Handling Real-Time Use Cases at Scale Using a Hybrid-Memory Architecture

Srini V. Srinivasan

Founder, Chief Development Officer
Aerospike

In-Memory Computing Summit
Silicon Valley
October 25, 2017

Use Cases

AEROSPIKE

Fraud Prevention for Interactive Payments

Business Challenge

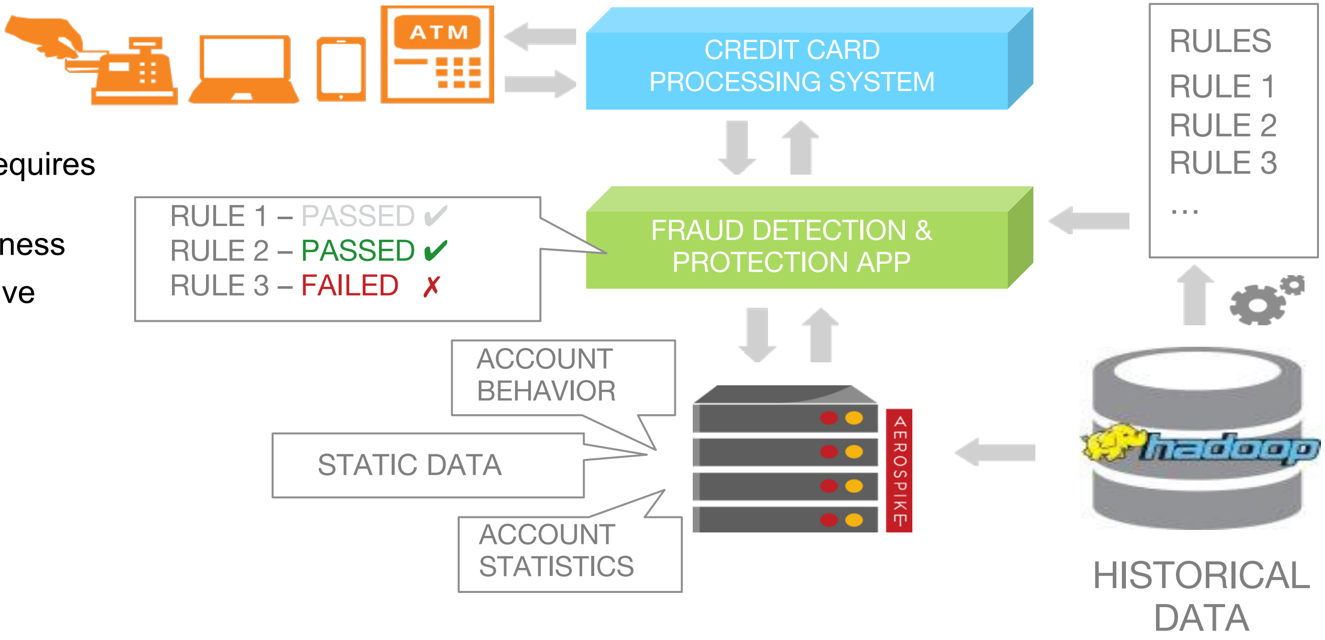
- Every payment transaction requires hundreds of DB reads/writes
- Missed latency SLA lost business
- Caching solution too expensive

Need to scale up

- 10 → 100 TB
- 10B → 100 B objects
- 200k → I Million+ TPS

Selected Hybrid Memory

- Built for Flash – eliminated inconsistencies
- Predictable Low latency at High Throughput
- 20 Server Cluster reduced from 150 in-memory cache servers



Retail Banking Positions – Risk Management

Business Challenge

- Must update stock prices, show balances on 300 positions
- process 250M transactions, 2 M updates/day
- Calculate risk metrics on portfolios on a continuous basis

