Neo4J

Intro

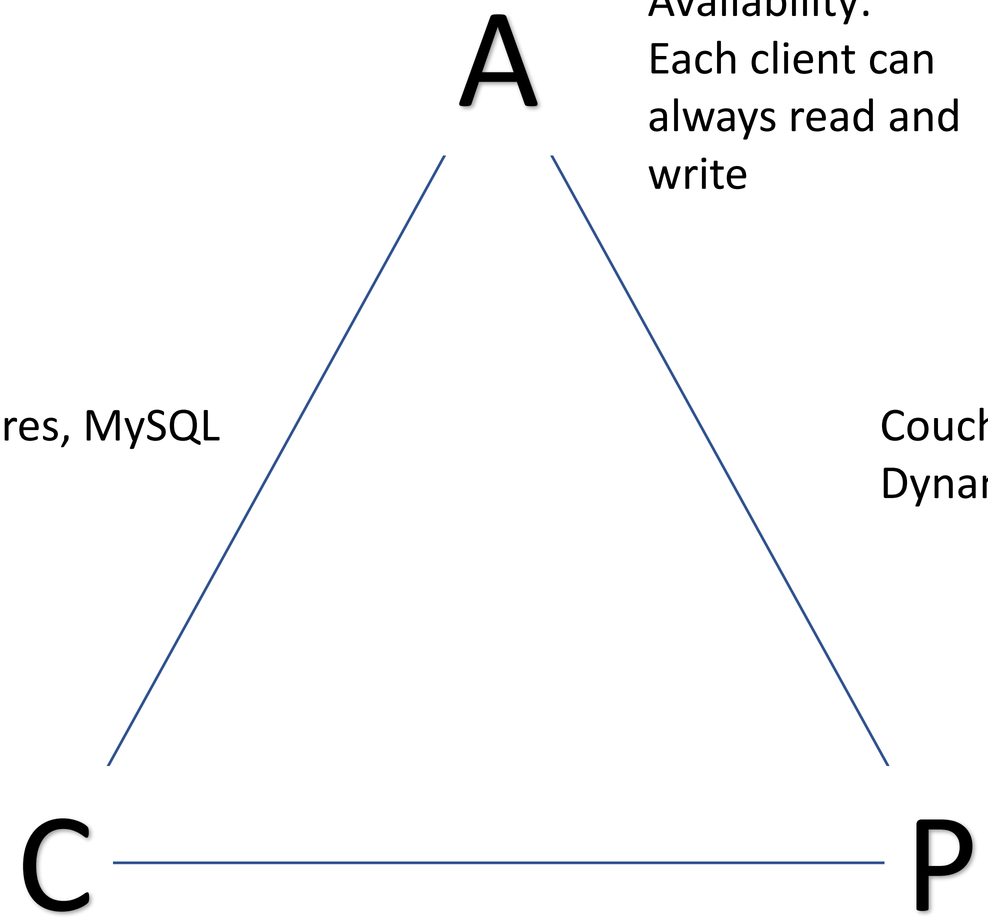
What is Neo4J

- Worlds leading graph database
- Fast traversal of nodes and relationships
- Four million hops per second
- Can store trillions of entities
- Cypher graph query language
- Scalable fault tolerant clusters

History

- Founded in 2007
- 1.0 Released in 2010
    - 34 billion node and relationship limit
- 2.0 released in 2013
- 3.0 released in 2016
    - 36m in funding secured
    - Unlimited nodes

Where does it fit in CAP Theorem



Availability:
Each client can always read and write

RDBMS, Postgres, MySQL
Neo4J

CouchDB, Cassandra, DynamoDB**,** Riak

Consistency: All clients always have the same view of data

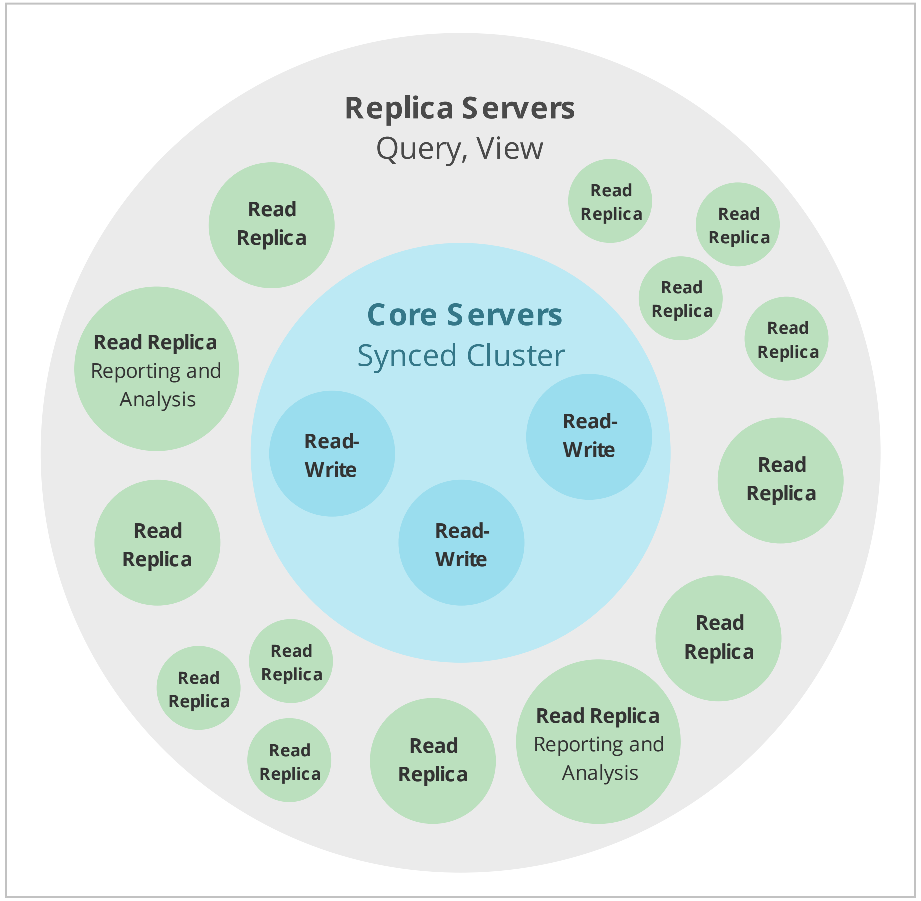
Big Table, Hbase, MongoDB, Redis

Partition Tolerance: Works well despite physical network partitions

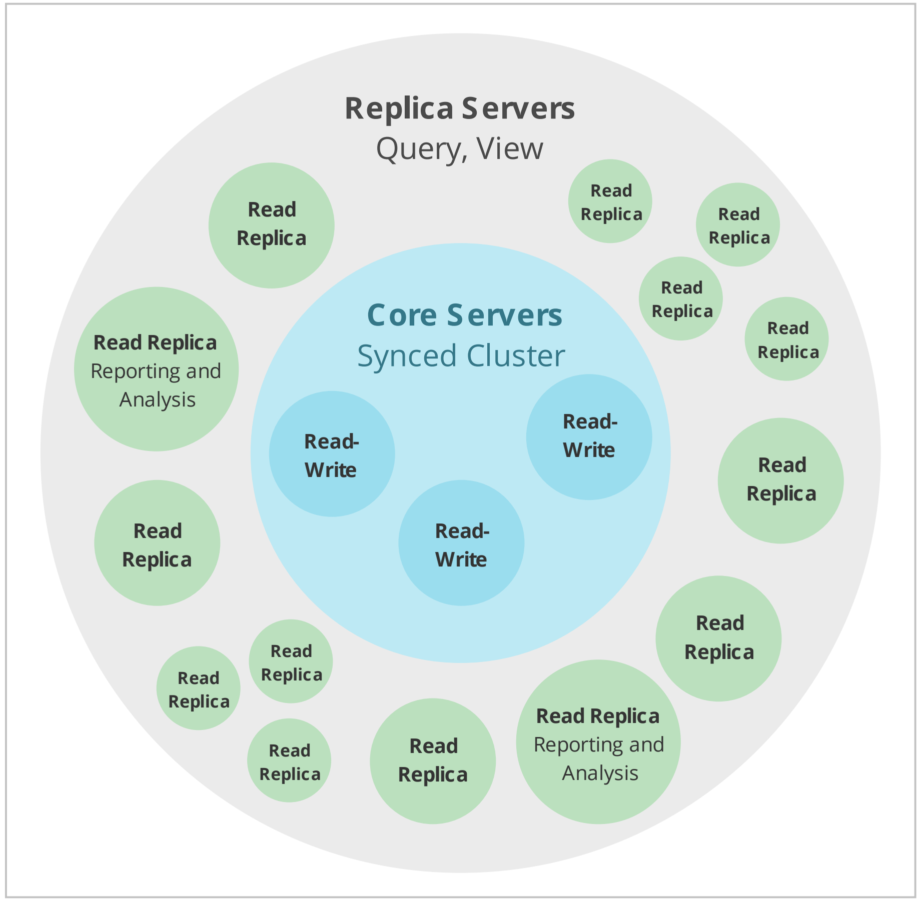
How is Neo4J Different

- Graph Storage
- Graph Processing Engine
- Relationships matter more than data points

