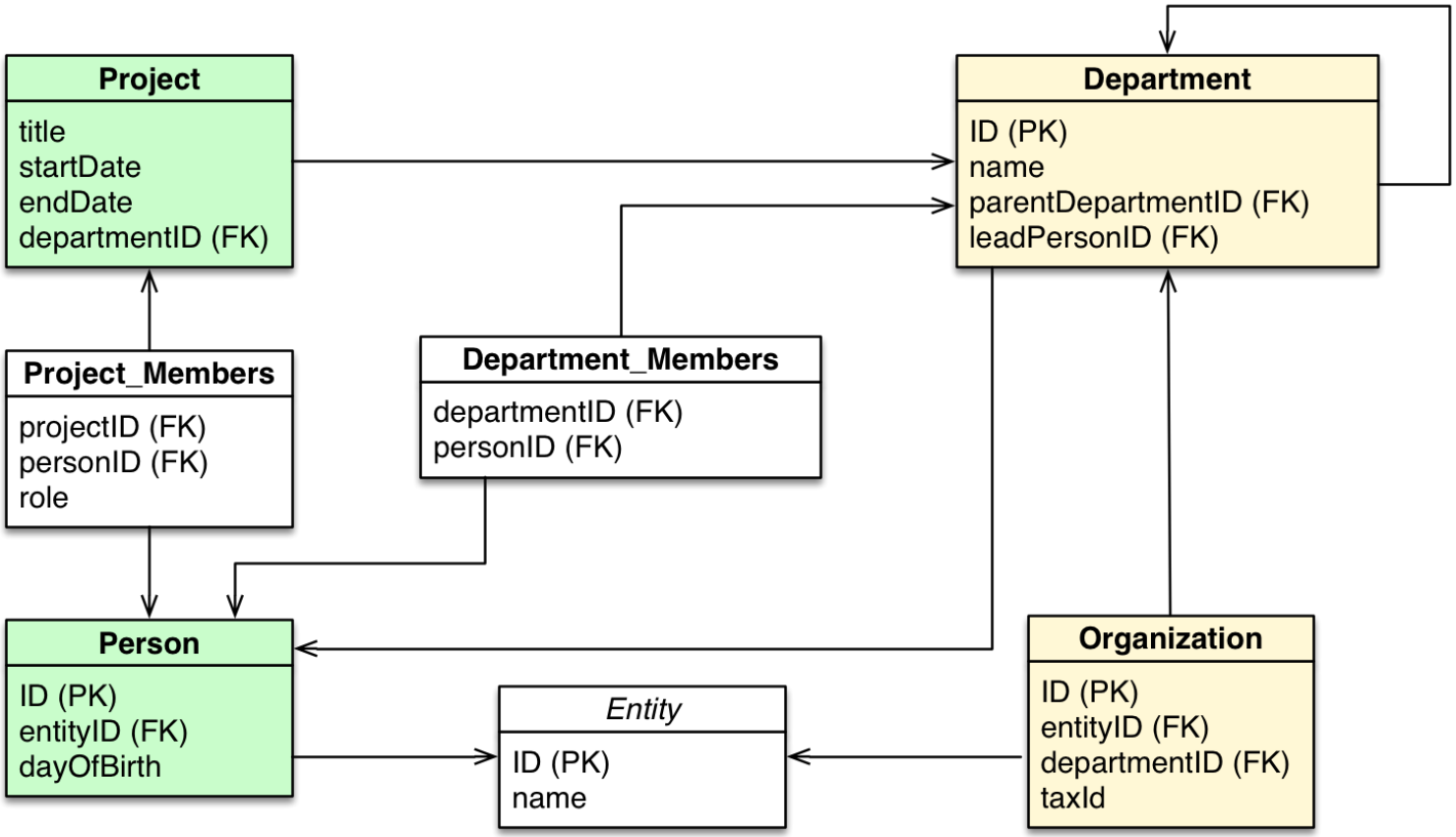
Contrast with RDBMS

- In conventional relational databases, links between data are stored in the data, and queries search for this data within the store and use the *join* concept to collect the related data
- Graph databases, by design, allow simple and fast retrieval of complex hierarchical structures that are difficult to model in relational systems

