In-Memory Computing Essentials

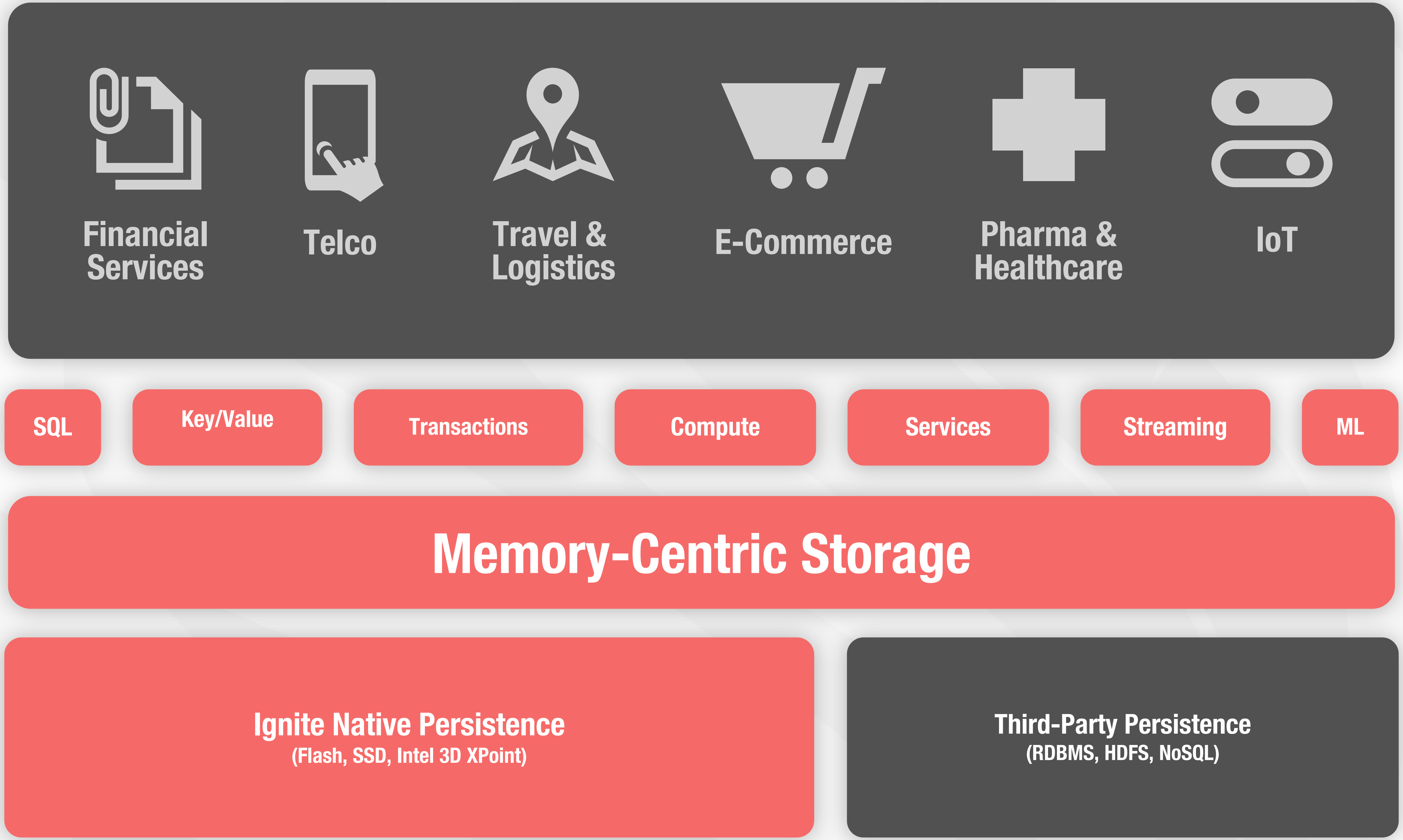
for Architects and Developers: Part 1

Denis Magda
Ignite PMC Chair
GridGain Director of Product Management

Agenda

- Apache Ignite Overview
- Clustering and Deployment
- Distributed Storage
- Distributed SQL
- Q&A

