

Introduction to Bridge Management Using



By:

M.Zakeri, M.Amirian, A.H. Sayadabdi

Iran University of Science and Technology

Fall 2016

Contents at a Glance

- What is OrientDB?
- Top 10 Key Advantages
- Data Modeling
- Query Language
- Distributed Architecture
- Getting Start!

Contents at a Glance

- Bridge Management System (BMS)
- BMS Design Challenges
- Bridge Information Network as Solution
- References

What is OrientDB?

- The first **Multi-Model** Open Source **NoSQL** DBMS.
- Combines the power of **graphs** and the flexibility of **documents** into one scalable, high-performance operational database.

Advantages for Going with OrientDB

- Built for Speed
 - **Big Data Equipped**
 - OrientDB handles relationships as physical links to the records, assigned only once, when the edge is created $O(1)$.
 - Compare this to an RDBMS that “computes“ the relationship every single time you query a database $O(\text{Log}N)$.
 - **With OrientDB, traversing speed is not affected by the database size. It is always constant, whether for one record or 100 billion records.**
 - This is critical in the **age of Big Data!**

Advantages for Going with OrientDB

- Unmatched Flexibility

	DBMS			
Features & Capabilities	OrientDB	MongoDB	Neo4j	MySQL
Operational Database**	✓	✓		✓
Graph Database	✓		✓	
Document Database	✓	✓		
Object-Oriented Concepts	✓			
Schema-full, Schema-less, Schema mix	✓			
User and Role Security	✓	✓		✓
Record Level Security	✓			
Record Level Locking	✓		✓	✓
SQL	✓			✓

Advantages for Going with OrientDB

- Unmatched Flexibility

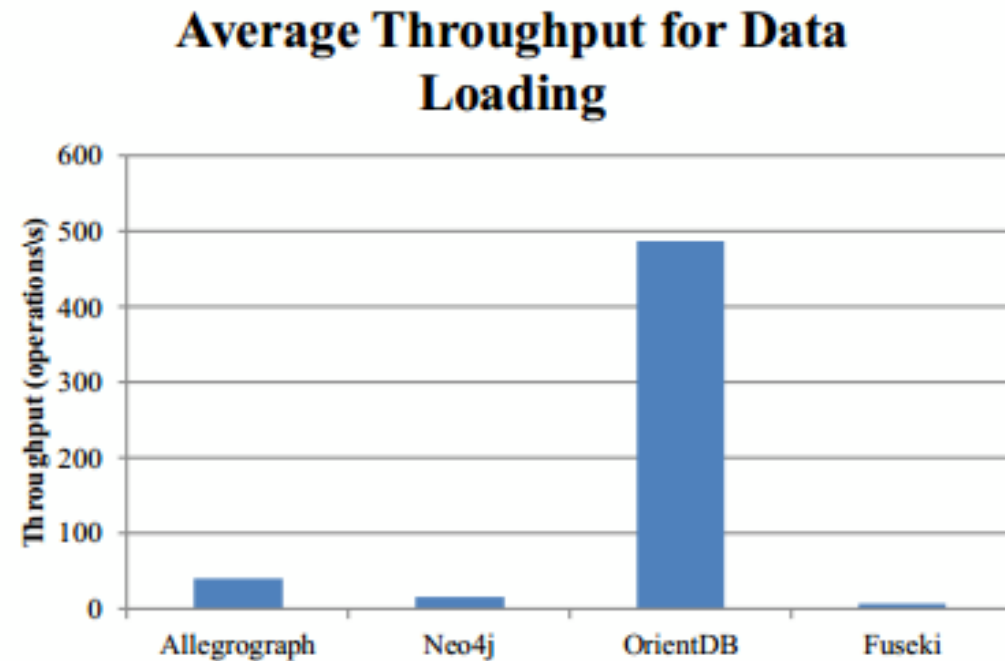
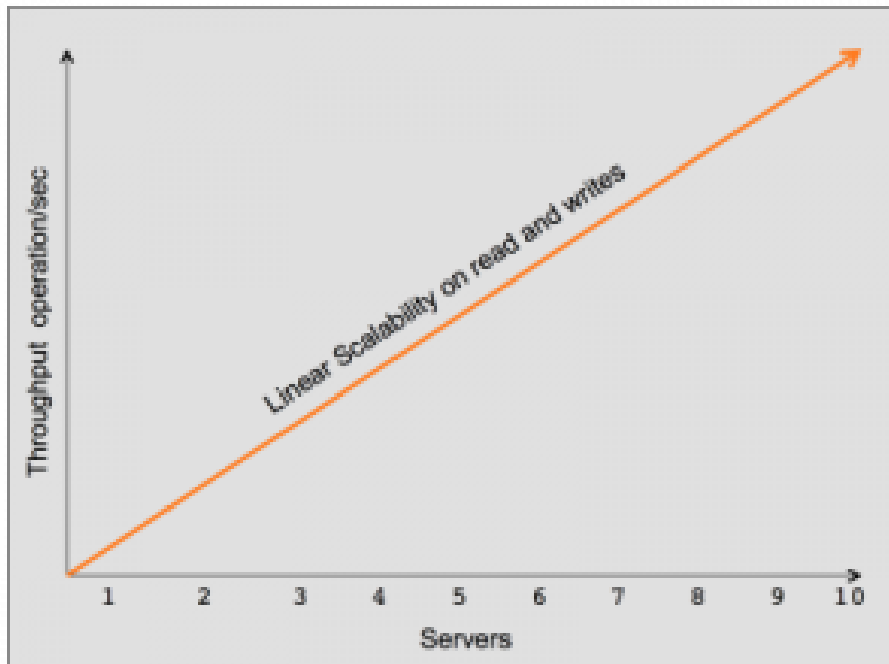
ACID Transaction	✓		✓	✓
Relationships (Linked Documents)	✓		✓	✓
Relationship Traversing	O(1)	O(LogN)	O(1)	O(LogN)
Custom Data Types	✓	✓		✓
Embedded Documents	✓	✓		
Multi-Master Replication	✓			
Sharding	✓	✓		
Elastic Scalability with Zero Configuration	✓			
Server-Side Functions	✓	✓		✓
Native HTTP Rest/JSON	✓	✓	✓	
Commercial Friendly License	✓	✓		✓
Embeddable with No Restrictions	✓			