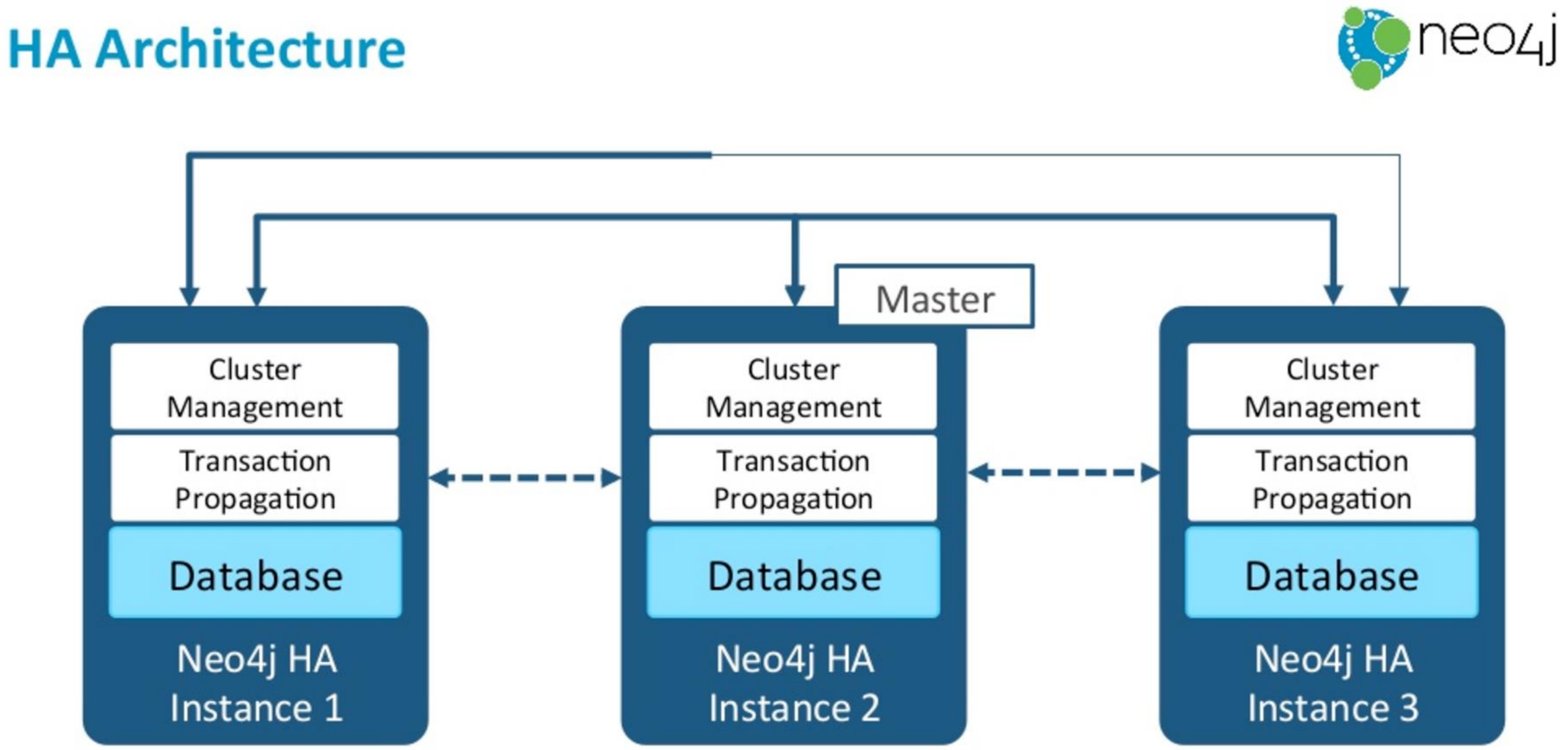
Neo4J Architecture Core Servers



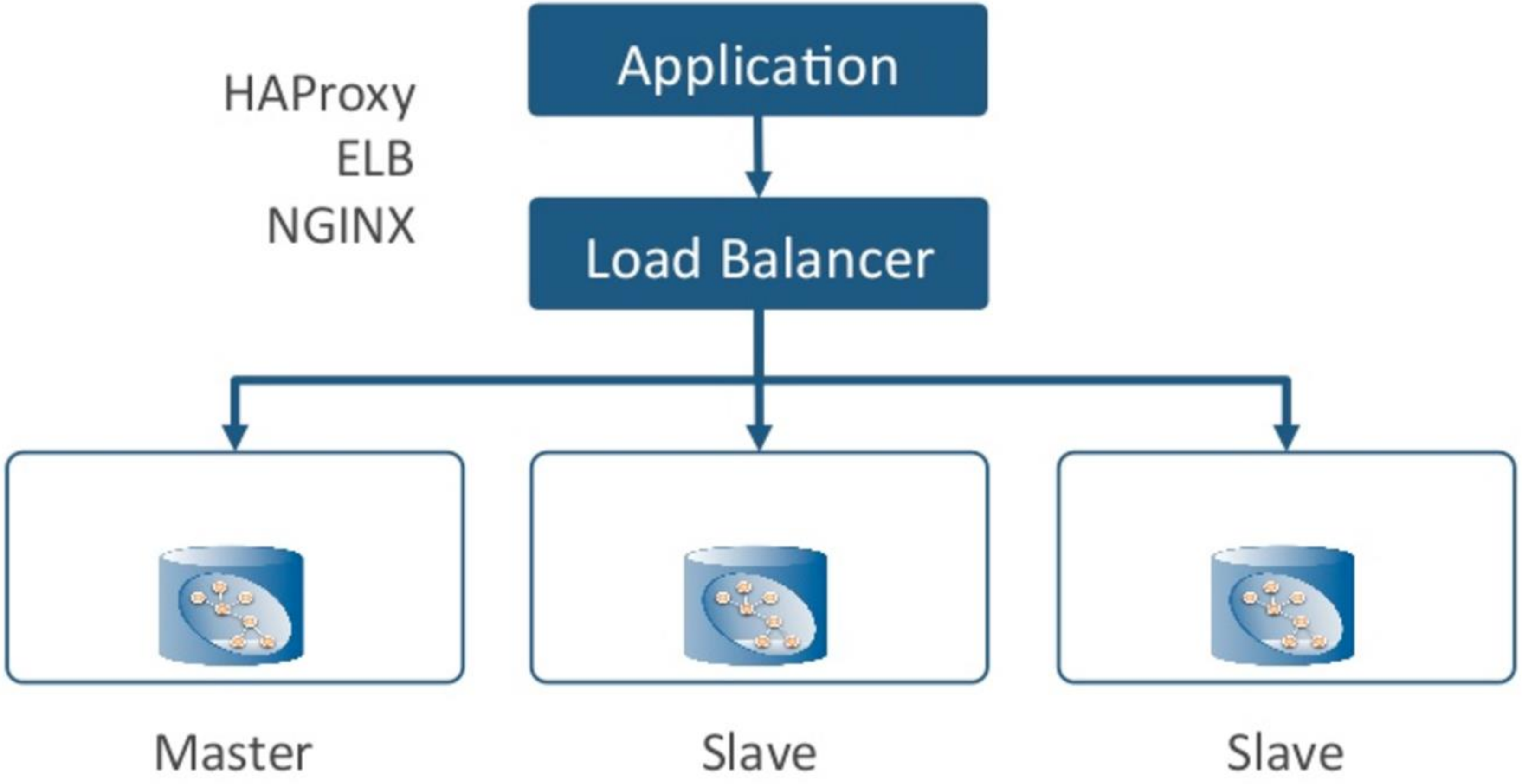
Neo4J Architecture Read Replicas



System Architecture

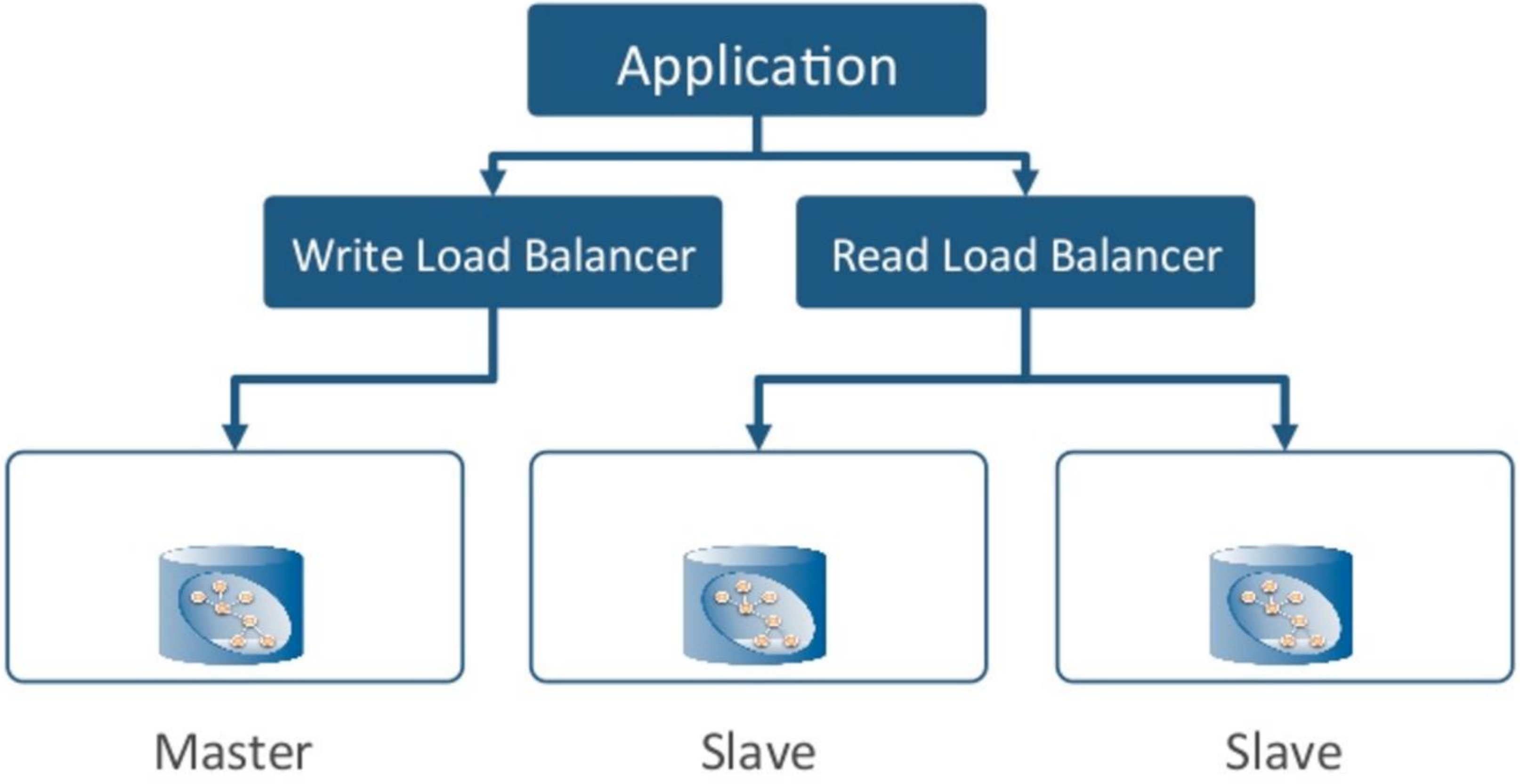


Load Balancing for Read Throughput