Caching solution failed

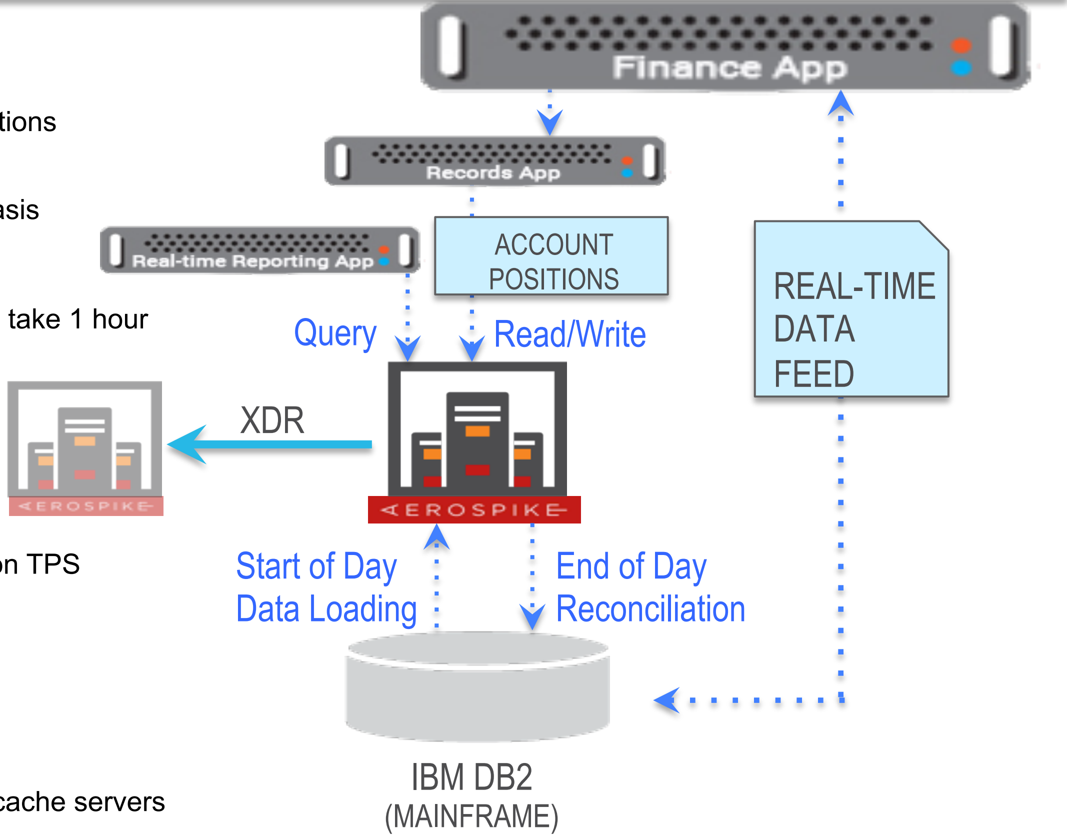
- Running out of memory, data inconsistencies, restarts take 1 hour
- 150 Servers -> Growing to 1000

Need to scale business

- 3 → 13 TB, 100 → 400 Million objects, 200k → I Million TPS

Hybrid Memory Advantage

- Built for Flash – eliminated inconsistencies
- Predictable Low latency at High Throughput
- 10-12 Server Cluster – reduced from 150 in-memory cache servers



Telco – Billing and Charging

Challenge

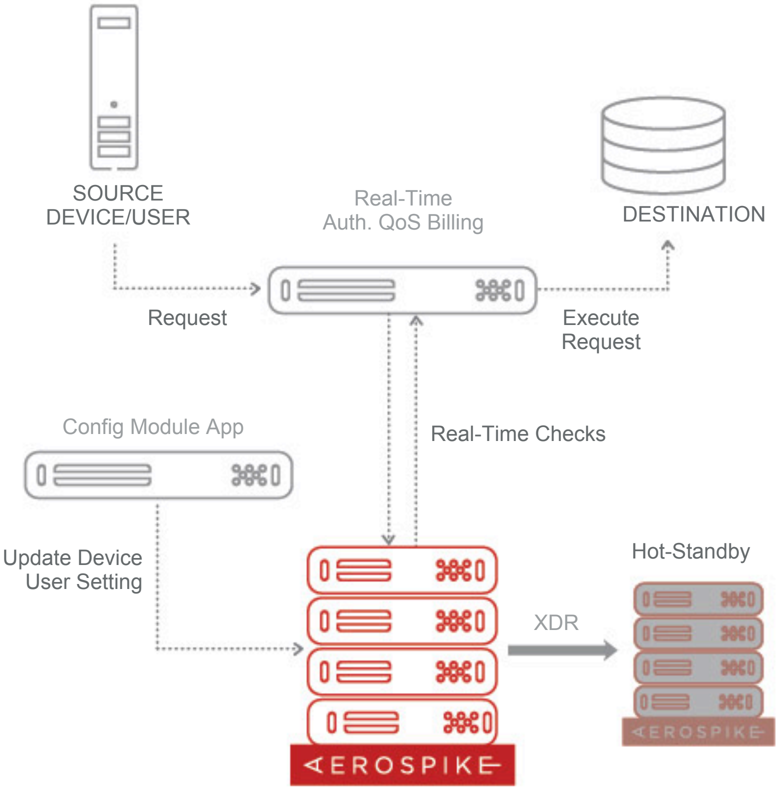
- Edge access to regulate traffic
- Accessible using provisioning applications (self-serve and through support personnel)
- Ensure accuracy in billing and charging
- Quick turn around for provisioning changes