Apache Ignite In-Memory Computing Platform



Financial Services

Telco

Travel & Logistics

E-Commerce

Pharma & Healthcare

IoT

SQL

Key/Value

Transactions

Compute

Services

Streaming

ML

Memory-Centric Storage

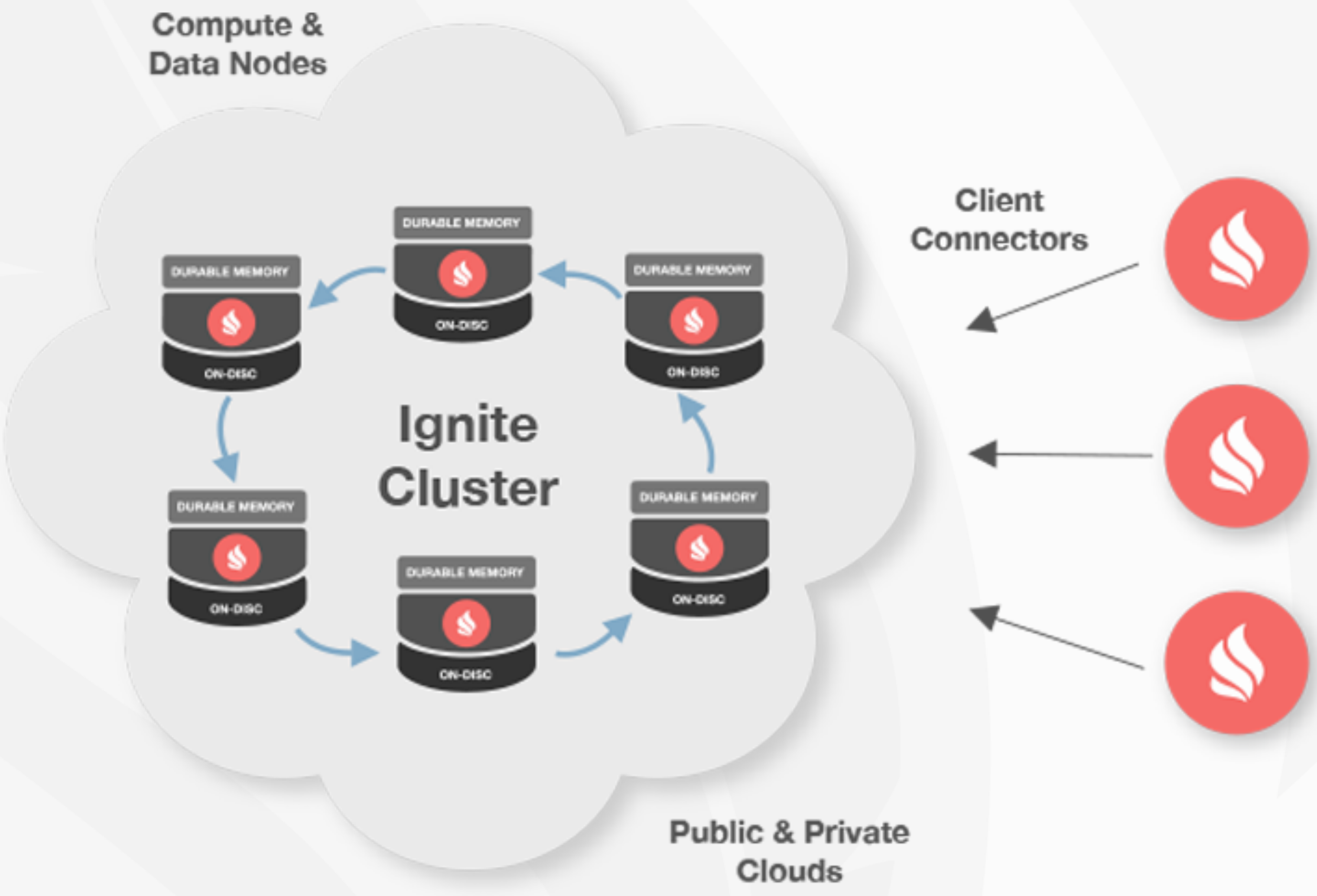
Ignite Native Persistence
(Flash, SSD, Intel 3D XPoint)

Third-Party Persistence
(RDBMS, HDFS, NoSQL)

Clustering and Deployment

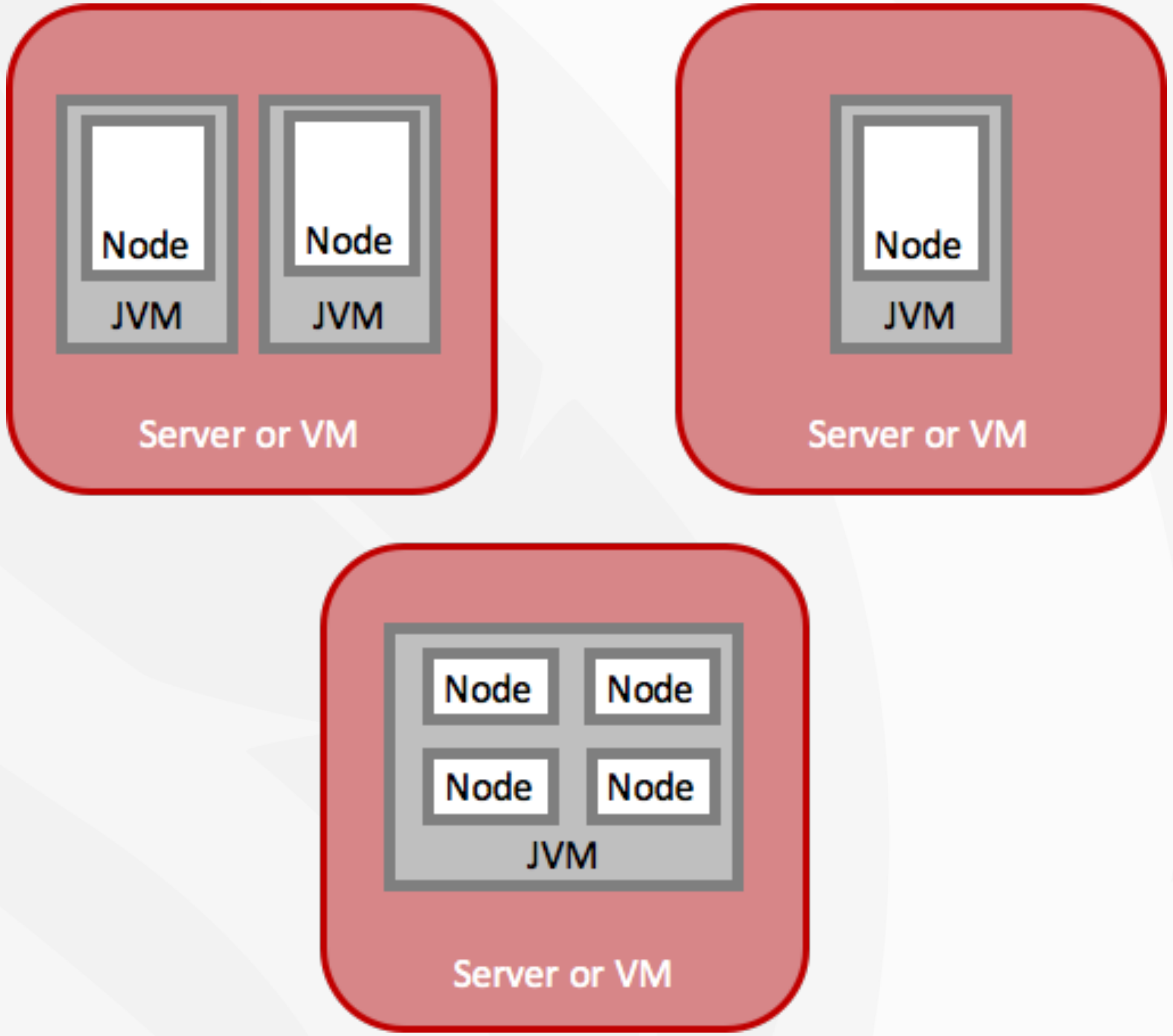
Clustering

- Server Nodes
  - Act as containers for data and computations
  - Generally started as standalone processes
- Client Nodes
  - Provide a cluster entry point to run operations
  - Embedded in applications code