Advantages for Going with OrientDB

- Zero-Configuration Multi-Master Architecture
 - **OrientDB supports a Multi-Master + Sharded architecture.**
 - **All the servers are masters.**
 - It provides **horizontal scalability** and **reliability**.

Advantages for Going with OrientDB

- Zero-Configuration Multi-Master Architecture



Advantages for Going with OrientDB

- Easy to Install and Use
- Low Total Cost of Ownership (TCO)
- Enterprise Ready
- Unparalleled in the Market
- Solid Product Stability & Maturity
- Open Source
- Reputation of Technical Excellence

Data Modeling: Basic Concepts

- **Record**

- The smallest unit that you can load from and store in the database
- Records come in four types:
 1. Document
 2. BLOB (legacy RecordBytes)
 3. Vertex
 4. Edge

Data Modeling: Basic Concepts

- **Class** and **Cluster**

- In OrientDB, **classes** define records. It is closest to the concept of a *table* in Relational databases.
- The **Cluster** is a place where a **group of records** are stored.
- **Like** the Class, it is comparable with the *collection* in traditional document databases, and in relational databases with the *table*.

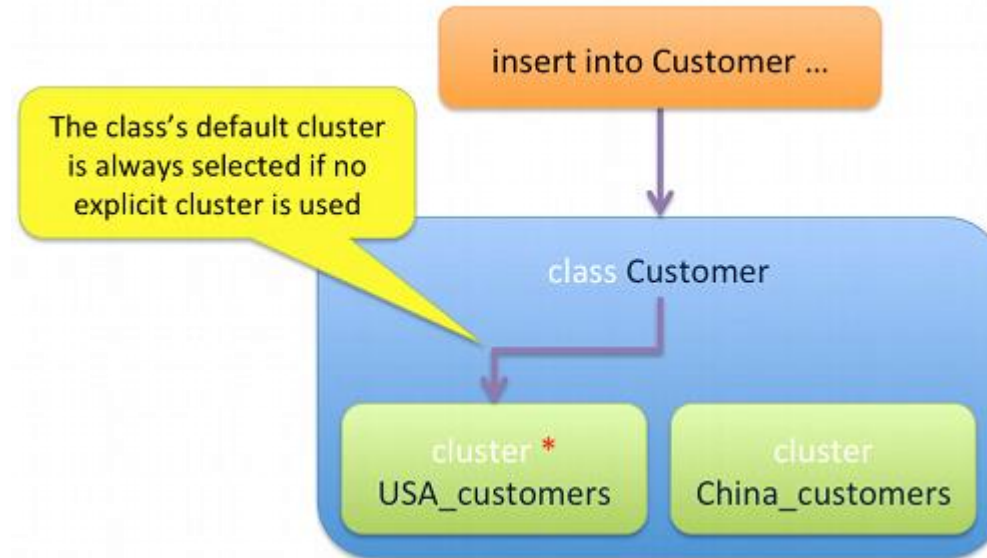
Data Modeling: Basic Concepts

- **Understanding Clusters**

- Where classes in provide you with a **logical framework** for organizing data, clusters provide **physical** or in-memory space in which OrientDB actually stores the data.
- Starting from **v2.2**, OrientDB automatically creates **multiple clusters per each class**.
 - By Default the number of clusters created is equals to the **number of CPU's cores** available on the server to improve using of **parallelism**.

Data Modeling: Basic Concepts

- **Understanding Clusters**



Data Modeling: Basic Concepts

• Record ID

- In OrientDB, each record has its own **self-assigned unique ID** within the database called Record ID or **RID**.
- It is composed of two parts:
 - #<cluster-id>:<cluster-position>
- Each database can have a maximum of 32,767 clusters, or **$2^{15} - 1$** .
- Each cluster can handle up to 9,223,372,036,780,000 records, or **2^{63}** , namely 9,223,372 trillion records.

Data Modeling: Basic Concepts

- **Relationships**

- OrientDB supports two kinds of relationships:
 - Referenced,
 - Embedded.
- It can manage relationships in a **schema-full** or **schema-less** scenario.
- There is no *JOIN* in OrientDB. Instead, it uses *LINK*.

Data Modeling: The Document Model

Relational Model	Document Model	OrientDB Document Model
Table	Collection	Class or Cluster
Row	Document	Document
Column	Key/value pair	Document field
Relationship	not available	Link

Data Modeling: The Graph Model

Relational Model	Graph Model	OrientDB Graph Model
Table	Vertex and Edge Class	Class that extends "V" (for Vertex) and "E" (for Edges)
Row	Vertex	Vertex
Column	Vertex and Edge property	Vertex and Edge property
Relationship	Edge	Edge

Query Language: SQL

- Instead of inventing "**Yet Another Query Language**", it begins with the widely used and well-understood language of **SQL**.
- Why SQL?
 - SQL is ubiquitous in the database development world.
 - It is familiar and more readable and concise than its competitors, such as **Map Reduce scripts** or **JSON** based querying.

Query Language: Examples

• Select

- orientdb> SELECT FROM `user`
- orientdb> SELECT FROM `CLUSTER:user`
- orientdb> SELECT FROM `user` WHERE `name` LIKE 'I%'

