



# MEMORY-CENTRIC ARCHITECTURE

*IN-MEMORY PERFORMANCE, DURABILITY OF DISK*

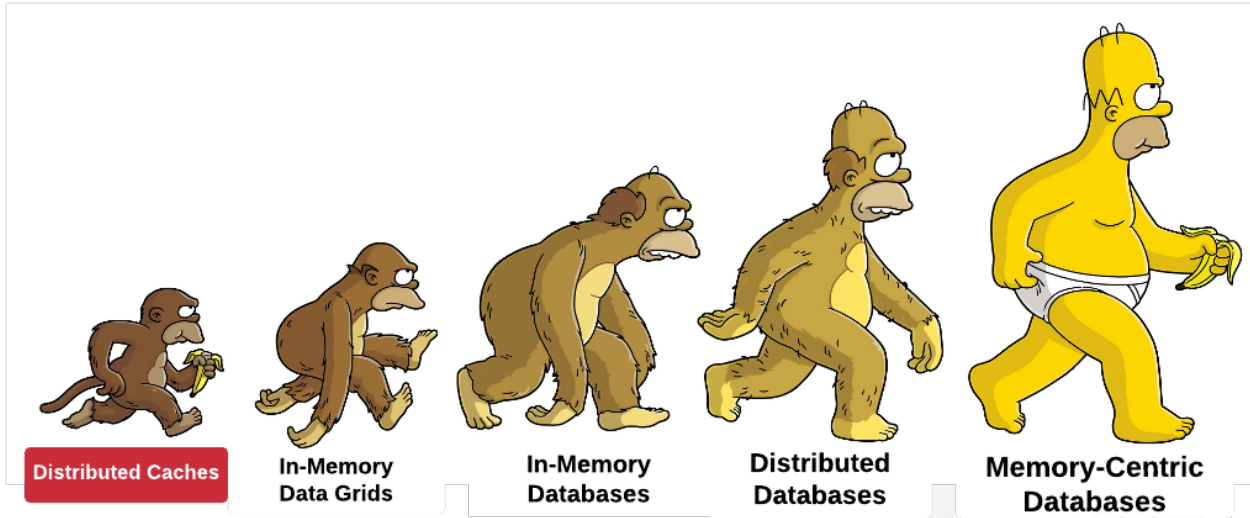
Dmitriy Setrakyan

*Chief Product Officer, Apache Ignite PMC*

# Agenda

- In-Memory Computing Evolution
  - Distributed Caches
  - In-Memory Data Grids
  - In-Memory Databases
  - Distributed Databases
  - Memory-Centric Databases
- Data Storage Ecosystem