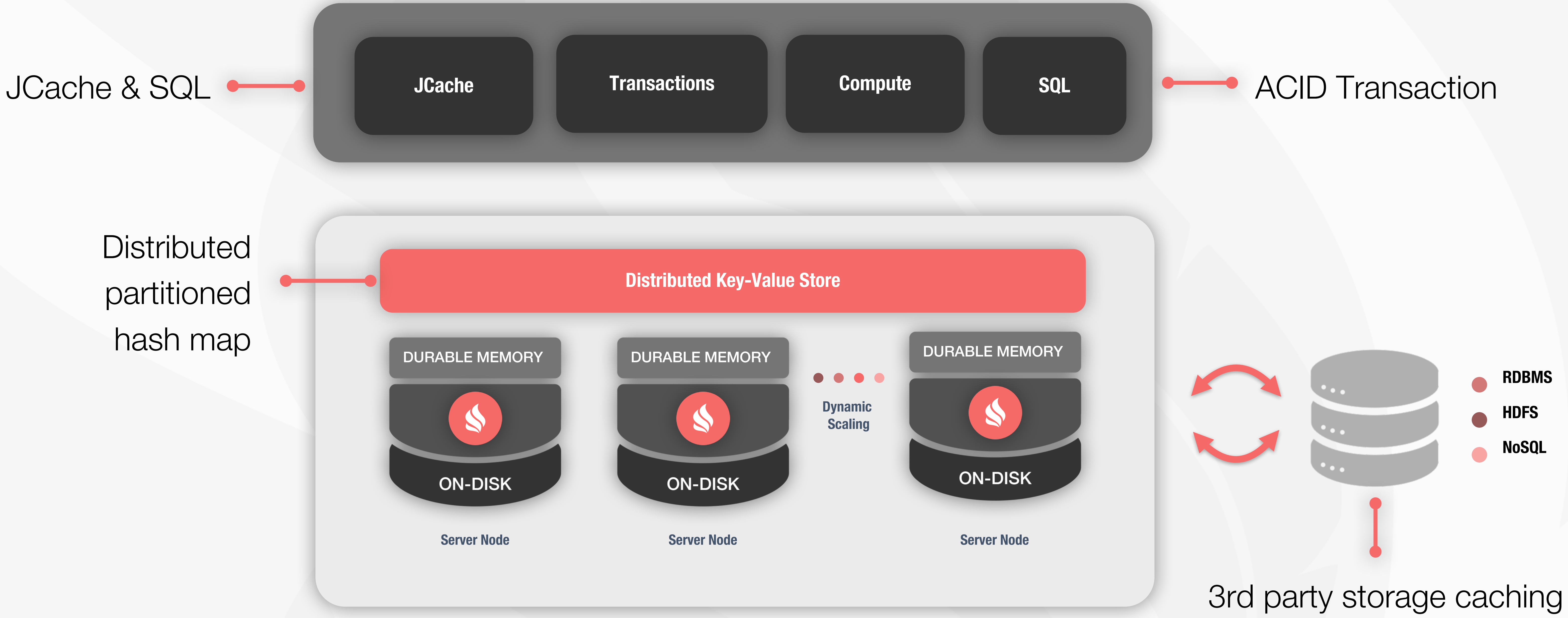
Deployment

- Nodes are logical entities
  - Runs in a JVM process
  - Many nodes in a single JVM process
- On-Premise and Cloud
  - Physical server or VM
  - AWS, Azure, Google Compute Engine
- Kubernetes, Mesos, YARN