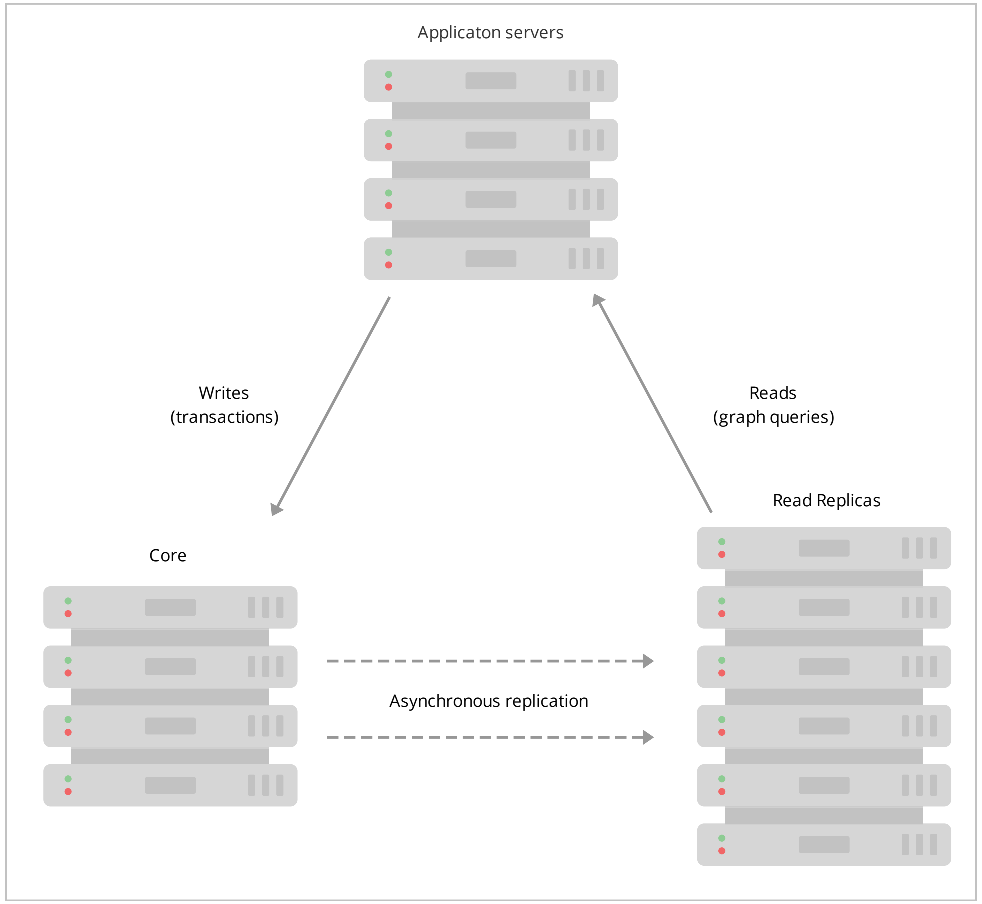


FLORIDA POLYTECHNIC UNIVERSITY

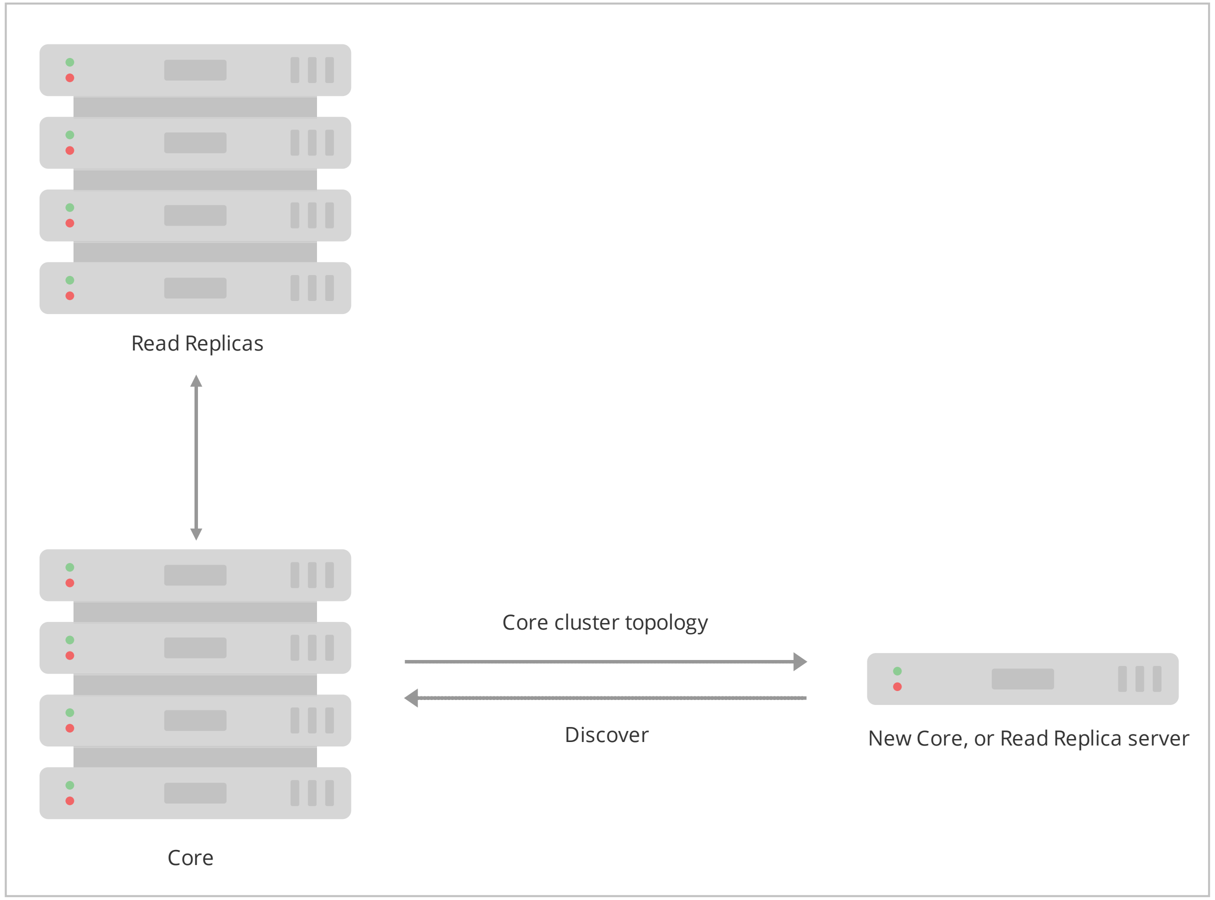
Load Balance for Writes



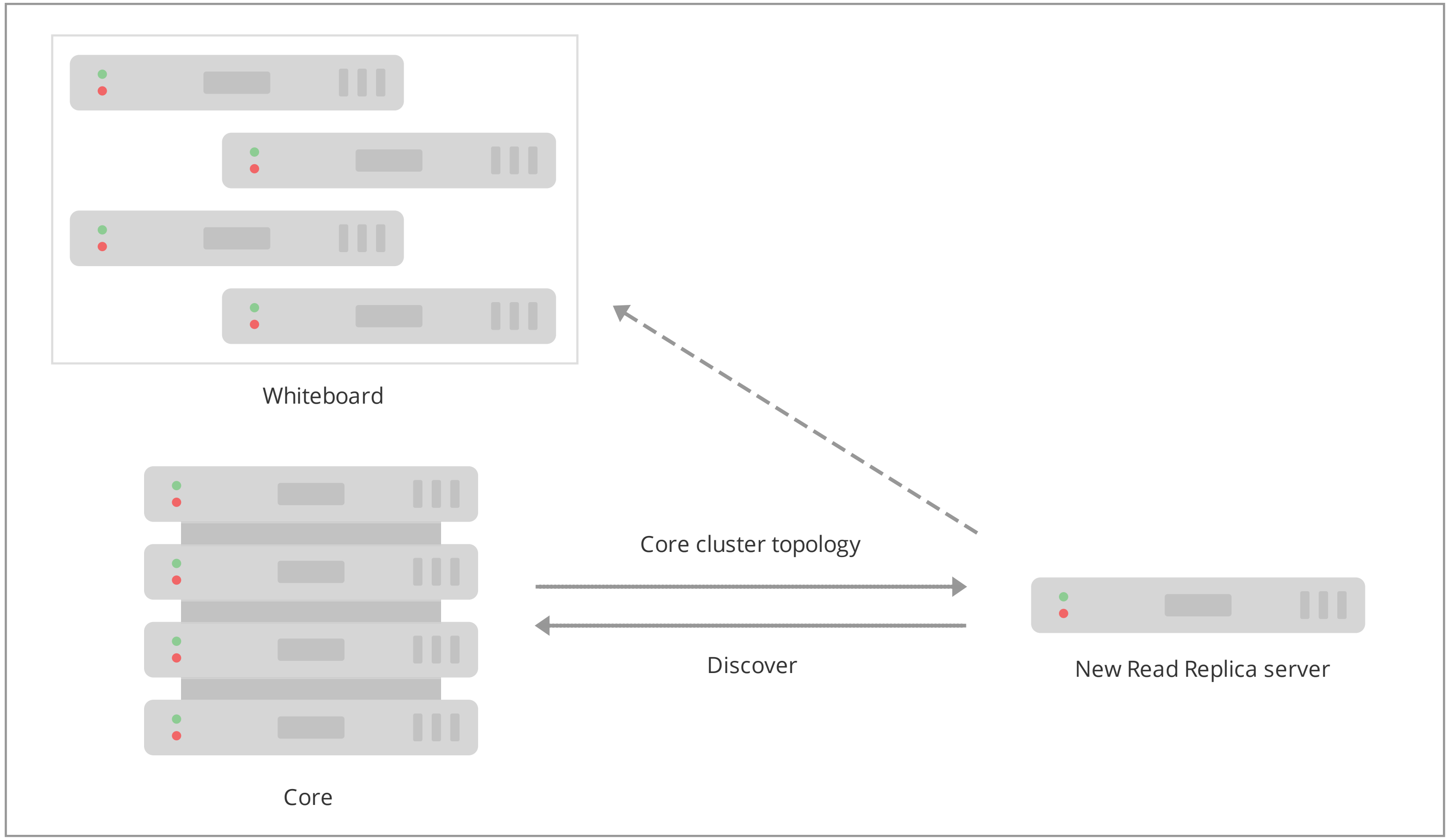
Causal Consistenty



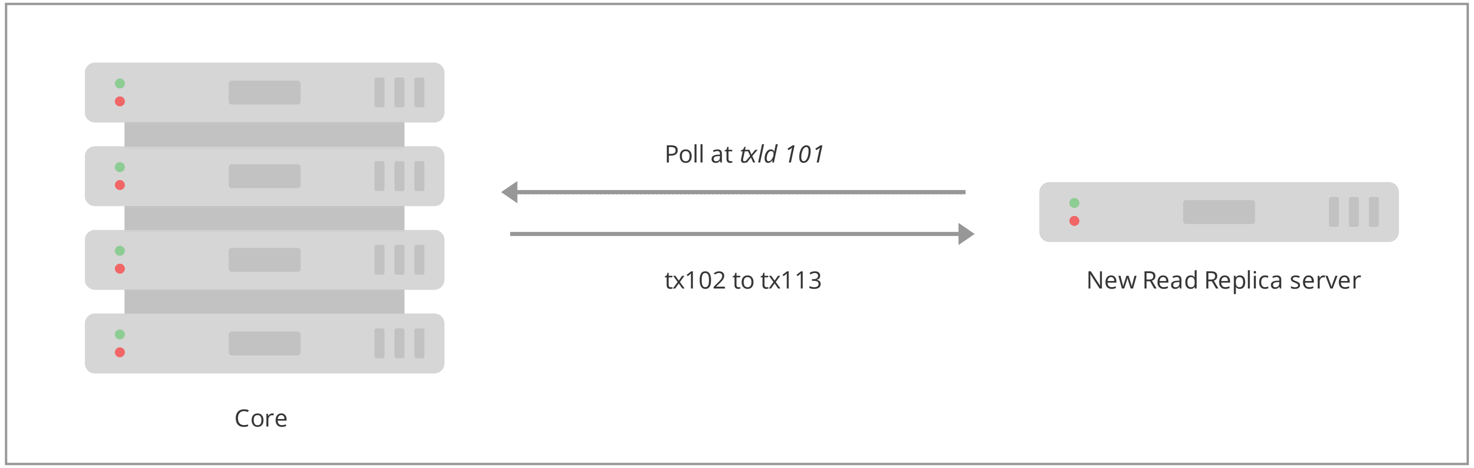