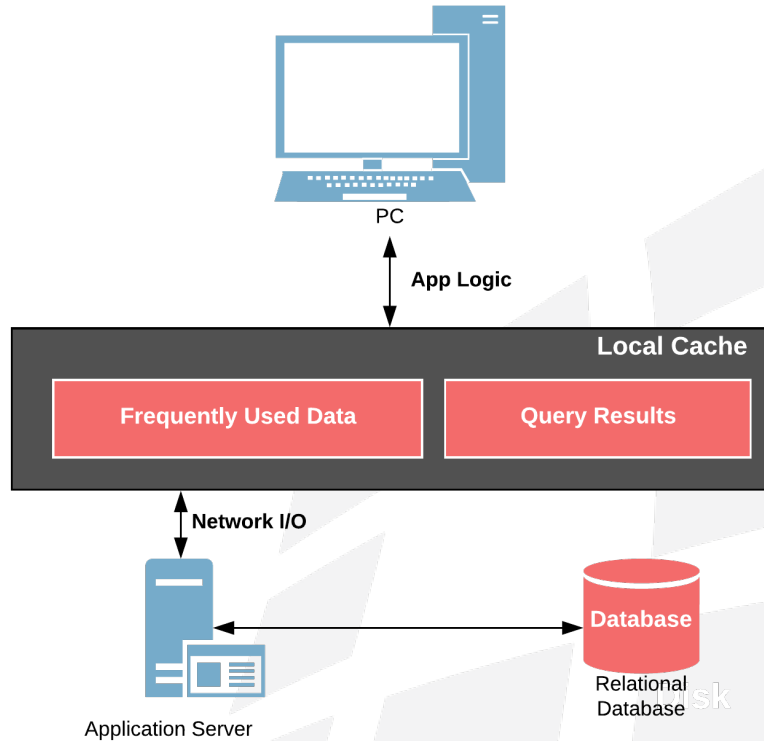
# Distributed Caches



**Cache** is a hardware or software component that stores data so future requests for that data can be served faster

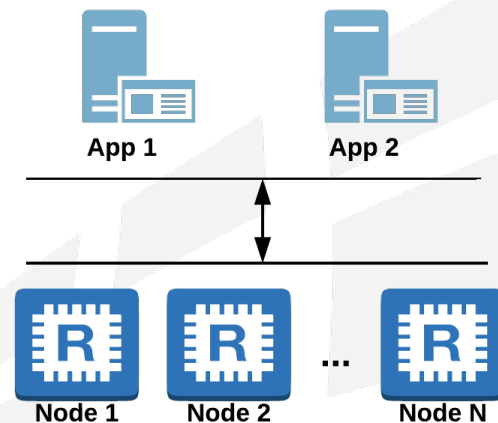
# Local Caching

- Application In-Process Caching
  - Querying Results
  - Most Frequently Used Data
- Browser Caching
- Benefits
  - Speed up Applications!
  - Network is slow!



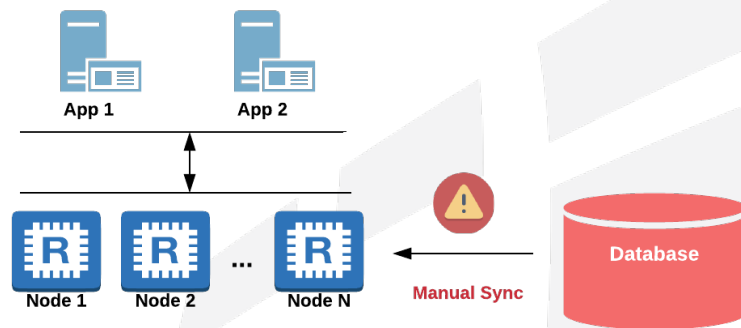
# Distributed Caches

- Share-nothing (consistent hashing or sharding)
- Simple client protocol
- Memcached, Redis, AWS ElastiCache
- Benefits
  - Shared cache
  - Beyond local RAM capacity
  - Fault tolerant and scalable

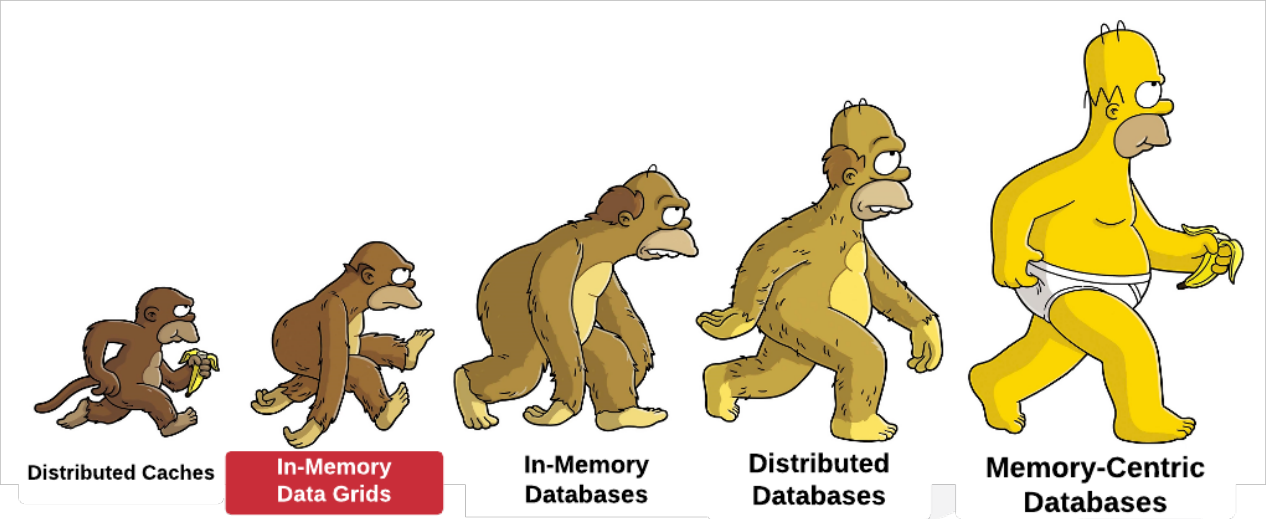


# Shortcomings of Distributed Caches

- Lack of queries (SQL or JOINS)
- Lack of ACID transactions
- Lack of collocated processing
- Lack of database synchronization
  - Cache-aside pattern
- Lack of native persistence
  - cannot persist more than fits in RAM
- RAM warm-up on restart