Need Extremely High Availability, Reliability, Low latency

- > TBs of data
- 10-100M objects
- 10-200K TPS

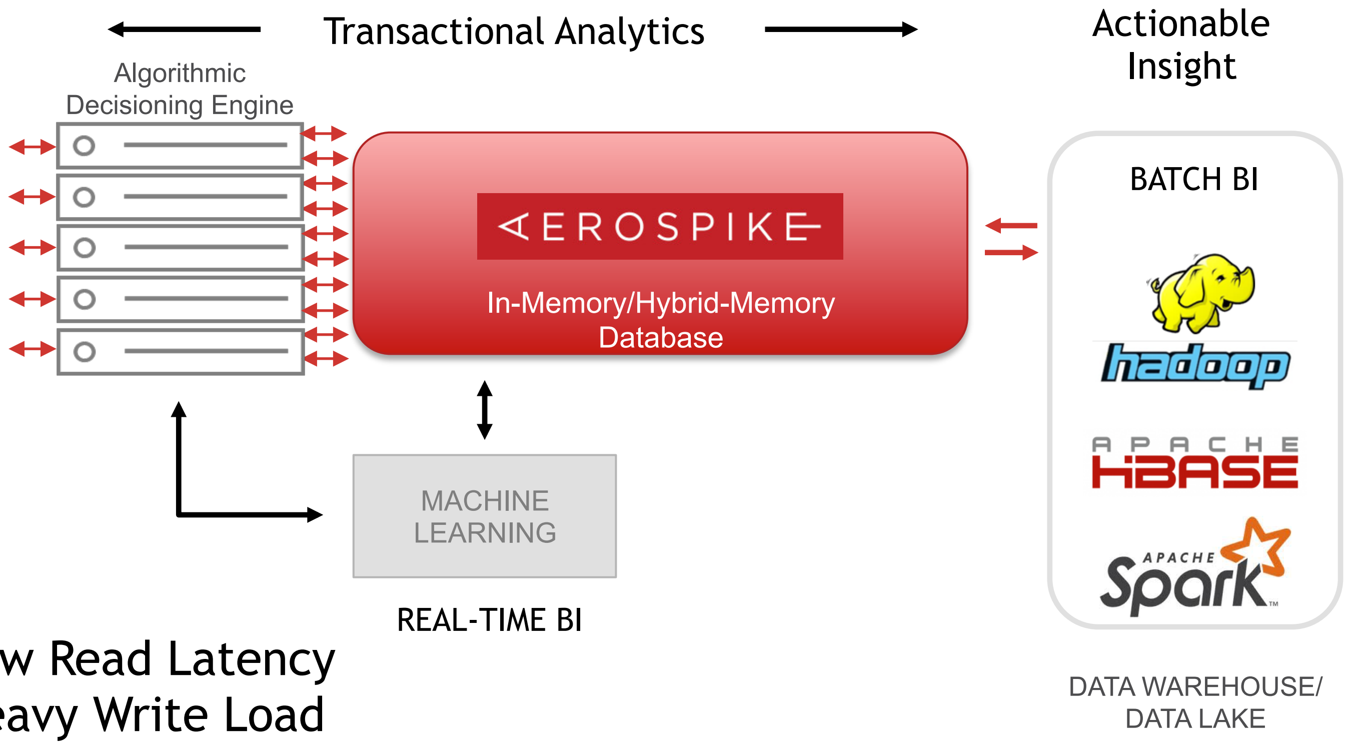
Selected Aerospike

- Clustered system
- Predictable low latency at high throughput
- Highly-available and reliable on failure
- Cross data center (XDR) support