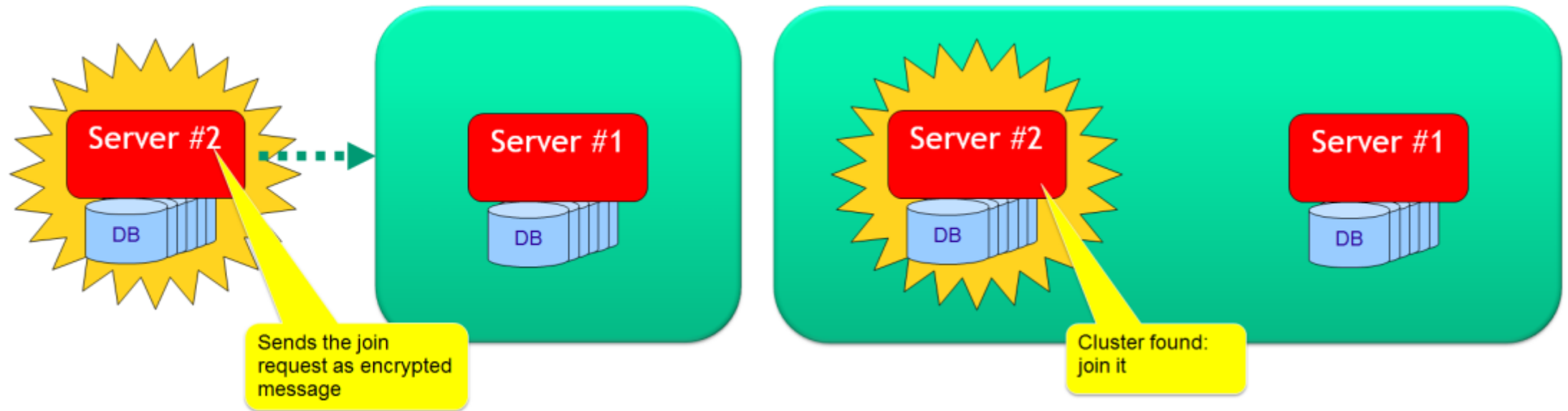
• Insert

- orientdb> INSERT INTO `Employee`(name, surname, gender) VALUES('Jay', 'Miner', 'M')