Clustering – Discovery Protocol



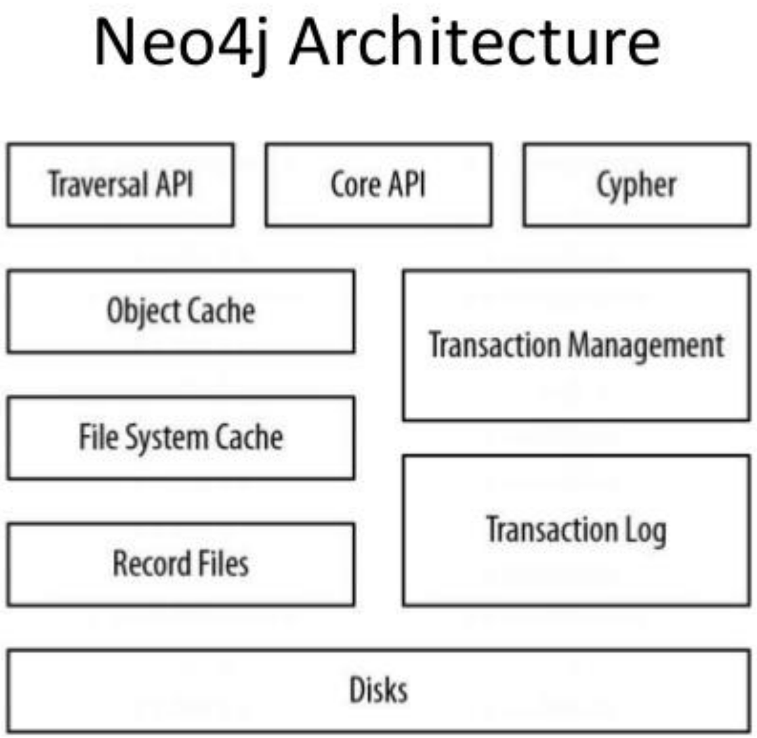
Clustering – Read Replica membership



Clustering Read Replica



Server Architecture



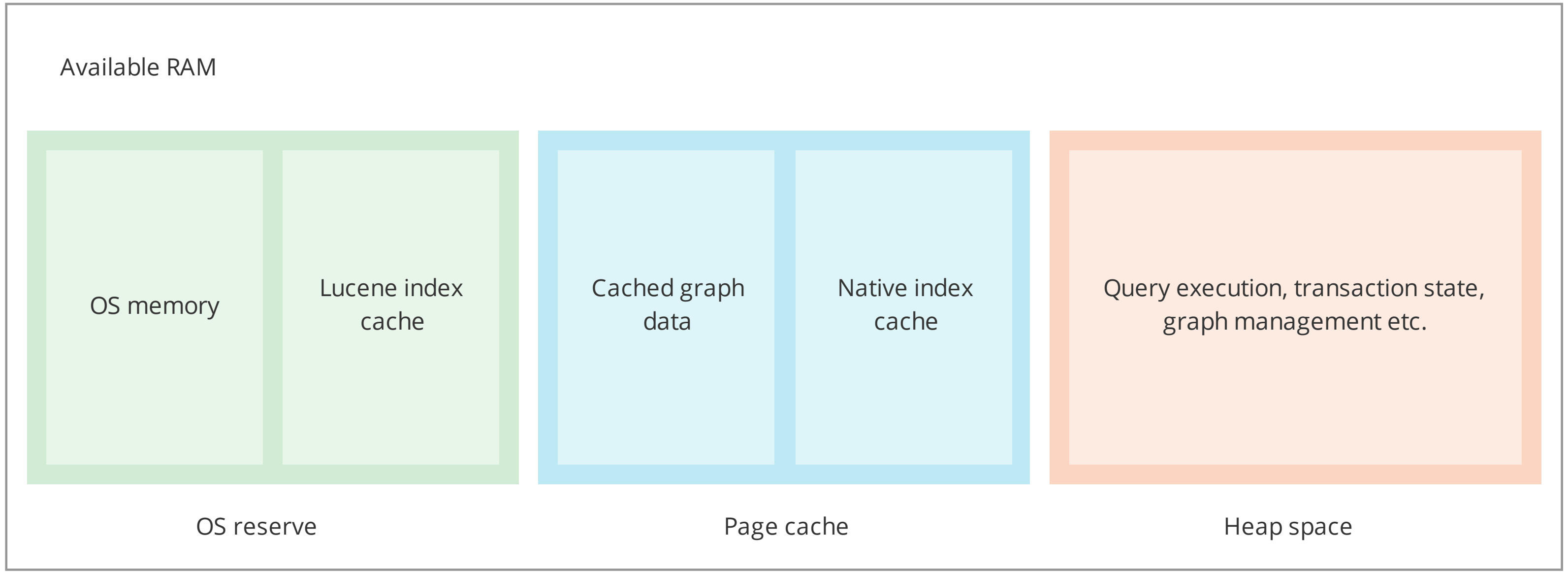
CQL