The Center of Applied Data Science

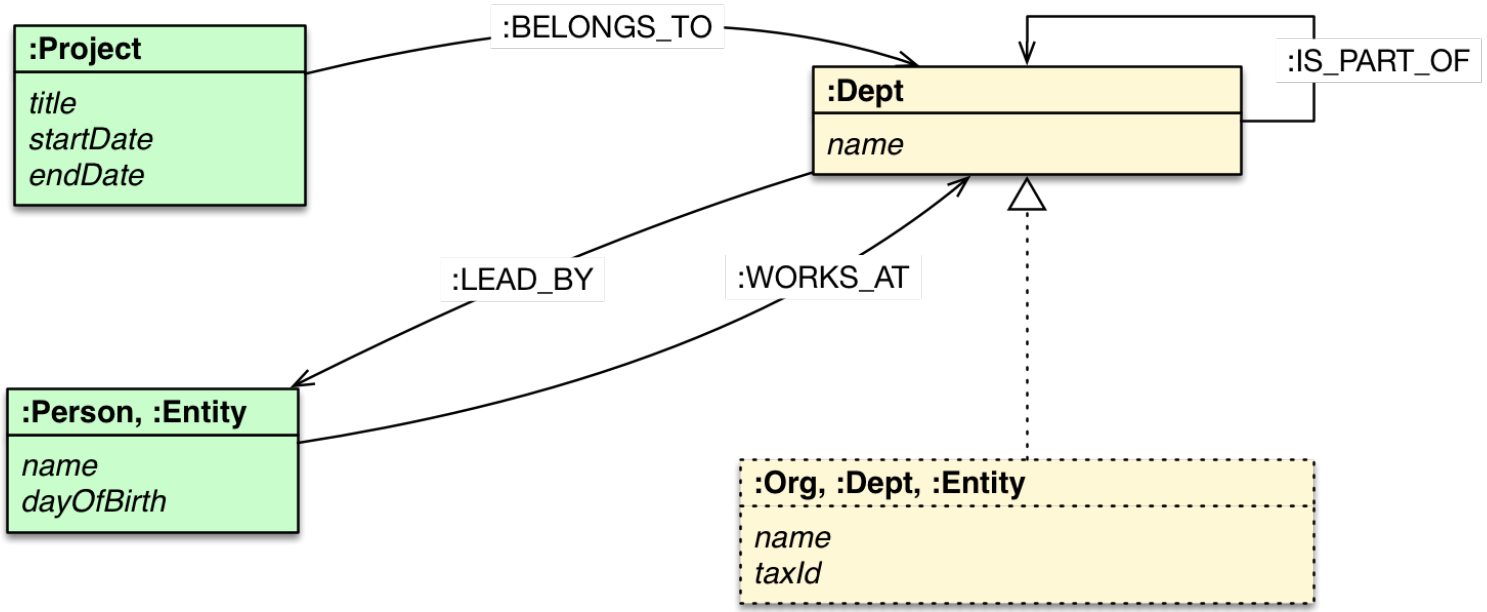
Language

Retrieving data from a graph database requires a query language other than SQL, which was designed for relational databases and does not elegantly handle traversing a graph


Project
title
startDate
endDate
departmentID (FK)
Department
ID (PK)
name
parentDepartmentID (FK)
leadPersonID (FK)
Project_Members
projectID (FK)
personID (FK)
role
Department_Members
departmentID (FK)
personID (FK)
Person
ID (PK)
entityID (FK)
dayOfBirth
Entity
ID (PK)
name
Organization
ID (PK)
entityID (FK)
departmentID (FK)
taxId

The Center of Applied Data Science

Relational data



The Center of Applied Data Science

Data model transformation

- Each entity table is represented by a label on nodes
- Each row in a entity table is a node
- Columns on those tables become node properties

It is important to have an understanding of the graph model before you start to import data.

from RDBMS to Graph (Neo4j)