Systems of Engagement

Fraud Prevention
Call Detail Record
Gaming
Banking
Payment Gateway
Real-time Bidding
Packet Routing
Msg / Chat
Look & Book
eCommerce
Session Management
IoT

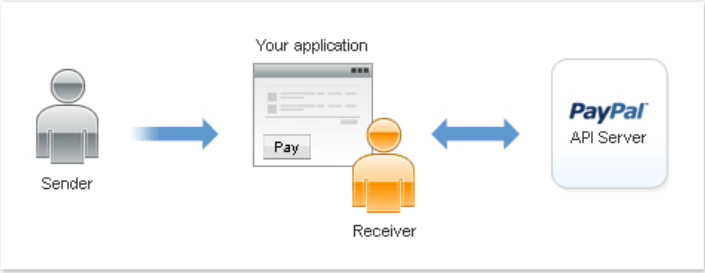


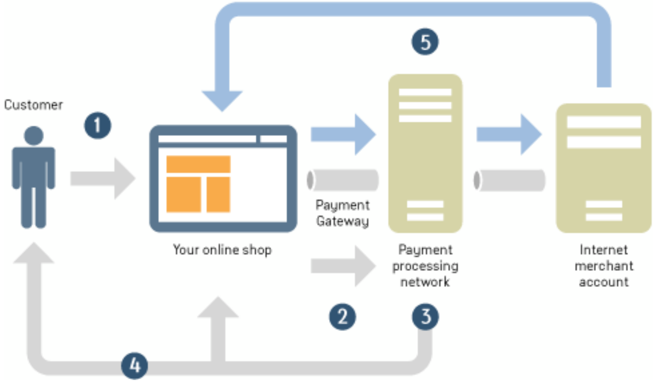
Low Read Latency
Heavy Write Load
100% Up Time