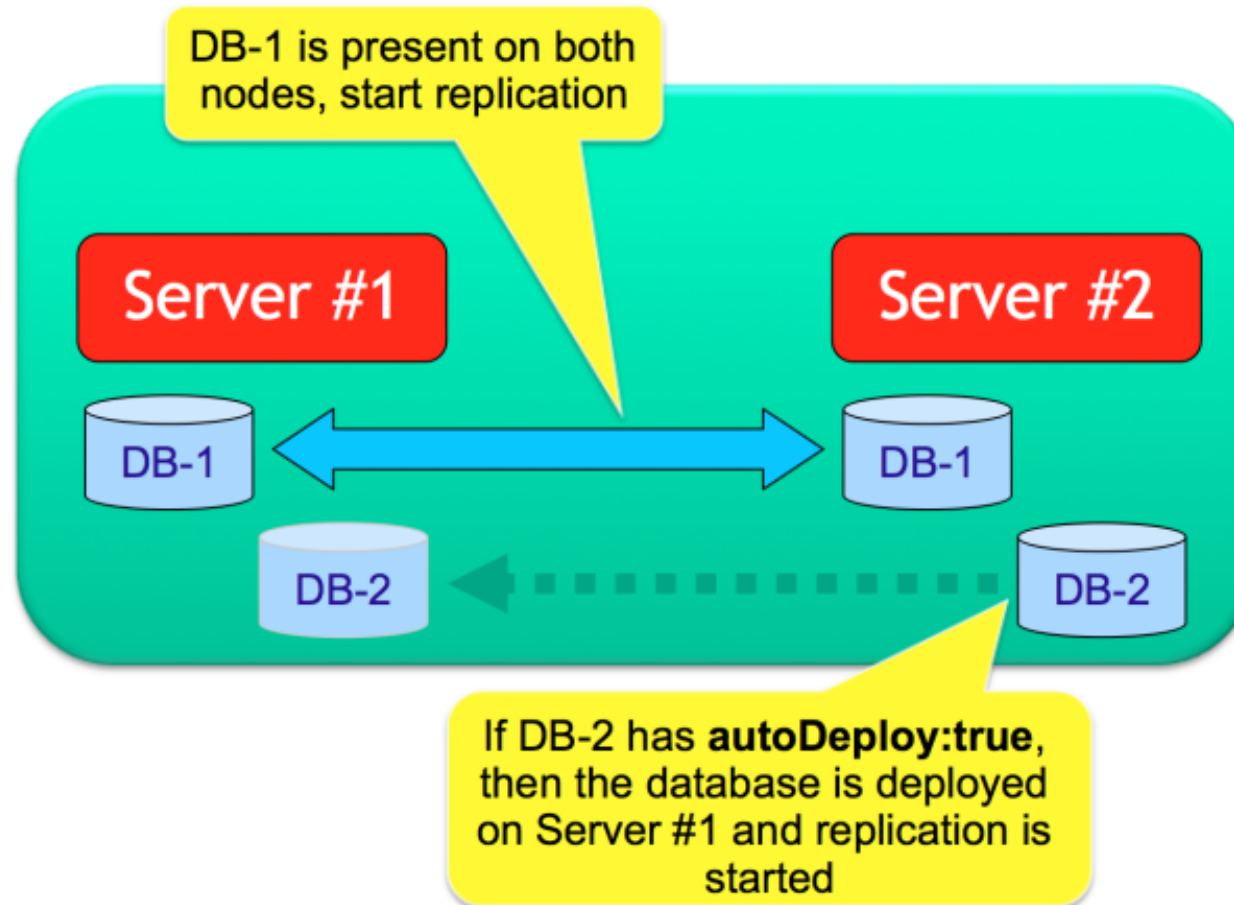
Distributed Architecture

- OrientDB can be distributed across different servers and used in different ways to achieve the maximum of **performance**, **scalability** and robustness.
- Uses the **Hazelcast** Open Source project.
- Setting up a Distributed Graph Database:
 - Prior to startup, copy the specific database directory, under `$ORIENTDB_HOME/database` to all servers.