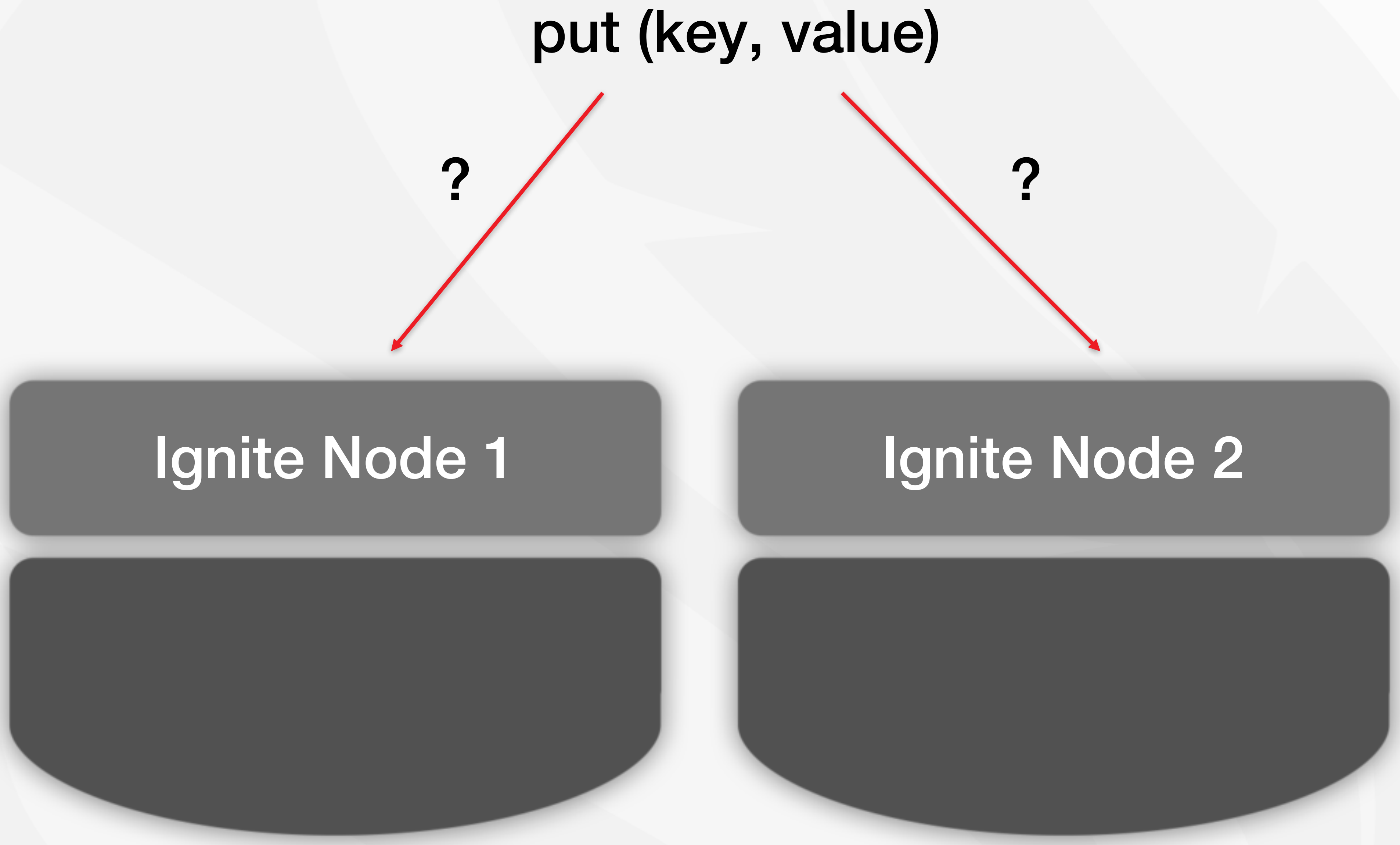


Distributed Storage

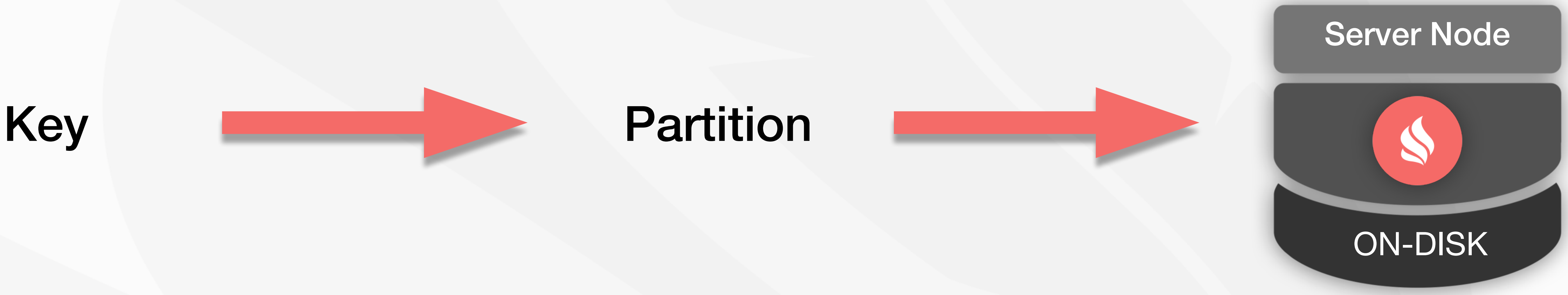
Distributed Storage



Where Entry Goes?