- Cypher Query Language is Neo4j's open graph query language

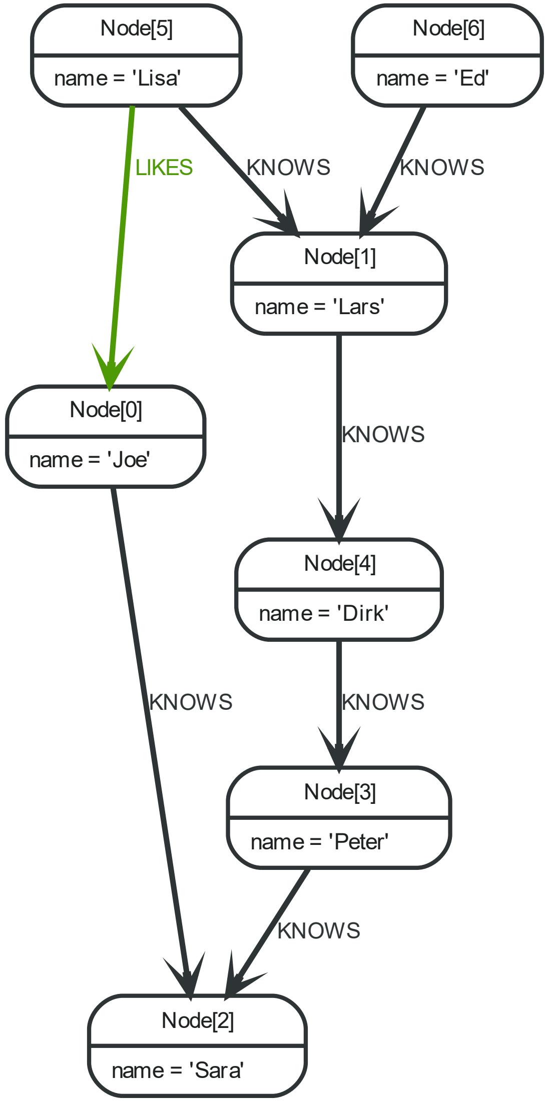
ACID Compliant

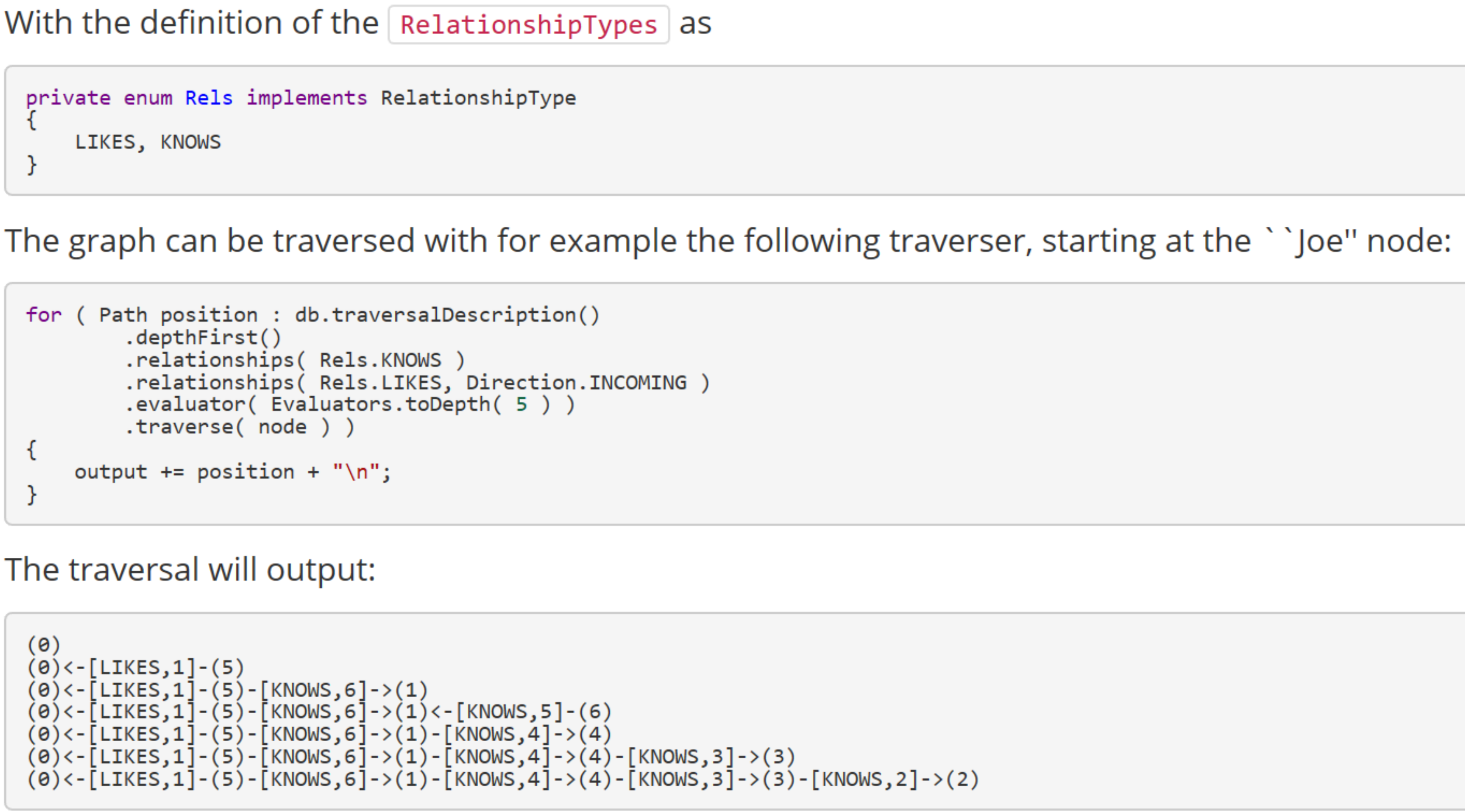
- Database operations that access the graph, indexes, or the schema must be performed in a transaction.