# In-Memory Data Grids



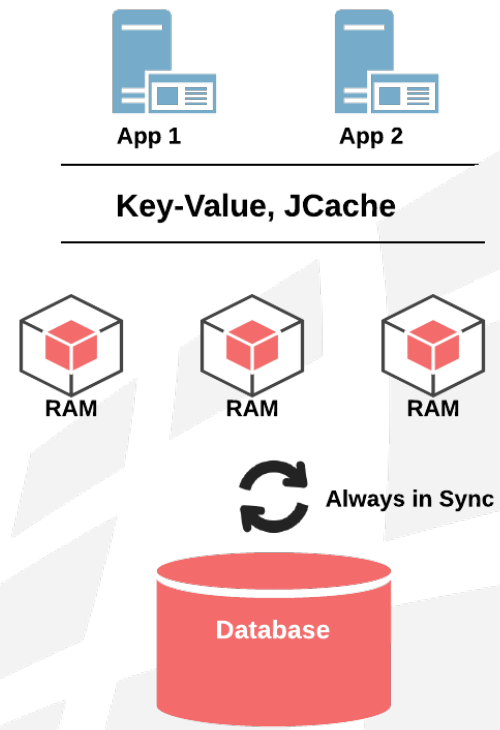


# In-Memory Data Grids

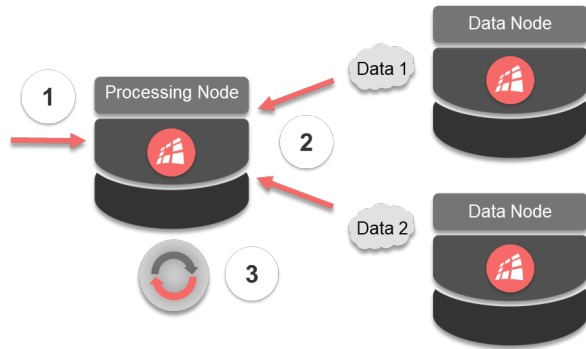
- Intelligent Distributed Caches
- Hazelcast, GigaSpaces, Apache Ignite

- Benefits

- ACID Transactions
- Query APIs
- Event notifications
- Continuous queries
- Database Read/Write-Through
- Collocated Processing

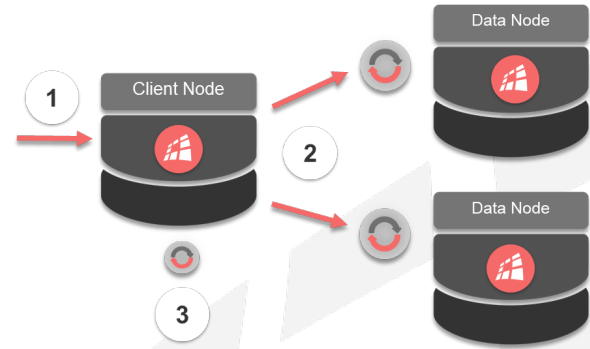


# Client-Server Processing



1. Initial Request
2. Fetch data from remote nodes
3. Process the entire data-set

# Co-Located Processing



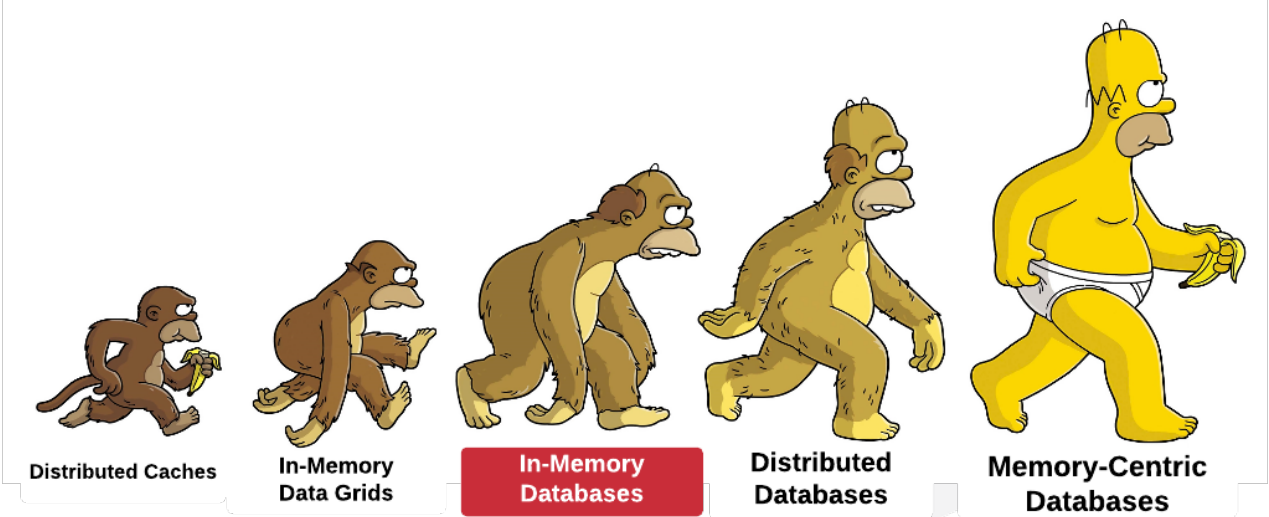
1. Initial request
2. Co-locate processing with data
3. Reduce multiple results into one