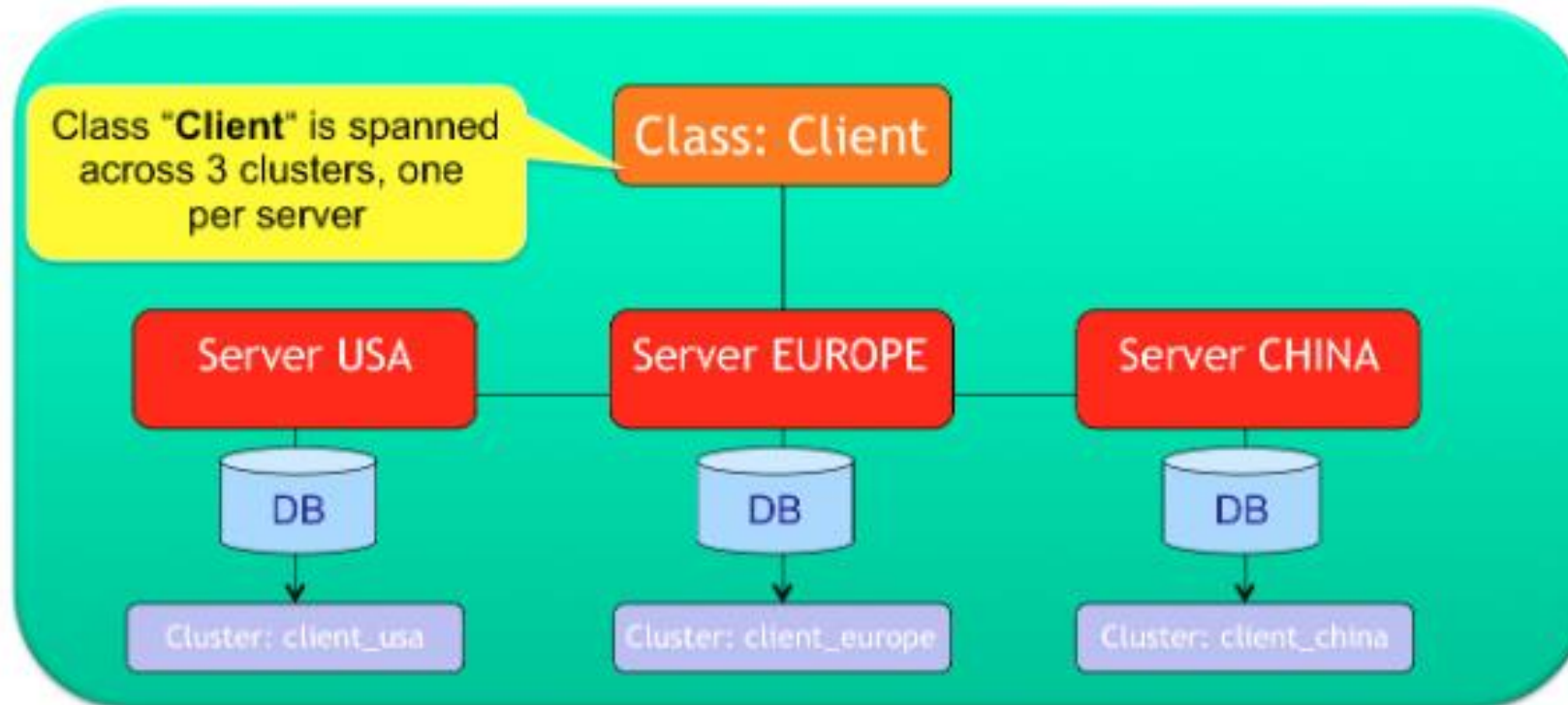
Distributed Architecture: Scaling



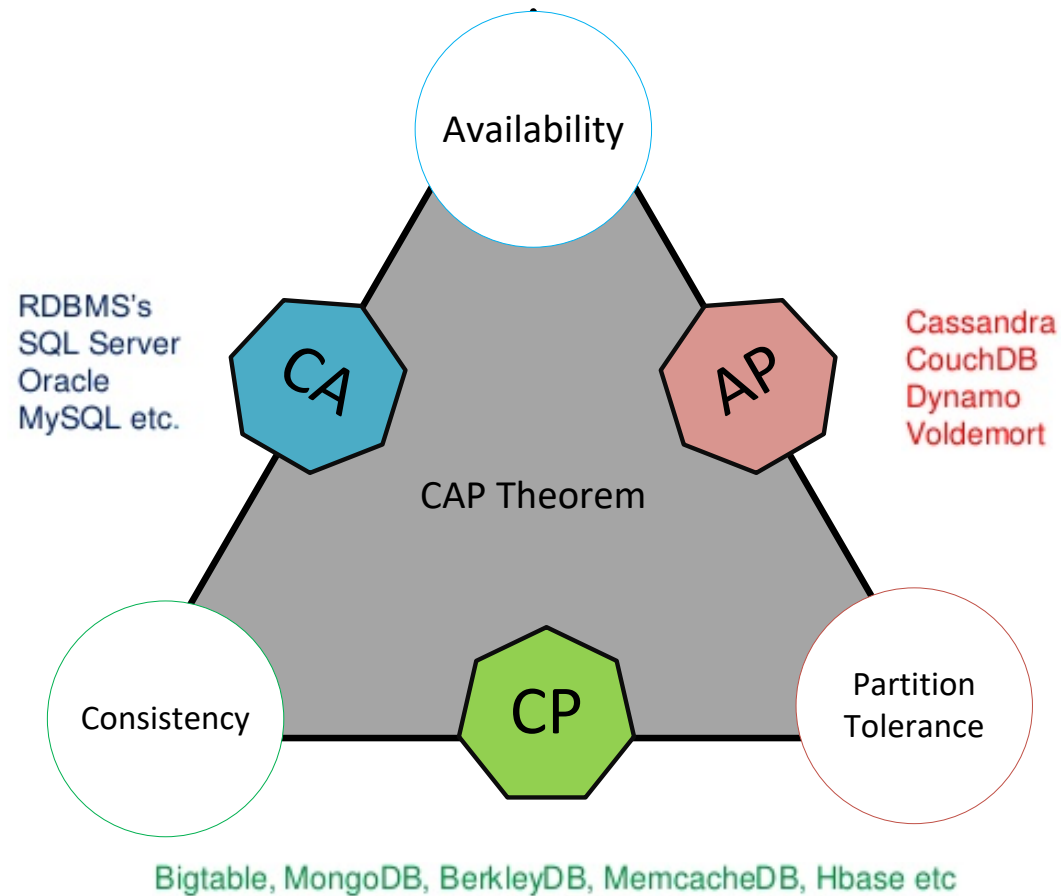
Distributed Architecture: Replication



Distributed Architecture: Sharding



OrientDB and CAP Theorem



OrientDB and CAP Theorem

- OrientDB supports both **CA** and **AP** depending of your cluster configuration.
- If you create 1 master and X slaves replicated as synchronous you have **CA**.
- If you using multi-master you could have **AP**.

Let's Start

- **Installation**

- Use Docker
- Binary Installation
- Source Code Installation

- **Run Server**

- **OrientDB Studio**

- *<http://localhost:2480>*

OrientDB Studio

The screenshot displays the OrientDB Studio interface. The top navigation bar includes 'Browse', 'Schema', 'Security', 'Graph', 'Functions', and 'DB'. The user is logged in as 'GratefulDeadConcerts (admin)'. The left sidebar shows the 'Properties' panel for a vertex with ID '#9:4'. The main canvas displays a graph visualization with a central vertex '#9:4' and several other vertices connected by edges labeled 'written_by' and 'sung_by'.