Key to Node Mapping



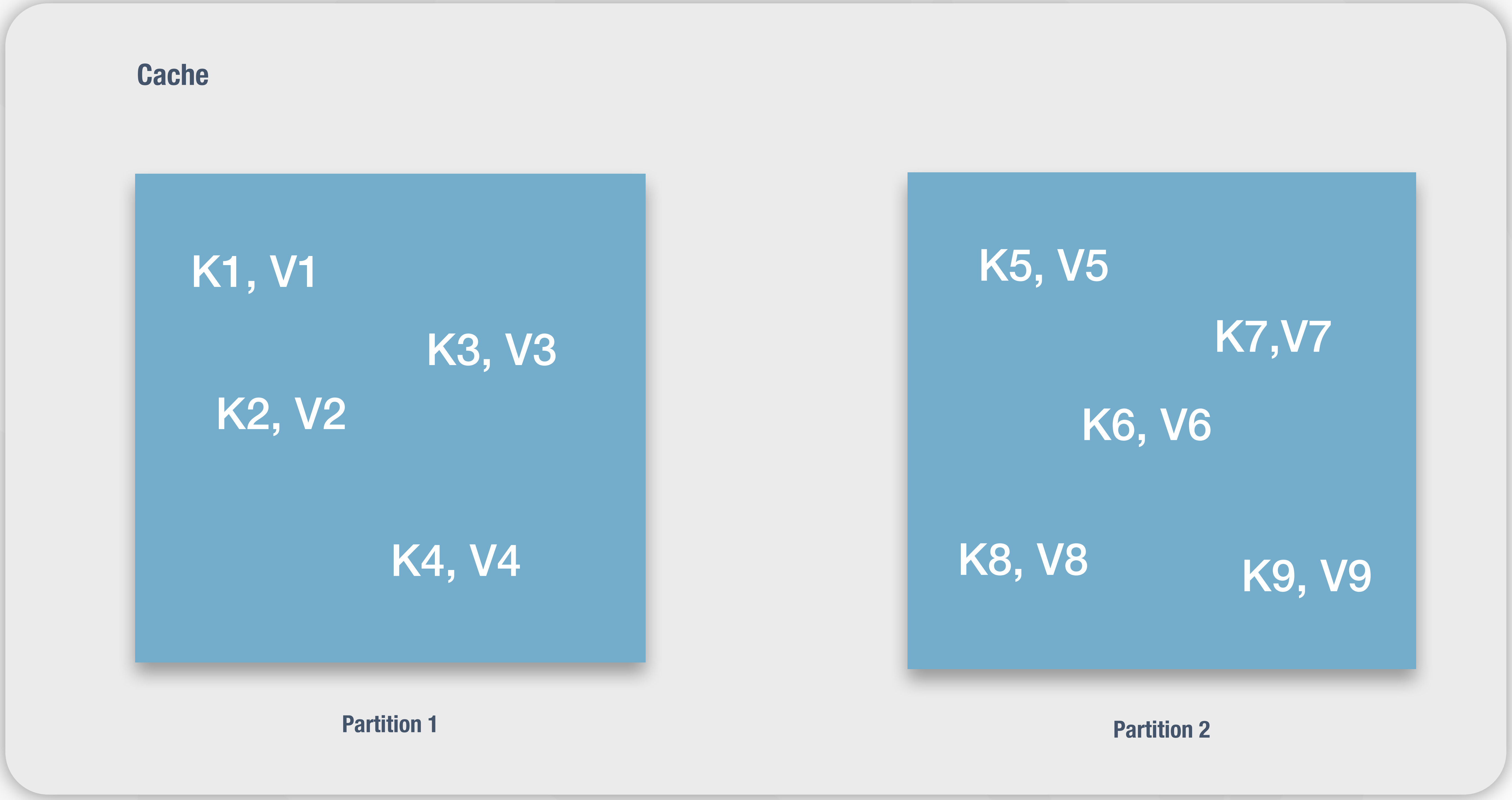
Key

Partition

Server Node

ON-DISK

Caches and Partitions



Cache

K1, V1

K3, V3

K2, V2

K4, V4