# Shortcomings of In-Memory Data Grids

- Custom Query Languages
  - No SQL, No JOINS
- Lack of native persistence
  - cannot persist more than fits in RAM
- RAM warm-up on restart



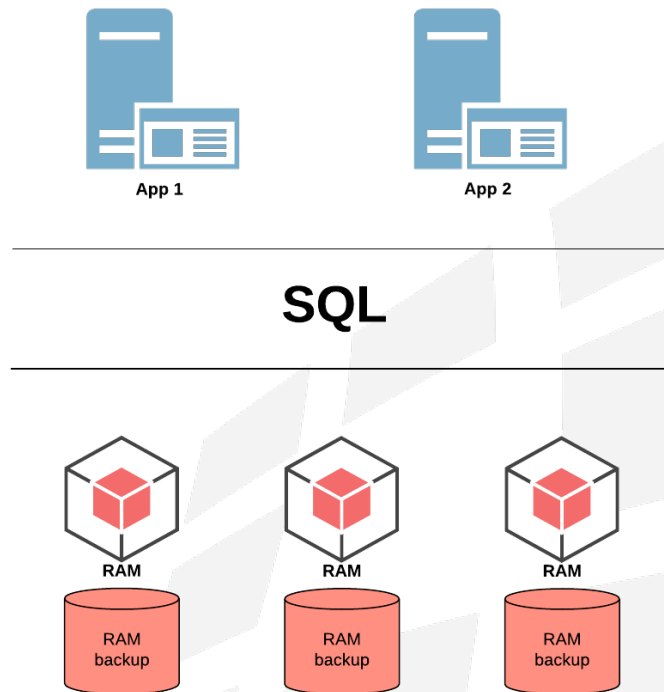
# In-Memory Databases



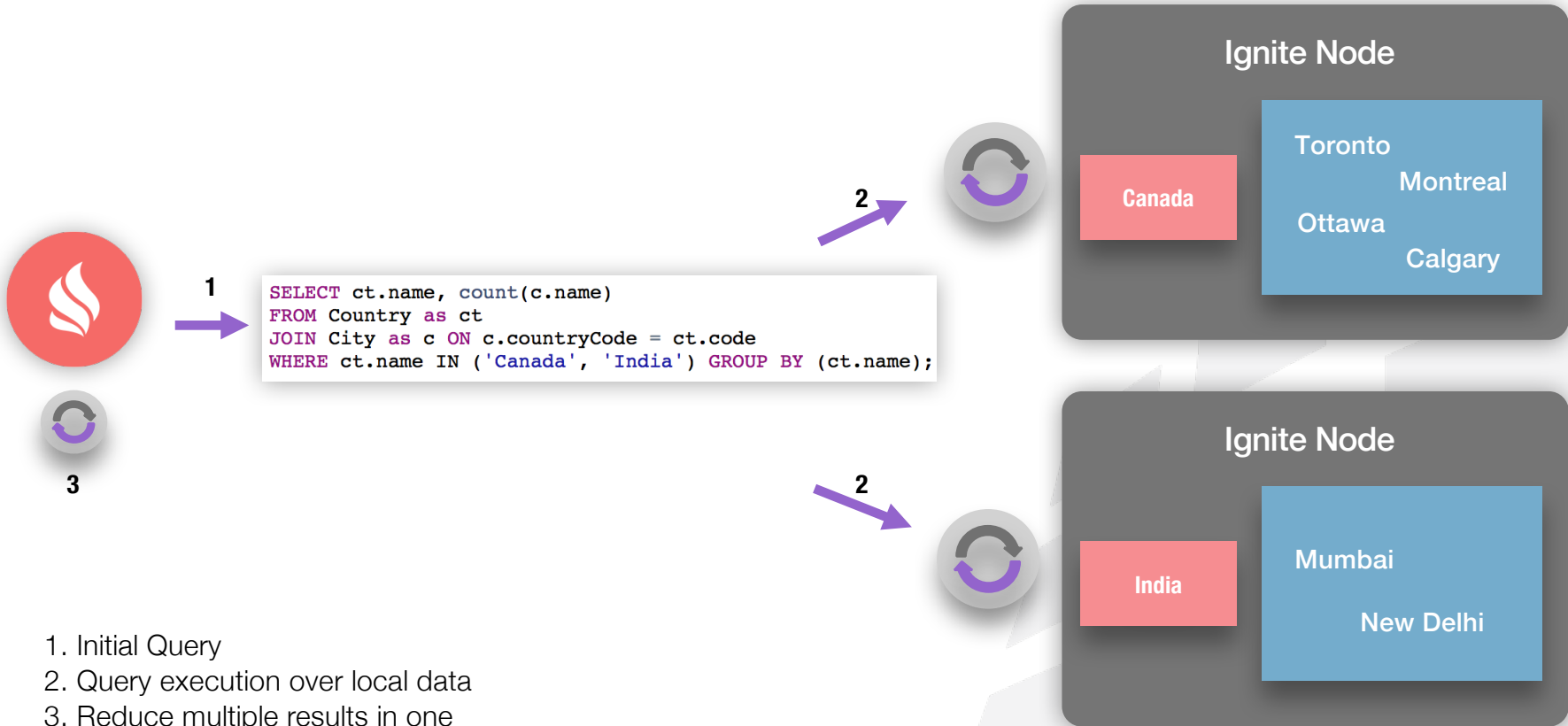
# In-Memory Databases

- Primary in-RAM Storage