Properties Panel:

Property	Value
@rid	#9:4
@class	V
name	BERTHA
song_type	original
performances	394
type	song

Graph Visualization:

- Central vertex: #9:4 (BERTHA)
- Vertices connected by 'written_by' edges: #9:27, #9:93, #9:1, #9:7
- Vertices connected by 'sung_by' edges: #9:3, #9:50, #9:8, #9:1
- Vertex #9:9 is connected to #9:2 by a 'written_by (+1)' edge.

Bridge Management System (BMS)

- A means for managing **bridges** throughout **design, construction, operation** and **maintenance** of the bridges.
- Main part is a computer software.
- Commonly uses **RDBMS** to store and retrieve bridge data.
- Develop by each country or state.

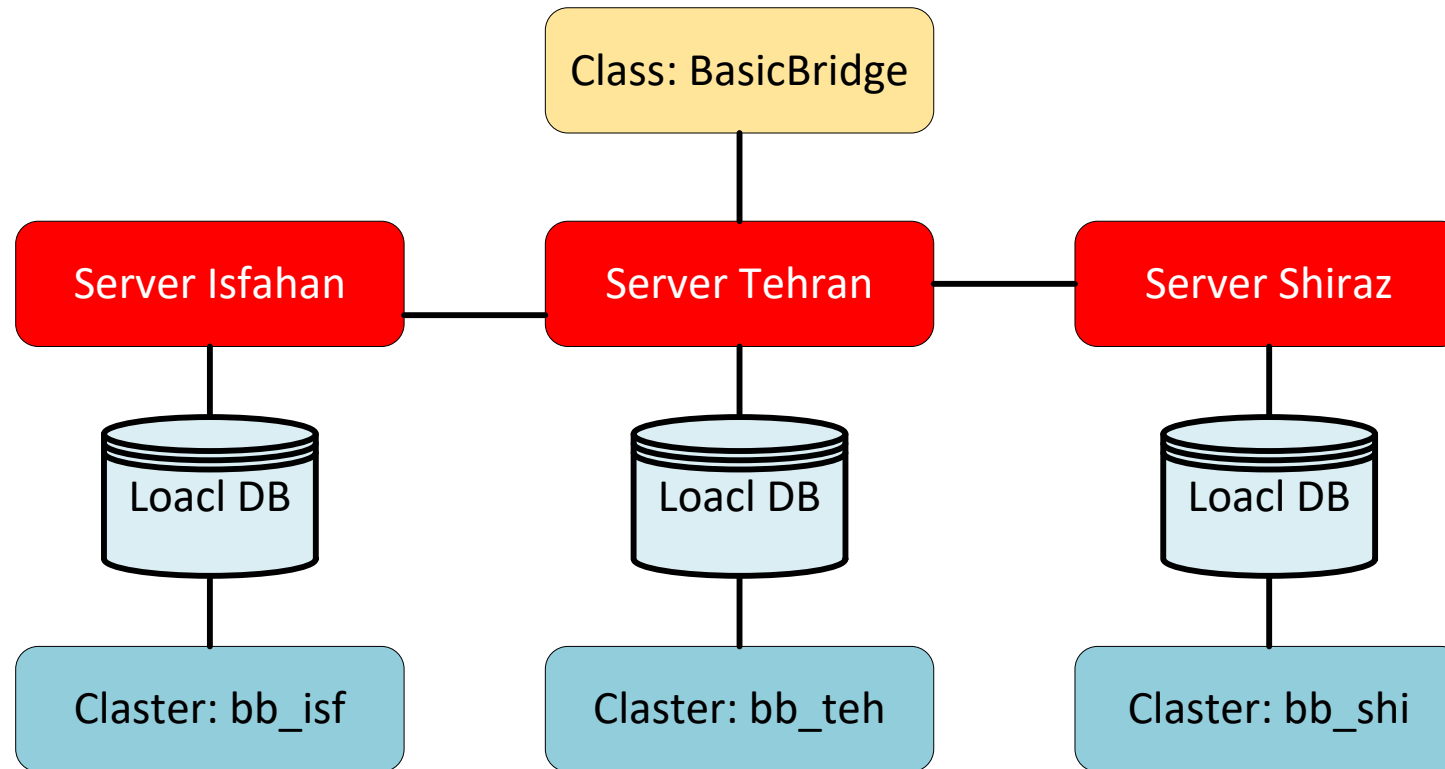
BMS Challenges

- The challenges in design is:
 - Various information,
 - **Big Data**,
 - Switching from OLTP to **OLAP**,
 - **Modeling and Decision making**,
 - **Data Visualization**.
- What is the solution?