https://neo4j.com/developer/graph-db-vs-rdbms/

Technologies

- Neo4j
- InfiniteGraph
- OrientDB

The Center of Applied Data Science

Graph databases usage

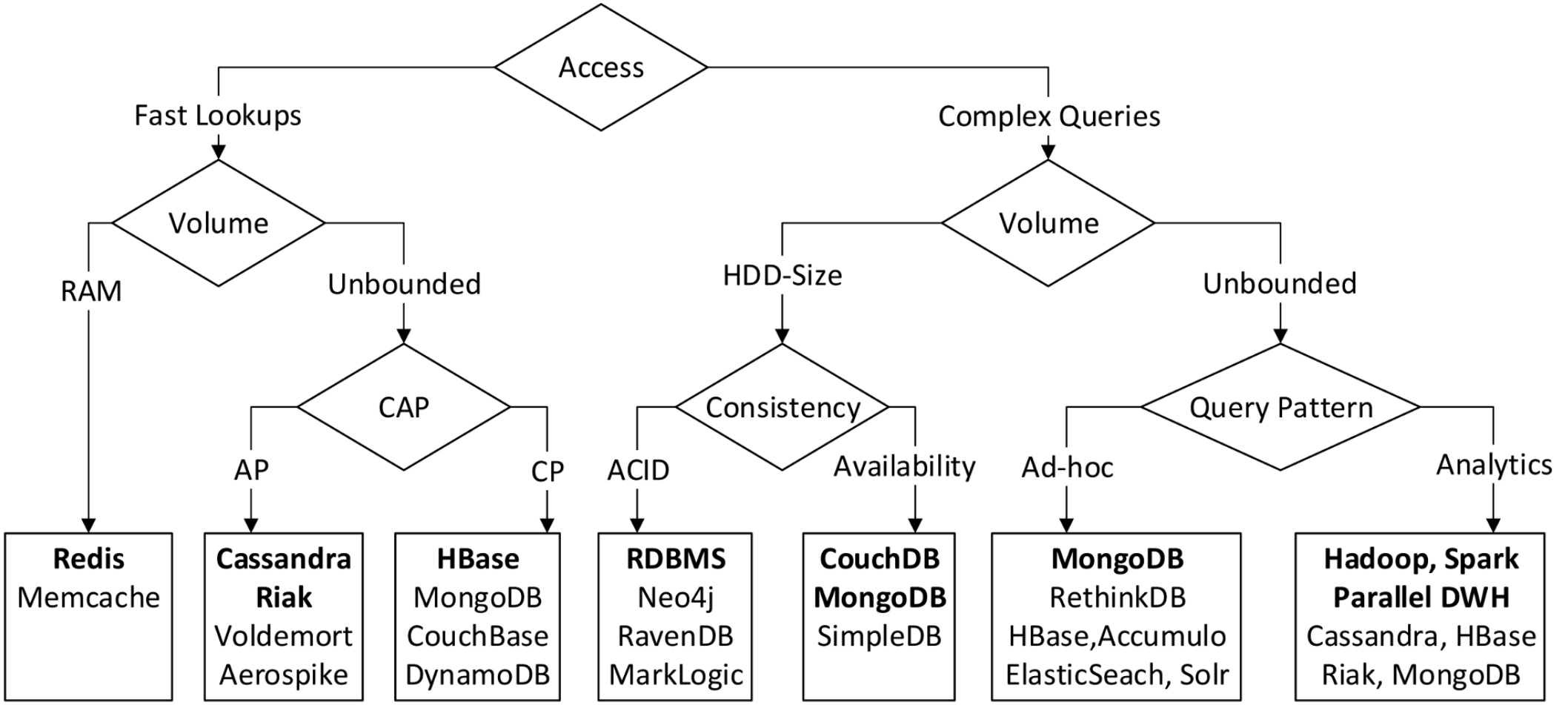
Well suited to problem spaces where we have connected data, such as:

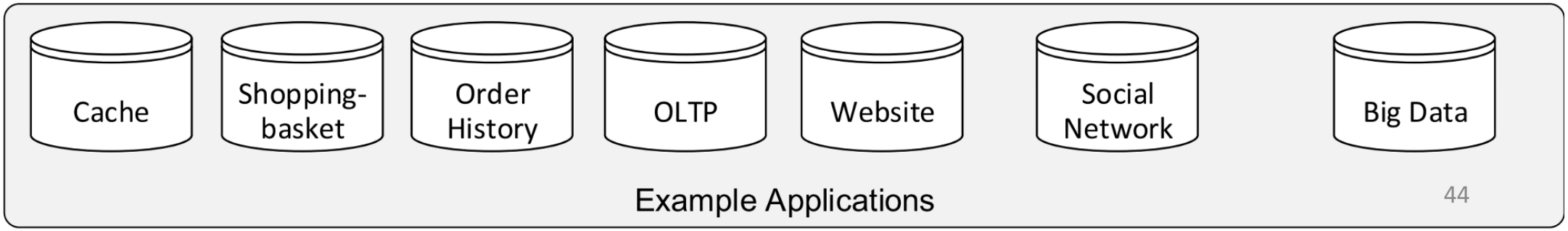
- social networks
- spatial data
- routing information for goods and money
- recommendation engines

The Center of Applied Data Science

So much choice…

HOW TO CHOOSE


Access
Fast Lookups
Complex Queries
Volume
RAM
Unbounded
CAP
AP
CP
Volume
HDD-Size
Unbounded
Consistency
ACID
Availability
Query Pattern
Ad-hoc
Analytics
Redis
Memcache
Cassandra
Riak
Voldemort
Aerospike
HBase
MongoDB
CouchBase
DynamoDB
RDBMS
Neo4j
RavenDB
MarkLogic
CouchDB
MongoDB
SimpleDB
MongoDB
RethinkDB
HBase,Accumulo
ElasticSeach, Solr
Hadoop, Spark
Parallel DWH
Cassandra, HBase
Riak, MongoDB


Cache
Shopping-basket
Order History
OLTP
Website
Social Network
Big Data
Example Applications

The Center of Applied Data Science

	Functional Requirements							
	Scan Queries	ACID Transactions	Conditional Writes	Joins	Sorting	Filter Query	Full-Text Search	Analytics
Mongo	x		x		x	x	x	x
Redis	x	x	x					
HBase	x		x		x			x
Riak							x	x
Cassandra	x		x		x		x	x
MySQL	x	x	x	x	x	x	x	x

The Center of Applied Data Science

	Non-functional Requirements										
	Data Scalability	Write Scalability	Read Scalability	Elasticity	Consistency	Write Latency	Read Latency	Write Throughput	Read Availability	Write Availability	Durability
Mongo	x	x	x		x	x	x		x		x
Redis			x		x	x	x	x	x		x
HBase	x	x	x	x	x	x		x			x
Riak	x	x	x	x		x	x	x	x	x	x
Cassandra	x	x	x	x		x		x	x	x	x
MySQL			x		x						x

Thank you.

The Center of Applied Data Science