Memory Configuration



Traversal





With the definition of the `RelationshipTypes` as

```
private enum Rels implements RelationshipType
{
    LIKES, KNOWS
}
```

The graph can be traversed with for example the following traverser, starting at the ``Joe'' node:

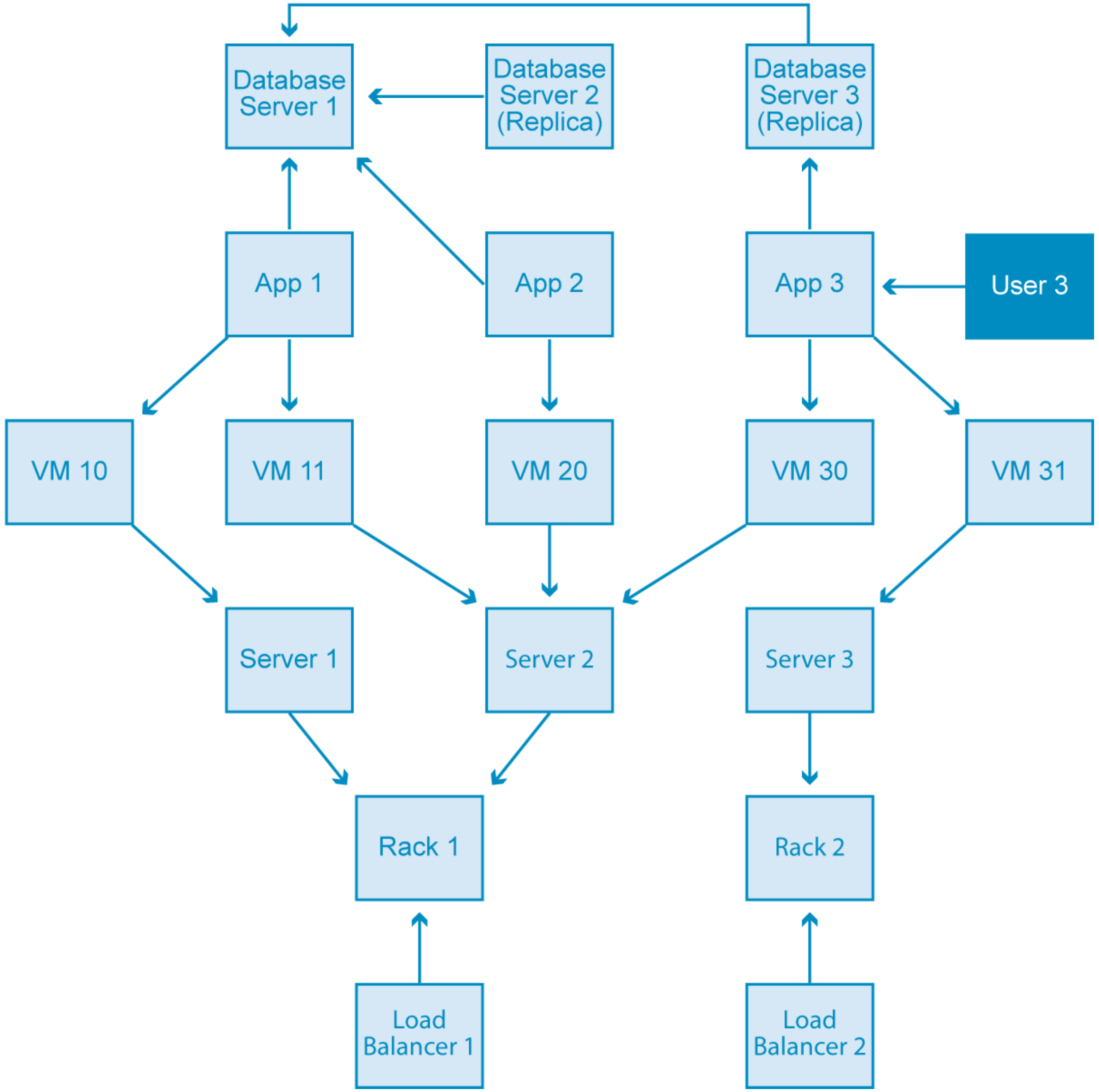
```
for ( Path position : db.traversalDescription()
        .depthFirst()
        .relationships( Rels.KNOWS )
        .relationships( Rels.LIKES, Direction.INCOMING )
        .evaluator( Evaluators.toDepth( 5 ) )
        .traverse( node ) )
{
    output += position + "\n";
}
```

The traversal will output:

```
(0)
(0)<-[LIKES,1]-(5)
(0)<-[LIKES,1]-(5)-[KNOWS,6]->(1)
(0)<-[LIKES,1]-(5)-[KNOWS,6]->(1)<-[KNOWS,5]-(6)
(0)<-[LIKES,1]-(5)-[KNOWS,6]->(1)-[KNOWS,4]->(4)
(0)<-[LIKES,1]-(5)-[KNOWS,6]->(1)-[KNOWS,4]->(4)-[KNOWS,3]->(3)
(0)<-[LIKES,1]-(5)-[KNOWS,6]->(1)-[KNOWS,4]->(4)-[KNOWS,3]->(3)-[KNOWS,2]->(2)
```