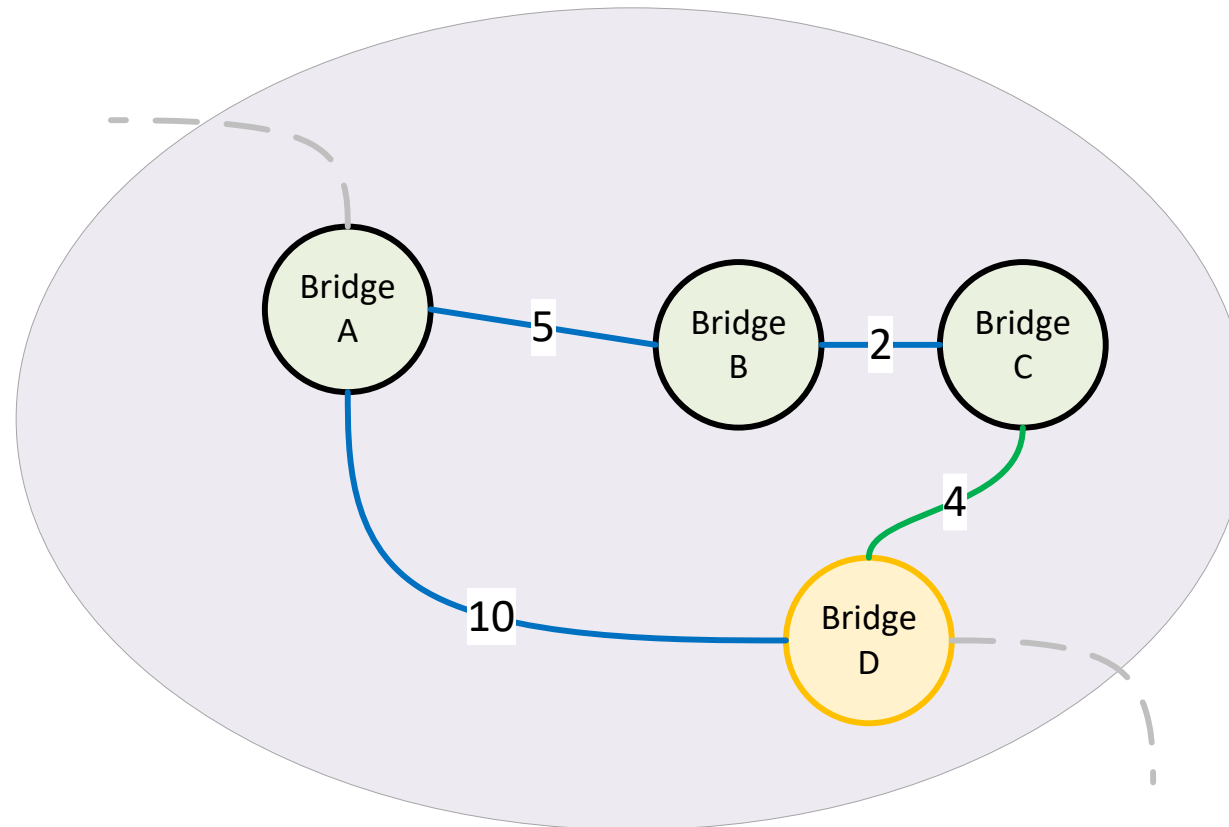
Bridge Information Network

- Using Multi-Model NoSQL Database
 - Such as **OrientDB**.
- Distributed Data.
- Using Graph Model to:
 - Data Visualization and Graph Algorithms.
 - Store additional data such as Roads.
- And achieve more!

Bridge Information Network: Clustering



Bridge Information Network: The Graph



References

1. OrientDB Official Website:
 - <http://orientdb.com/>
2. *Bridge Management a System Approach for Decision Making*
 - *Article by: Reginald W. Stratt*

Thank you for your attention!

○ Any questions?

- *m-zakeri@live.com*
- *m_amirian@comp.iust.ac.ir*
- *sayadabdi_a@comp.iust.ac.ir*