Partition 1

K5, V5

K7,V7

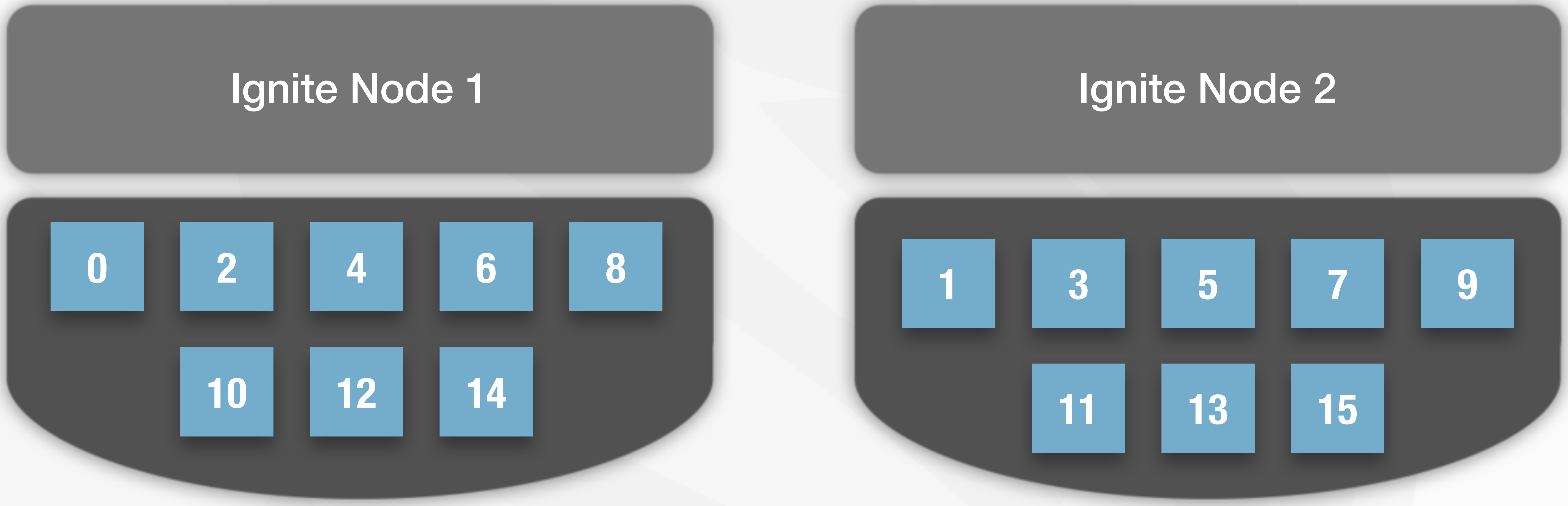
K6, V6

K8, V8

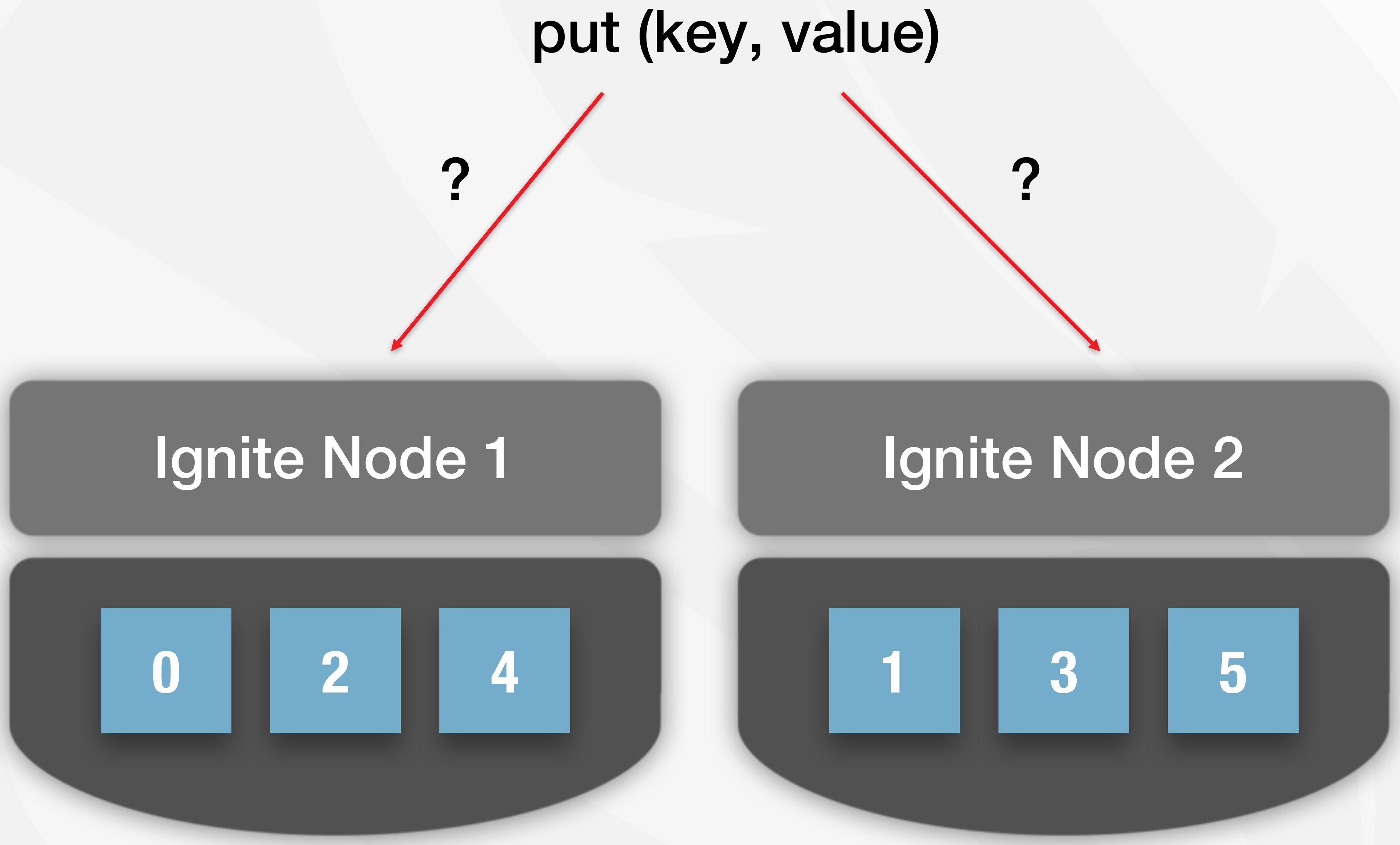
K9, V9

Partition 2

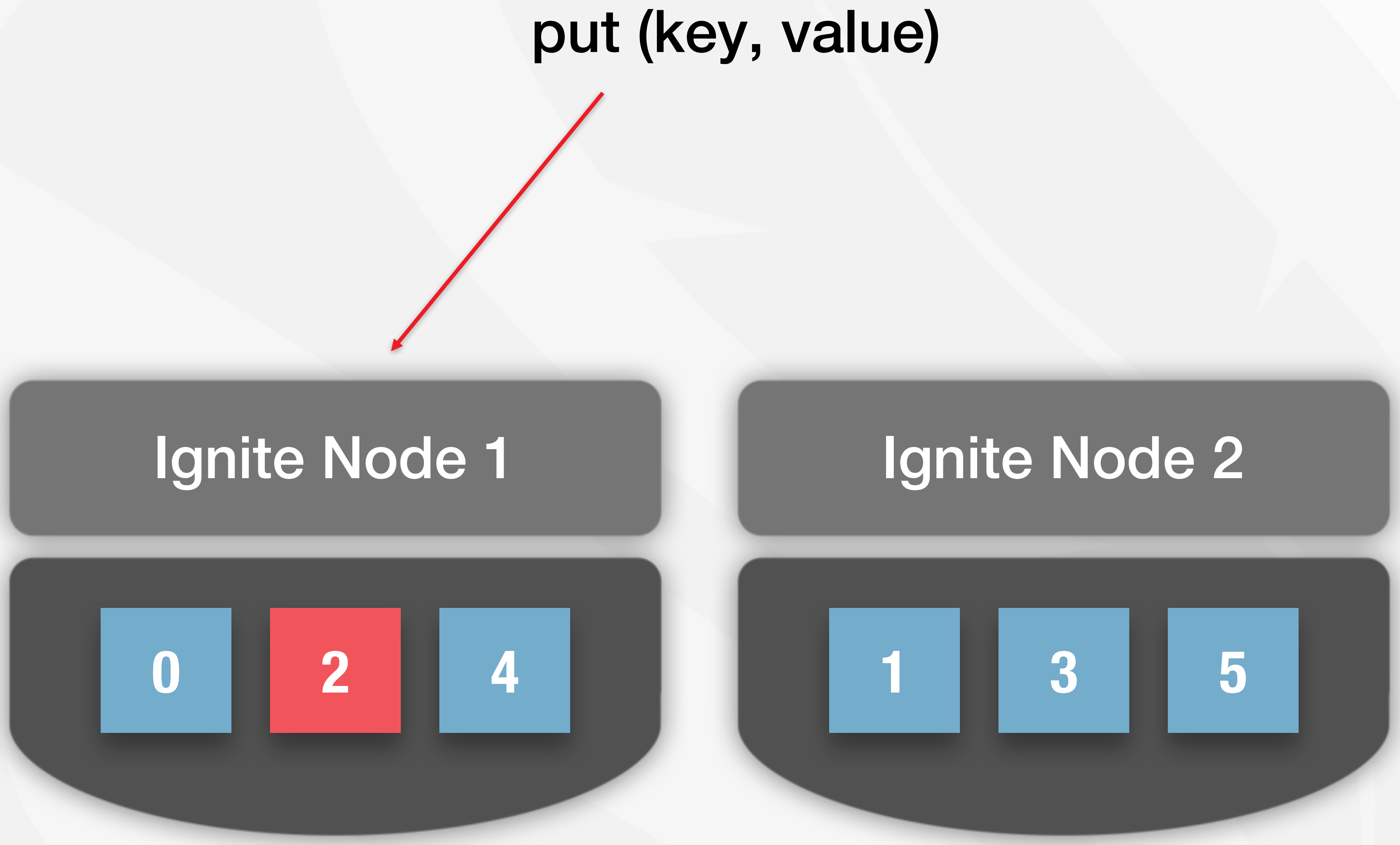