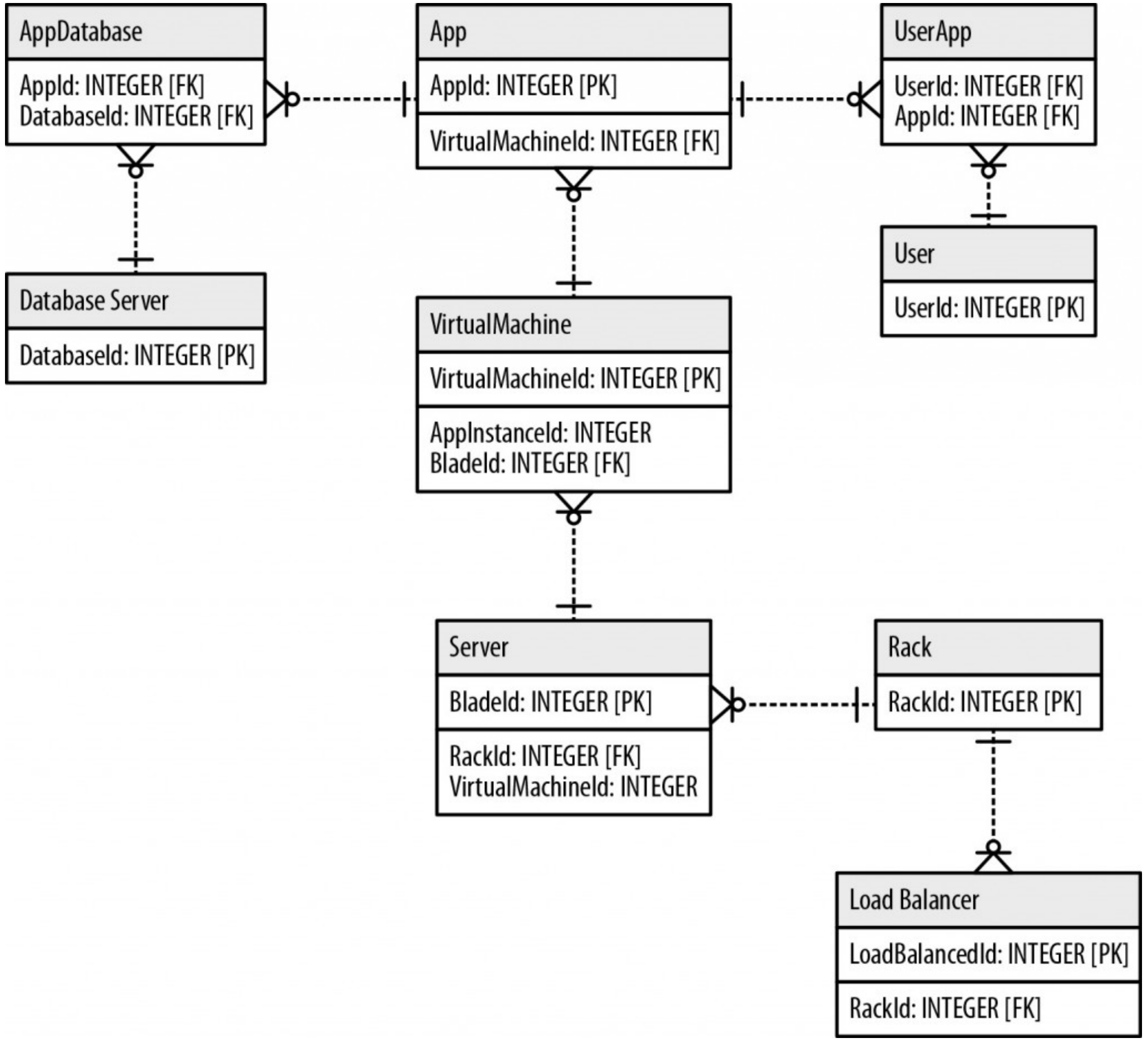
FLORIDA POLYTECHNIC UNIVERSITY

Data Modeling – Whiteboarding



FLORIDA POLYTECHNIC UNIVERSITY

Relational Model



AppDatabase

AppId: INTEGER [FK]
DatabaseId: INTEGER [FK]

App

AppId: INTEGER [PK]

VirtualMachineId: INTEGER [FK]

UserApp

UserId: INTEGER [FK]
AppId: INTEGER [FK]

Database Server

DatabaseId: INTEGER [PK]

User

UserId: INTEGER [PK]

VirtualMachine

VirtualMachineId: INTEGER [PK]

AppInstanceId: INTEGER
BladeId: INTEGER [FK]

Server

BladeId: INTEGER [PK]

RackId: INTEGER [FK]
VirtualMachineId: INTEGER

Rack

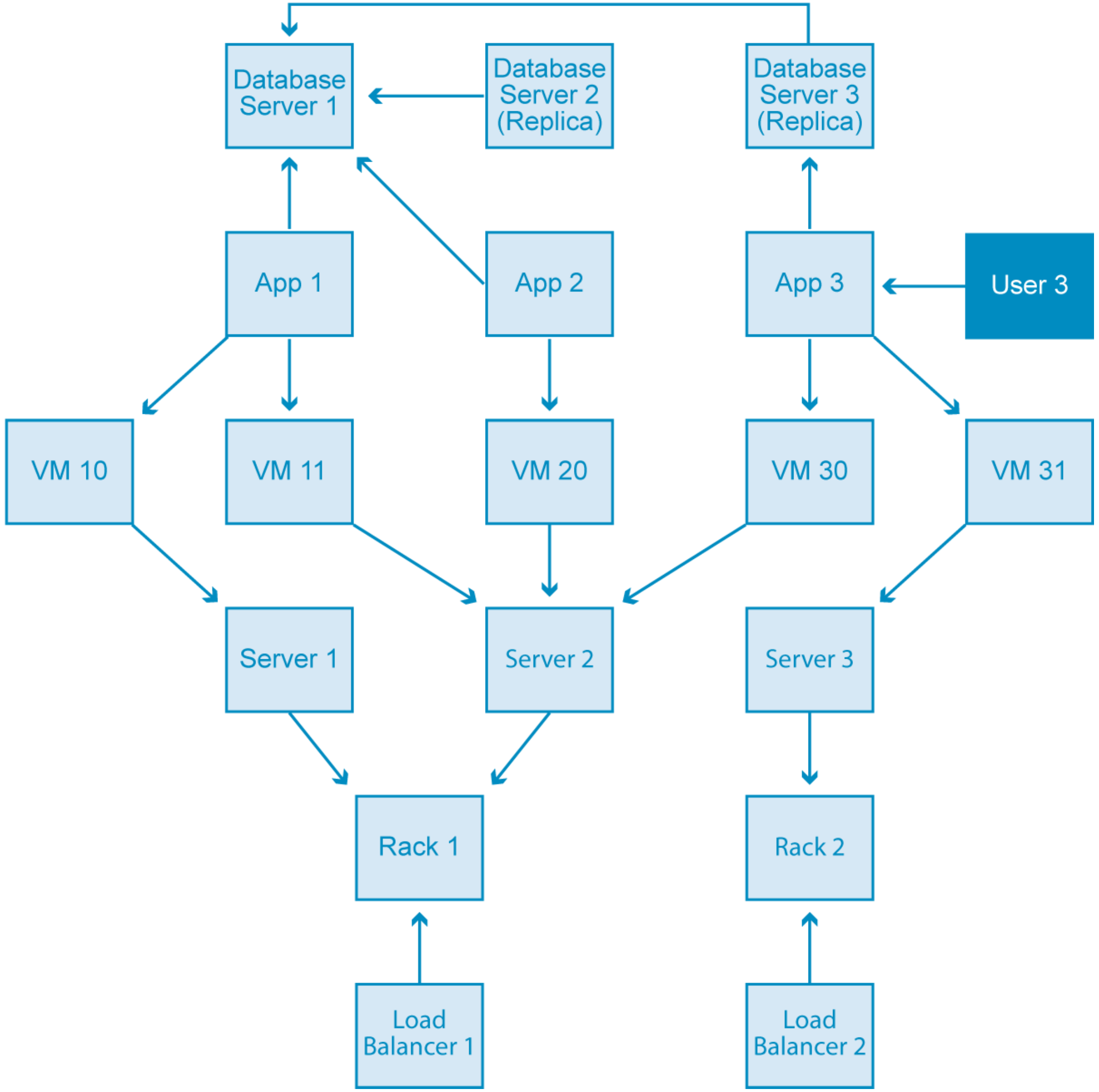
RackId: INTEGER [PK]