- In-Memory Data Grids for SQL
- VoltDB, SAP Hana

- Benefits
  - Distributed SQL
  - ACID Transactions
  - Better Scalability



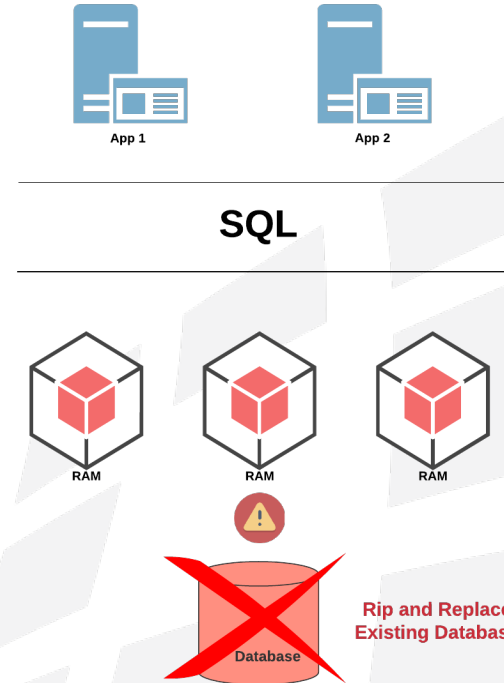
# Collocated Joins



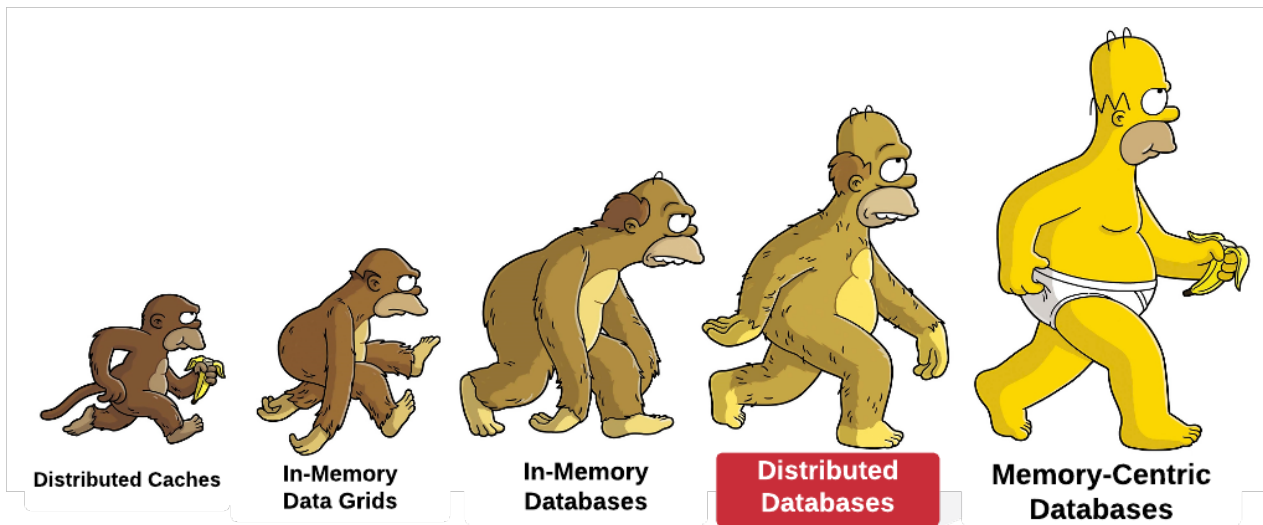
1. Initial Query
2. Query execution over local data
3. Reduce multiple results in one