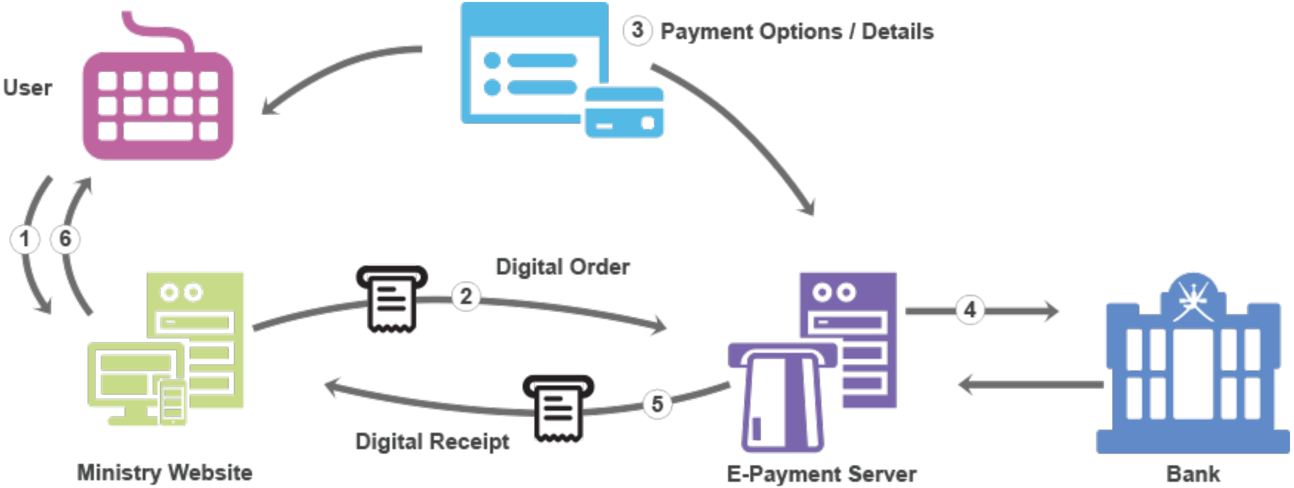
Example

The Scale Problem in Payment Fraud Detection

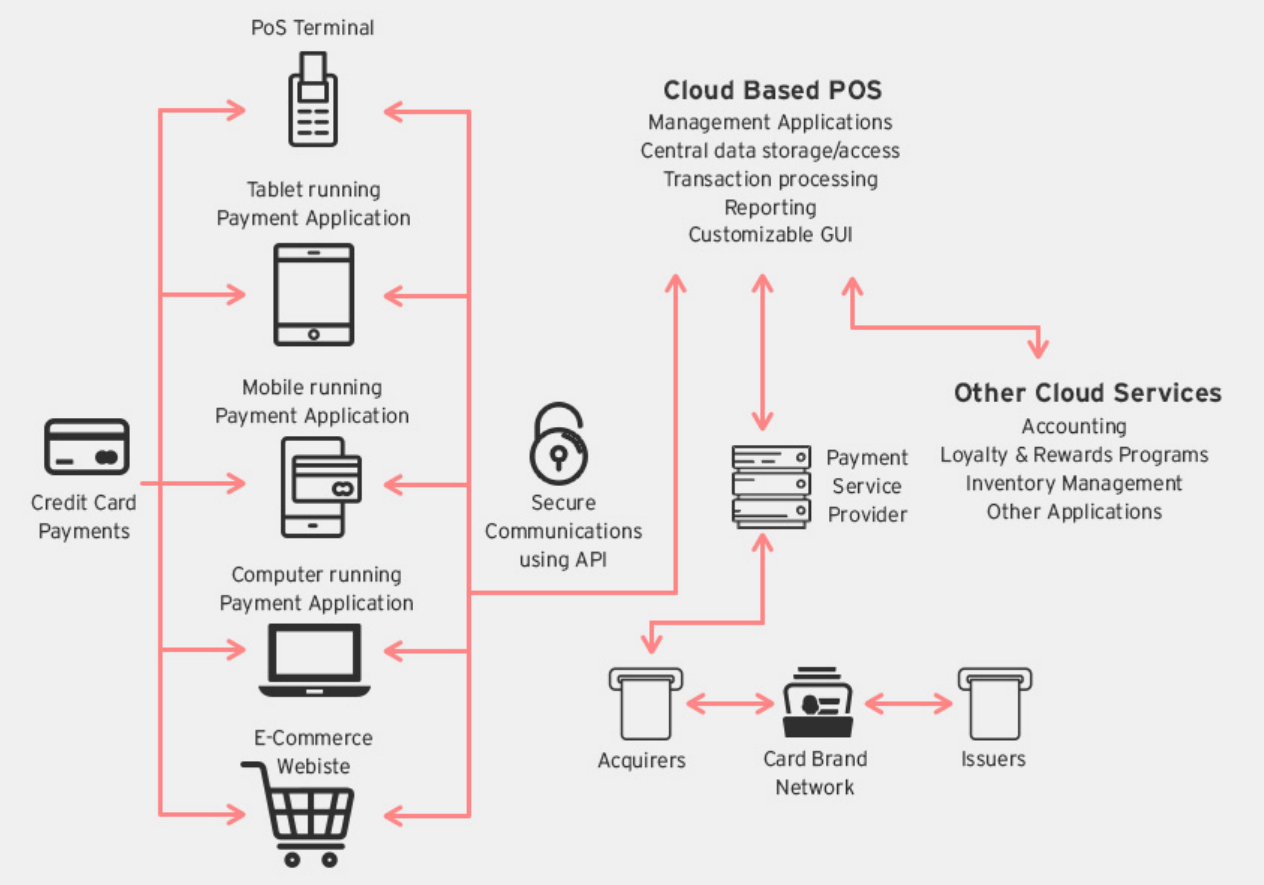
Payment systems are evolving fast



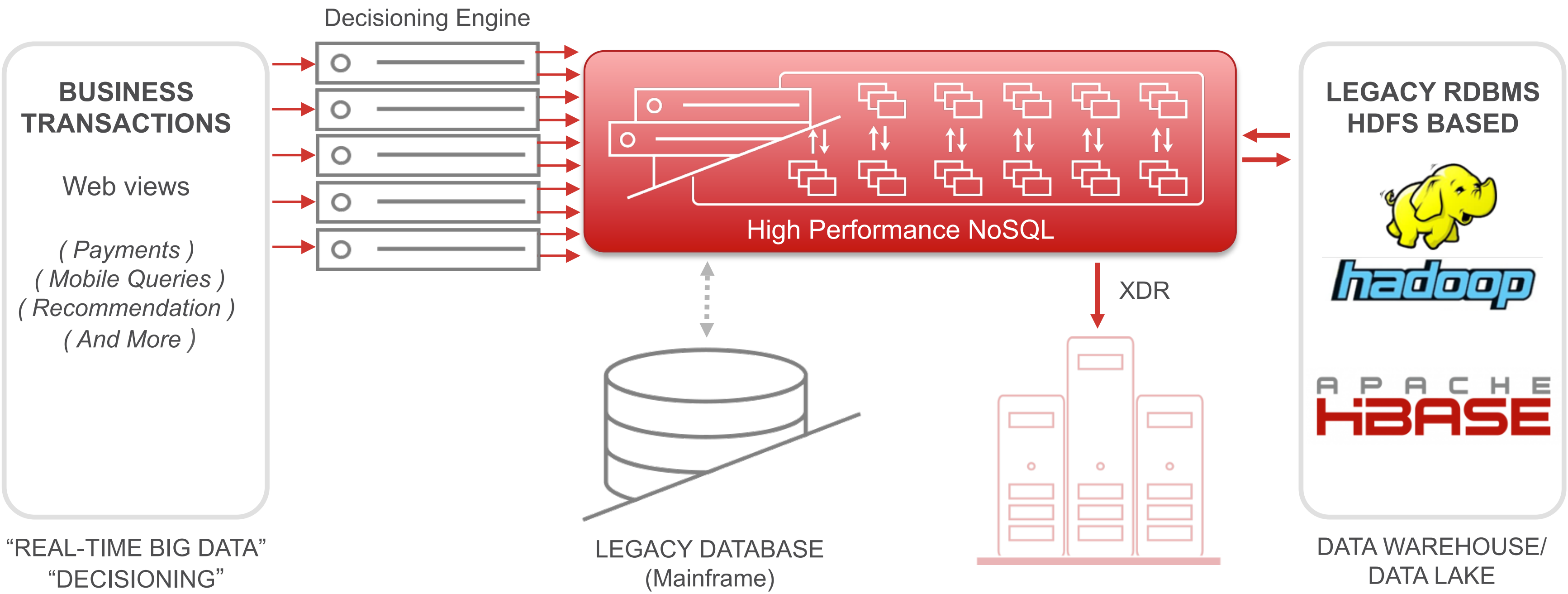




Payment systems have lots of actors



Operational Scale Explosion



Decisioning Engine

BUSINESS TRANSACTIONS

Web views

(Payments)
(Mobile Queries)
(Recommendation)
(And More)

High Performance NoSQL

XDR

LEGACY RDBMS HDFS BASED

“REAL-TIME BIG DATA”
“DECISIONING”

LEGACY DATABASE
(Mainframe)