Partitions Distribution



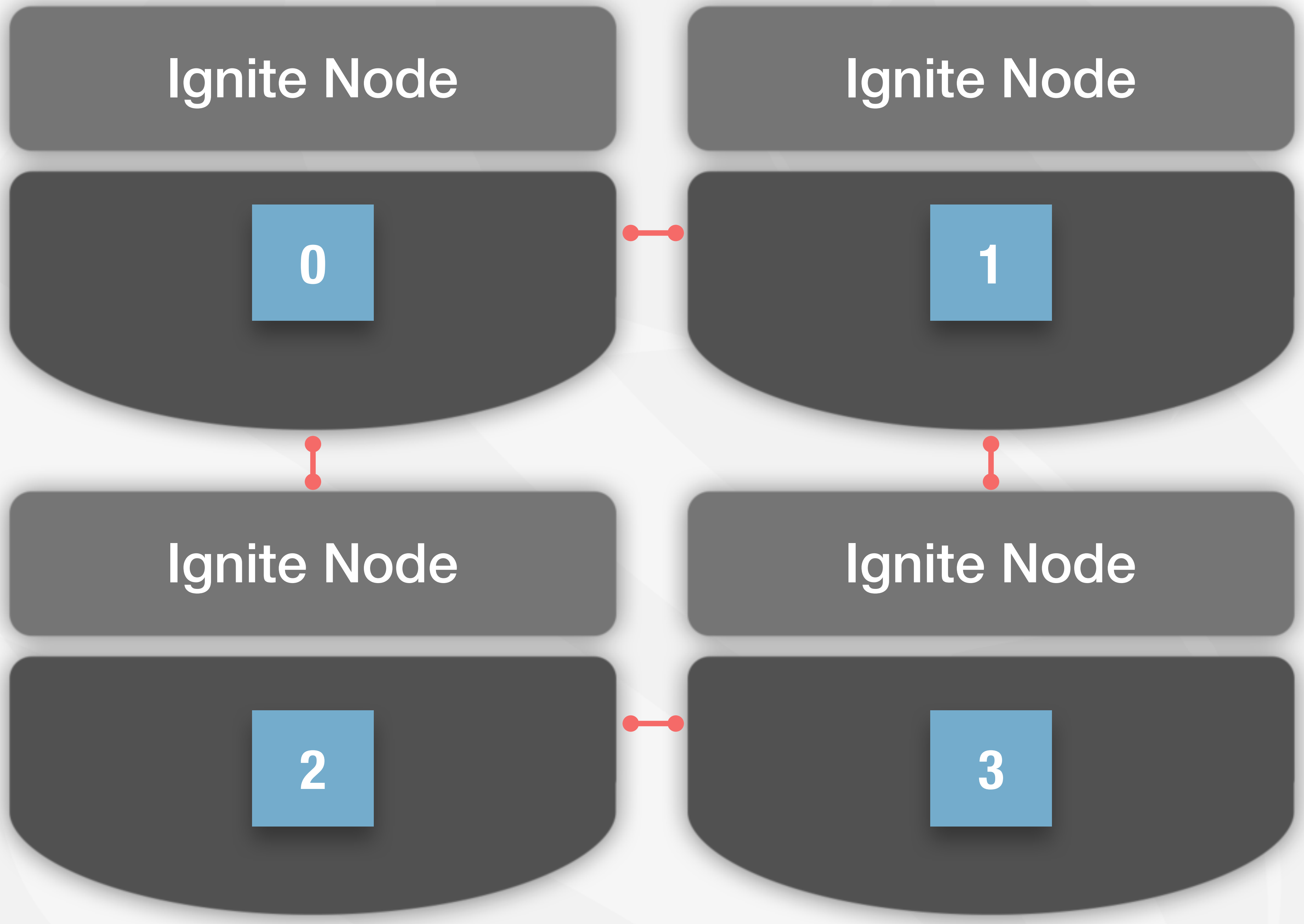
Where Entry Goes?



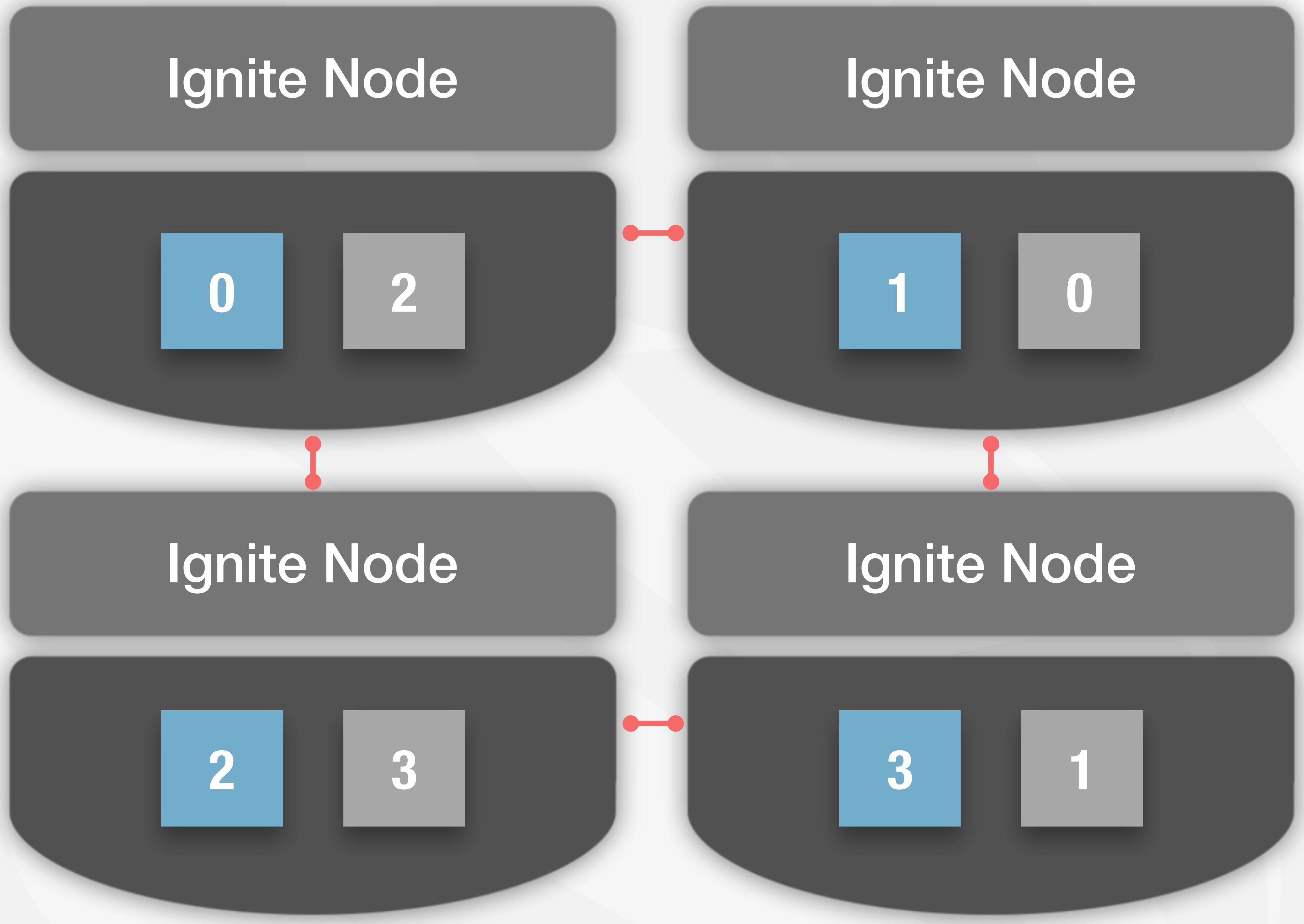
Where Entry Goes?