# Shortcomings of In-Memory Databases

- Rip-n-Replace existing database
  - cannot keep or reuse existing database
- Lack of collocated processing
- Lack of native persistence
  - cannot persist more than fits in RAM
- RAM warm-up on restart



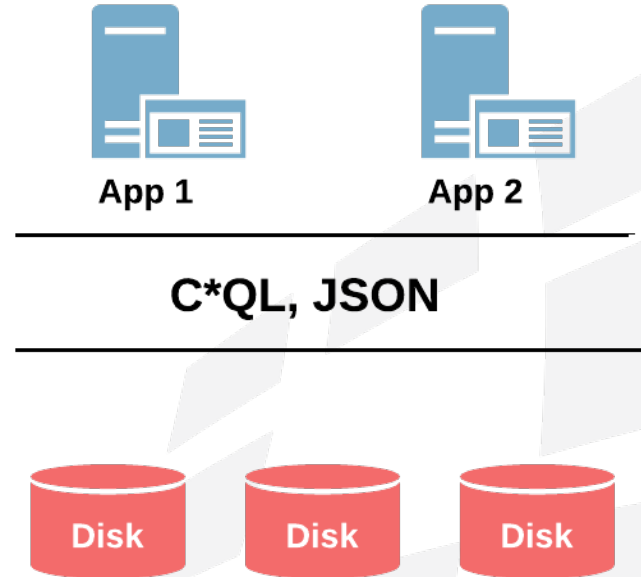
# Distributed Databases





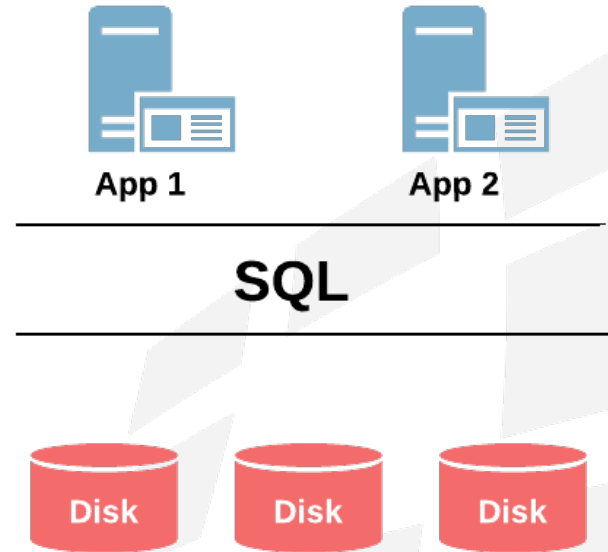
# Distributed NoSQL Databases

- Cassandra, Mongo
  - Distributed persistence
  - Fault tolerant and scalable
- Shortcomings
  - Lack of SQL (no JOINS)
  - Lack of distributed transactions
  - Lack of in-memory processing