Load Balancer

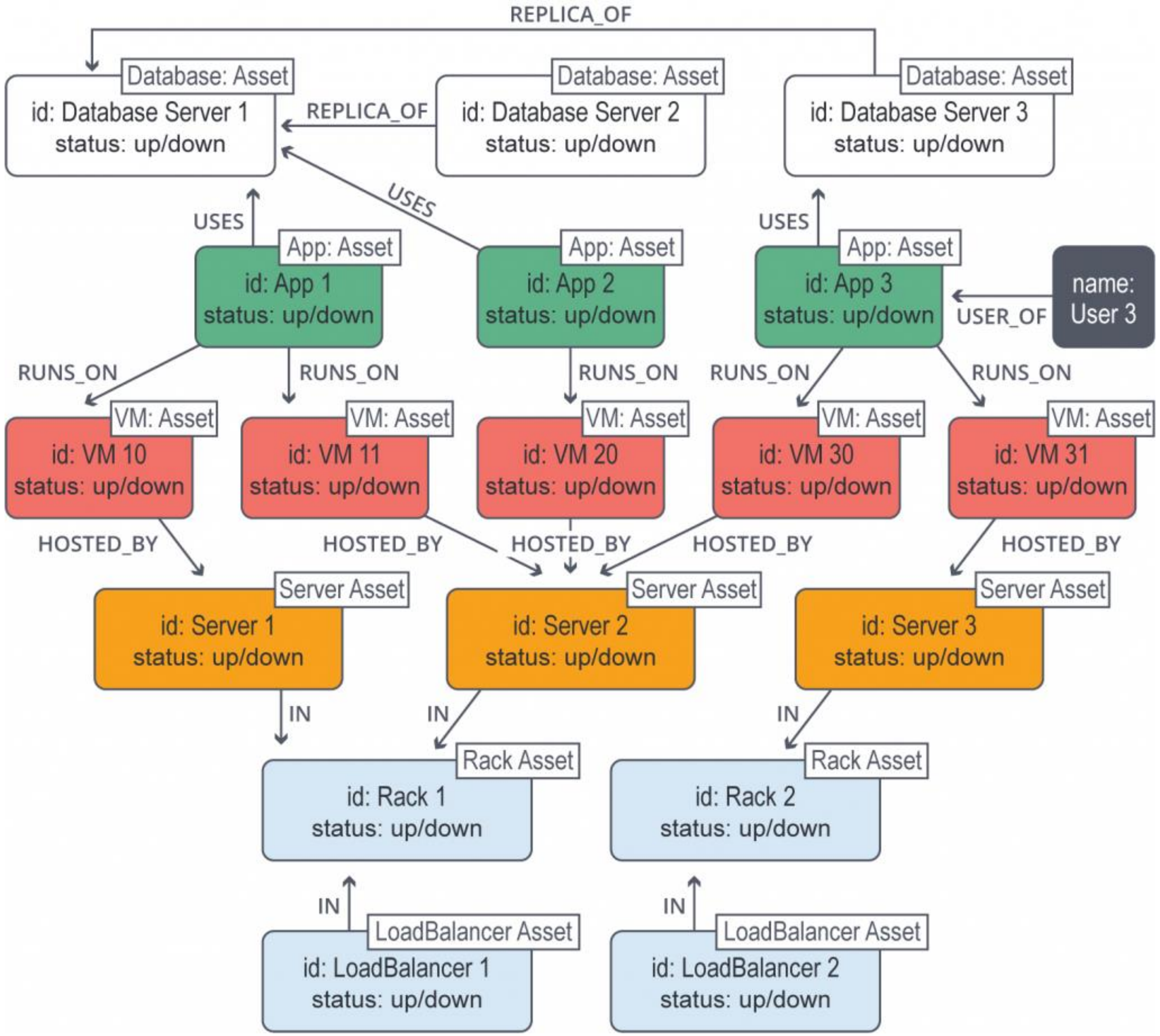
LoadBalancedId: INTEGER [PK]

RackId: INTEGER [FK]

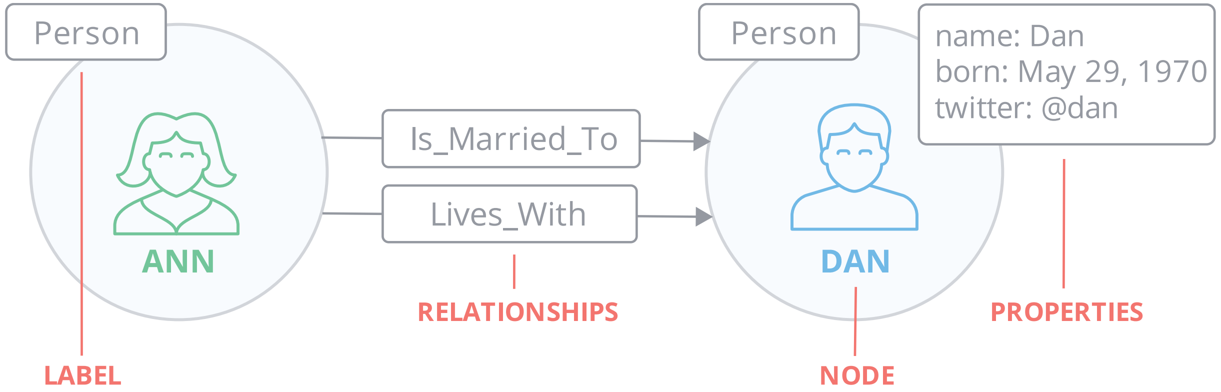
Graph Model


Database Server 1
Database Server 2 (Replica)
Database Server 3 (Replica)
App 1
App 2
App 3
User 3
VM 10
VM 11
VM 20
VM 30
VM 31
Server 1
Server 2
Server 3
Rack 1
Rack 2
Load Balancer 1
Load Balancer 2

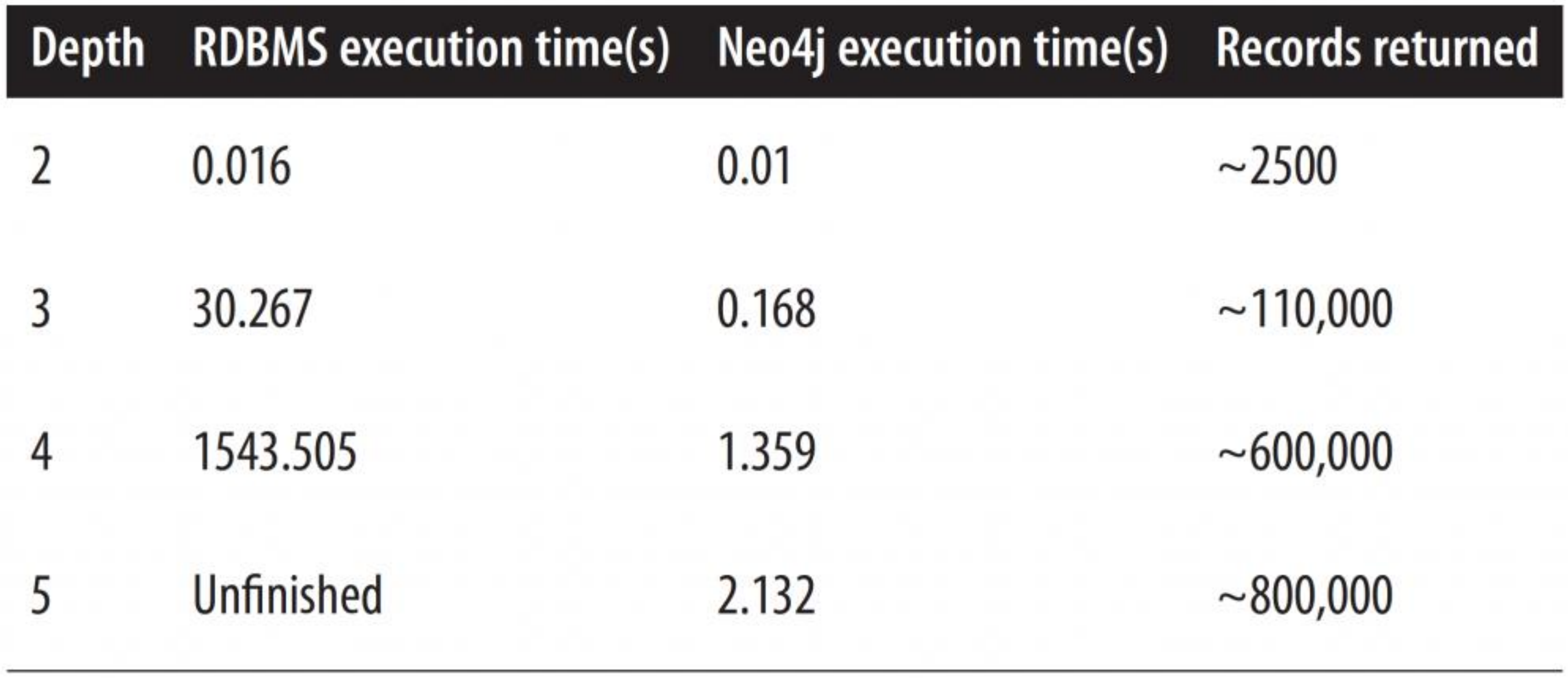
Enriched Model



Property Graph Model



Performance



Depth	RDBMS execution time(s)	Neo4j execution time(s)	Records returned
2	0.016	0.01	~2500
3	30.267	0.168	~110,000
4	1543.505	1.359	~600,000
5	Unfinished	2.132	~800,000

Intelligent Commerce for eBay App on Google Assistant

- Knowledge graph (Neo4j)
- Natural language understanding
- Artificial intelligence
- Store, remember and learn from past interactions with shoppers.
- eBay chose Neo4j as the native graph database that holds the probabilistic models that aid understanding in the conversational shopping scenario.
- The Neo4j graph contains both the product catalog and the attributes of shopper interactions while seeking products.

Examples

- https://neo4j.com/graphgist/degrees-offered-by-the-university-of-oviedo
- https://neo4j.com/graphgist/fitness-and-nutritional-recommendations
- https://neo4j.com/graphgist/credit-card-fraud-detection

FLORIDA POLYTECHNIC UNIVERSITY