Backup Copies



Backup Copies



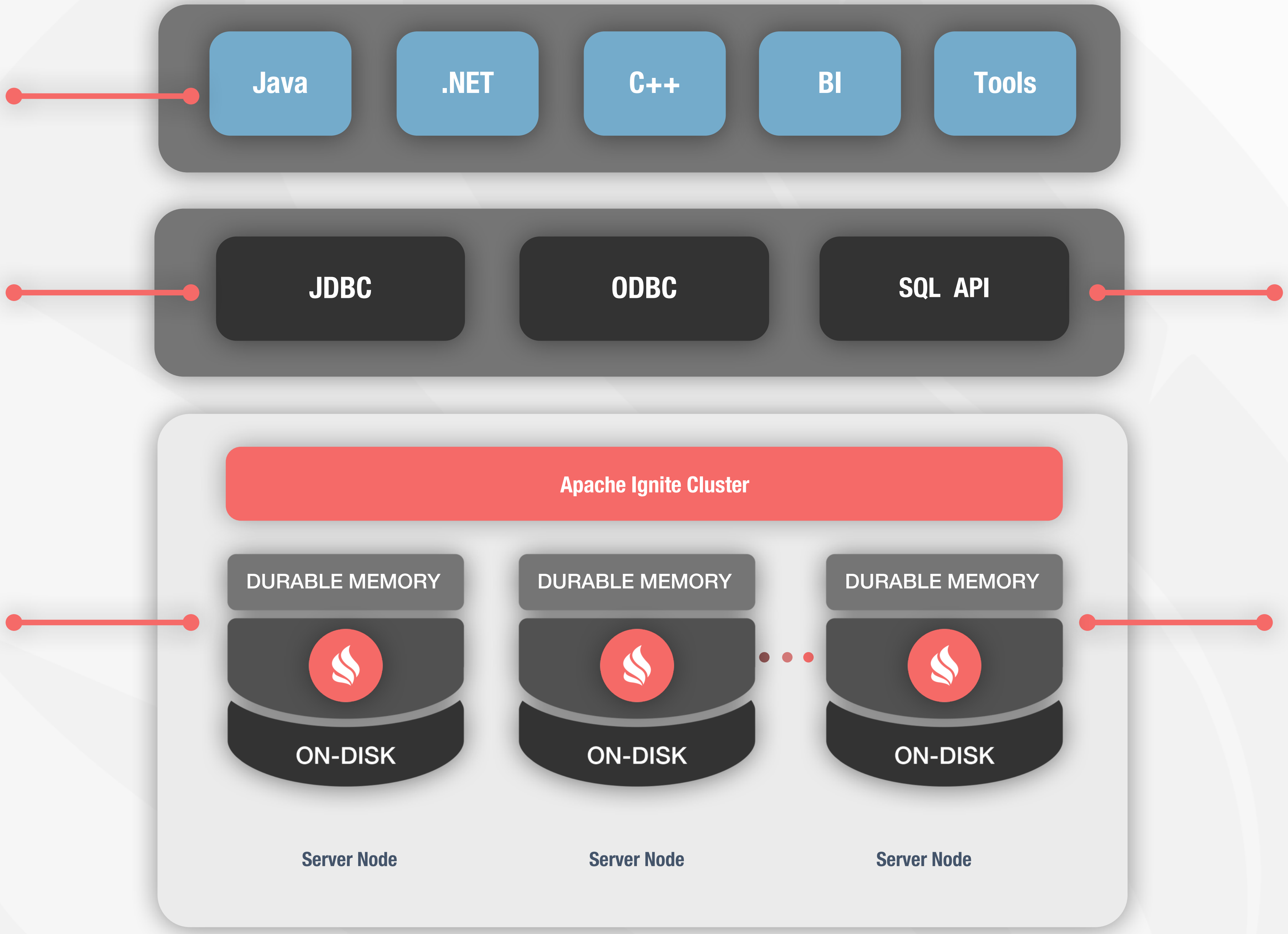
Distributed SQL

Distributed SQL

Cross-platform Compatibility

DDL, DML Support

Indexes in RAM or Disk



Java

.NET

C++

BI

Tools

JDBC

ODBC

SQL API

Apache Ignite Cluster

DURABLE MEMORY

ON-DISK

Server Node

DURABLE MEMORY

ON-DISK

Server Node

DURABLE MEMORY

ON-DISK

Server Node

SELECT, UPDATE, INSERT, MERGE, DELETE, CREATE and ALTER

Dynamic Scaling

Connectivity

- JDBC
- ODBC
- REST
- Java, .NET and C++ APIs

```
// Register JDBC driver.
Class.forName("org.apache.ignite.IgniteJdbcThinDriver");

// Open the JDBC connection.
Connection conn = DriverManager.getConnection("jdbc:ignite:thin://192.168.0.50");
```

```
./sqlline.sh --color=true --verbose=true -u jdbc:ignite:thin://127.0.0.1/
```

Data Definition Language

- CREATE/DROP TABLE
- CREATE/DROP INDEX
- ALTER TABLE
- Changes Durability
    - Ignite Native Persistence

```sql
CREATE TABLE `city` (
  `ID` INT(11),
  `Name` CHAR(35),
  `CountryCode` CHAR(3),
  `District` CHAR(20),
  `Population` INT(11),
  PRIMARY KEY (`ID`, `CountryCode`)
) WITH "template=partitioned, backups=1, affinityKey=CountryCode";
```

Data Manipulation Language

- ANSI-99 specification
- Fault-tolerant and consistent
- INSERT, UPDATE, DELETE
- SELECT
  - JOINs
  - Subqueries

```
SELECT country.name, city.name, MAX(city.population) as max_pop
FROM country JOIN city ON city.countrycode = country.code
WHERE country.code IN ('USA','RUS','CHN')
GROUP BY country.name, city.name ORDER BY max_pop DESC LIMIT 3;
```

IMC Essentials Part 2

- Affinity Collocation
- Collocated Processing and Distributed Computations
- Collocated Processing and SQL
- Machine Learning
- Memory Architecture and Persistence

Wednesday, December 13, 2017, 11:00am PT / 2:00pm ET

https://www.gridgain.com/resources/webinars/in-memory-computing-essentials-architects-and-developers-part-2

Any Questions?

Thank you for joining us. Follow the conversation.
http://ignite.apache.org

#apacheignite
#denismagda