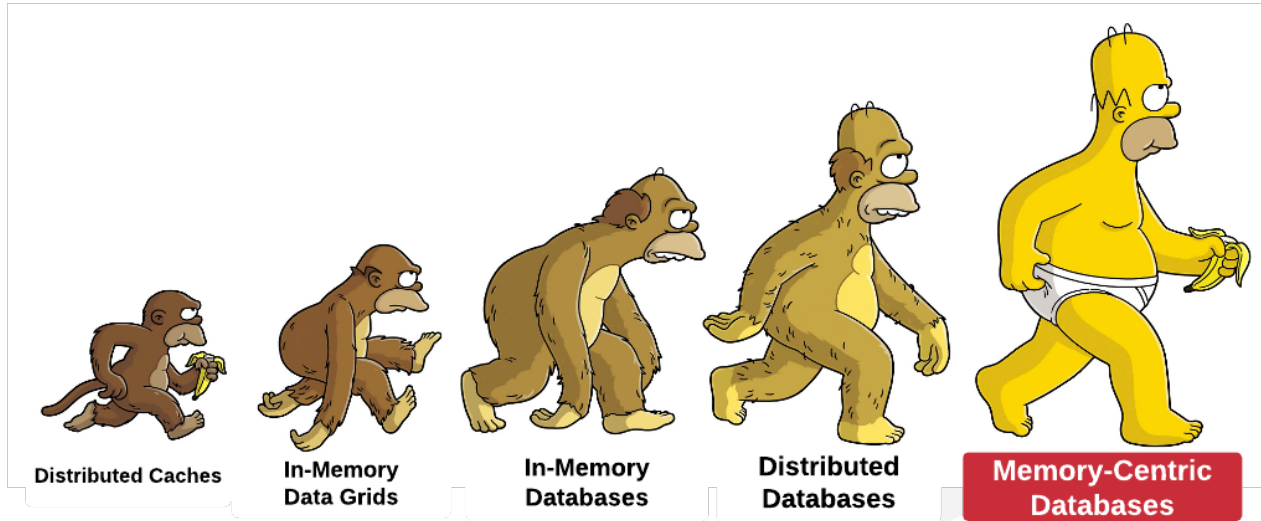


# Distributed SQL Databases

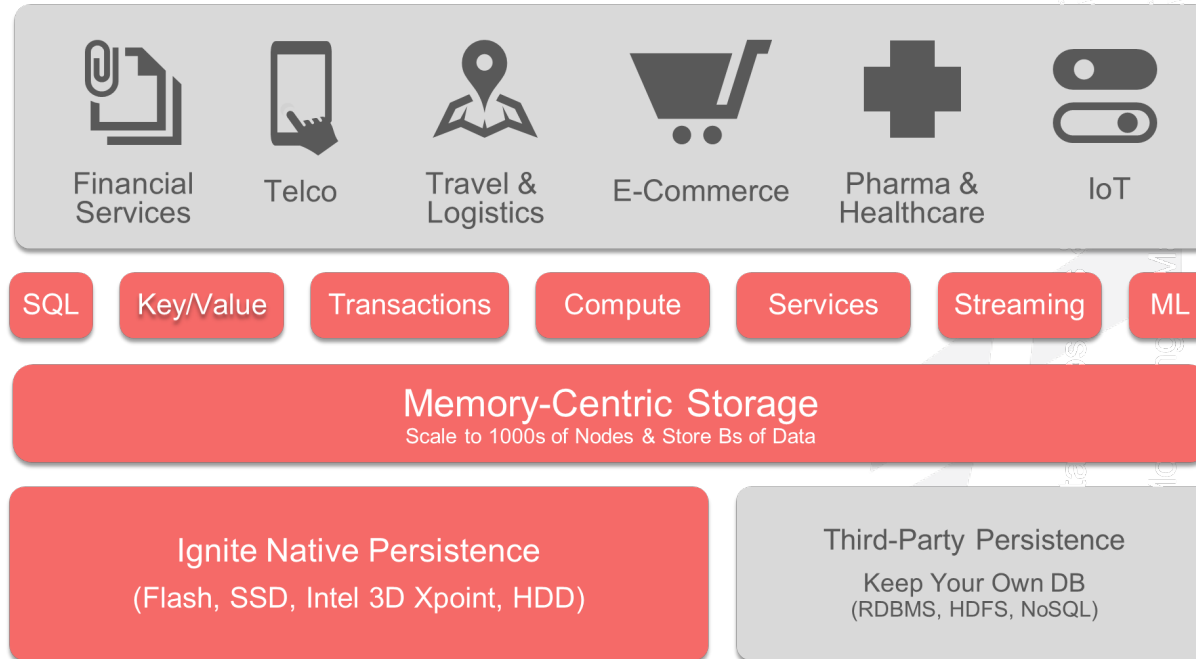
- CockroachDB, Cloud Spanner
  - Cloud-native
  - Fault tolerant and scalable
  - SQL (distributed JOINS)
- Shortcomings
  - Lack of in-memory processing
  - Lack of key-value API
  - Lack of event notifications



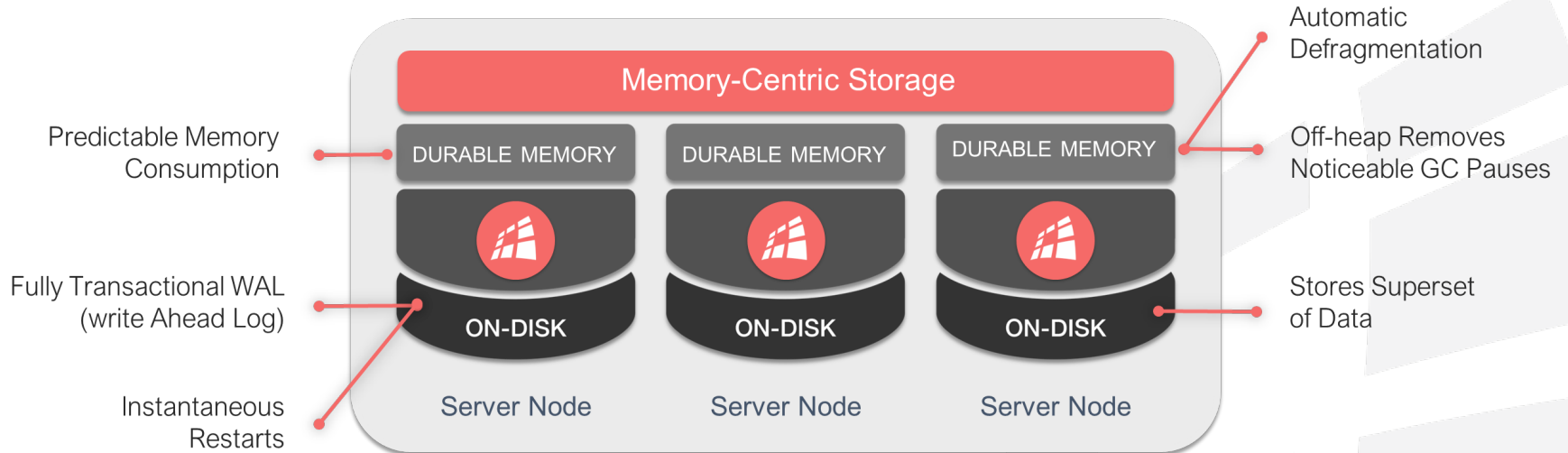
# Memory-Centric Databases



# GridGain Memory-Centric Architecture



# Memory Centric Storage

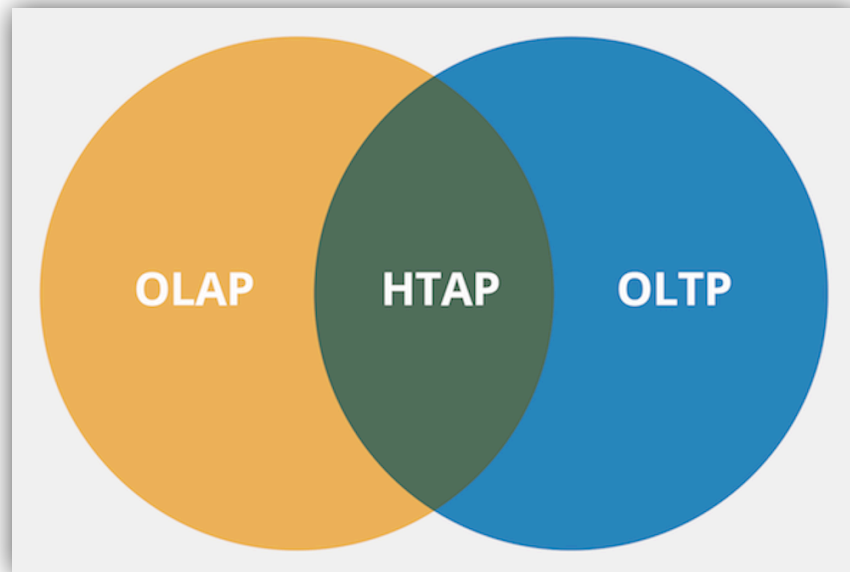


# Memory & Disk Utilization

Mode	Description	Major Advantage
<b>In-Memory</b>	Pure In-Memory Storage	Maximum performance possible (data is never written to disk)
<b>In-Memory + 3<sup>rd</sup> Party DB</b>	Caching layer (aka. in-memory data grid) for existing databases – RDBMS, NoSQL, etc	Horizontal scalability Faster reads and writes
<b>In-Memory + Full Copy on Disk</b>	The whole data set is stored both in memory and on disk	Survives cluster failures
<b>100% on Disk + In-Memory Cache</b>	100% of data is in Ignite native persistence and a subset is in memory	Unlimited data scale beyond RAM capacity

# Supports HTAP Principles

- One Platform Multiple Workloads
  - OLTP and OLAP
  - Real-Time Streaming and Batching
  - No ETL
- Distributed Everything
  - Scalability
  - No Single Point of Failure
  - SQL, Calculations, ML, etc.
  - Cloud Native



# Data Storage Ecosystem

Feature	RDBMS	NoSQL	IMDG	IMDB	GridGain and Ignite
Scale Out	X	✓	✓	✓	✓
Availability	X	✓	✓	✓	✓
Consistency	✓	X	✓	✓	✓
In-Memory	✓	X	✓	✓	✓
Persistence	✓	✓	X	X	✓
SQL	✓	X	X	✓	✓
Key-Value	X	✓	✓	X	✓
Collocated Processing	X	X	✓	X	✓



# Any Questions?

<https://www.gridgain.com>

<https://ignite.apache.org>