DATA WAREHOUSE/
DATA LAKE

500
Business Trans per sec

X

5000
Calculations per sec

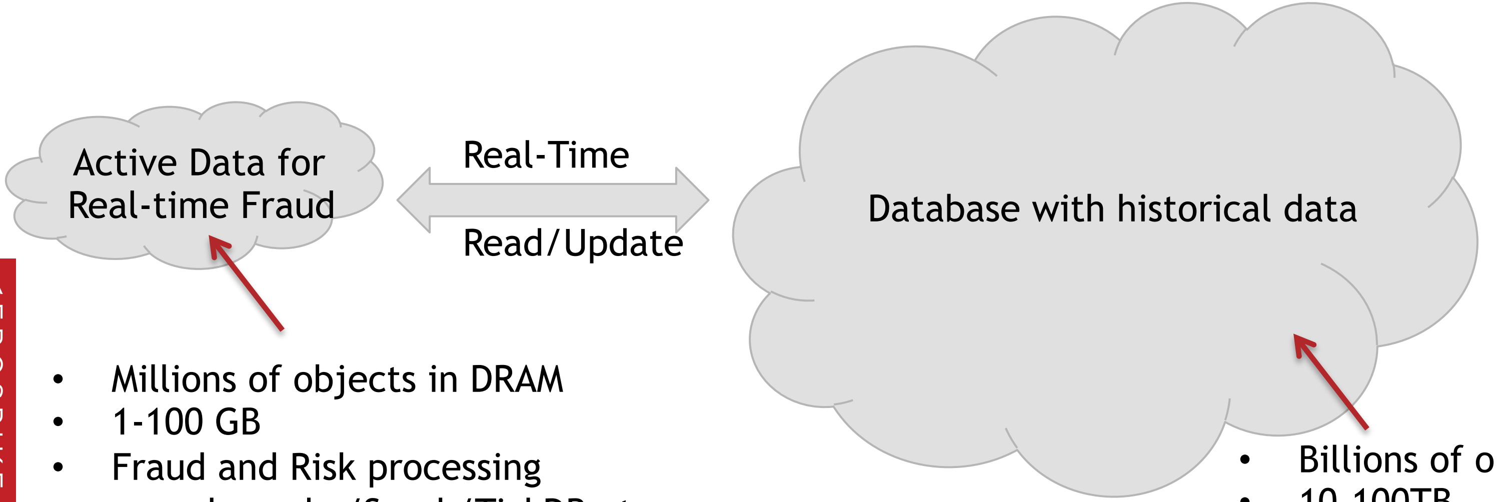
=

2.5 M
Database Transactions per sec

The billions of objects problem – Streaming vs DBMS

Streaming system can only store a limited number of objects in memory

Joining the active objects (millions) to database objects (billions) is best done using a distributed KVS


Active Data for Real-time Fraud
Real-Time
Read/Update
Database with historical data

- Millions of objects in DRAM
- 1-100 GB
- Fraud and Risk processing
  - bespoke/Spark/TickDB etc.

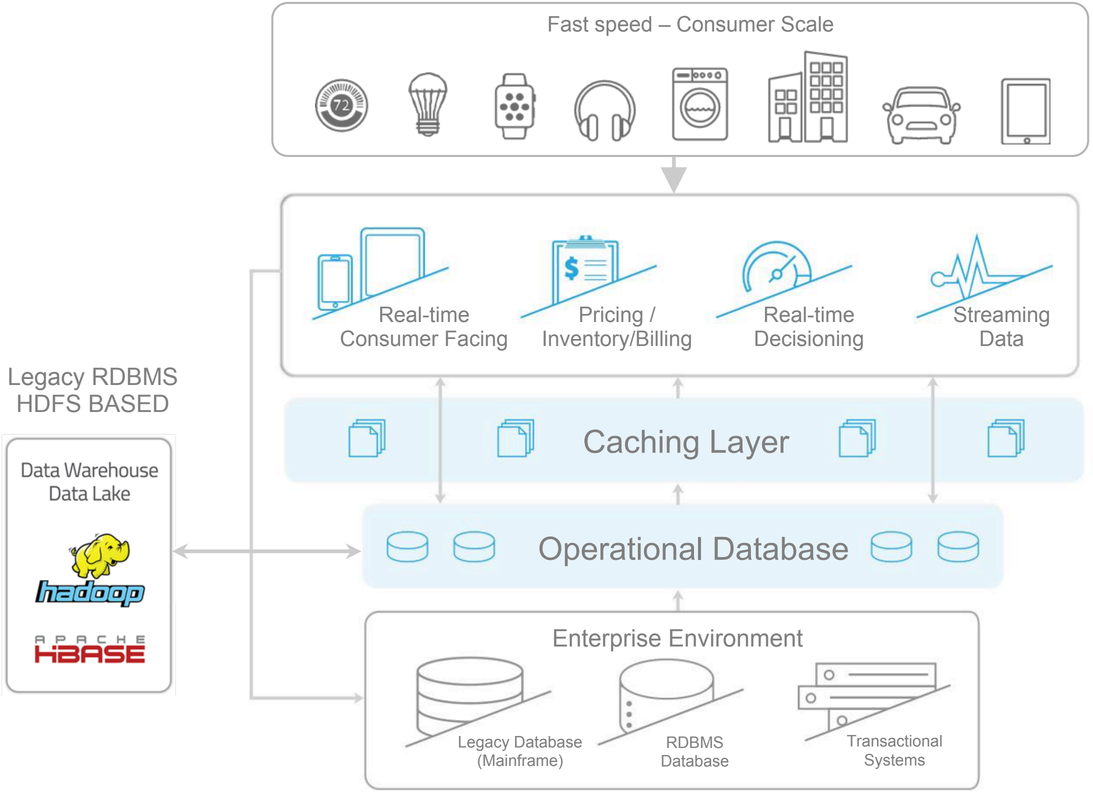
- Billions of objects on Storage
- 10-100TB
- Cassandra, Aerospike, etc.

Hybrid Memory Database

Systems of Engagement … what is required ?

- >> TPS (speed) … greater than 1 million tps
- >> Scale … greater than 5 to 10 TB
- << Low Latency … ~ 1 msec per transaction
- Reliability … ~ five 9s
- TCO … the lowest reasonable cost

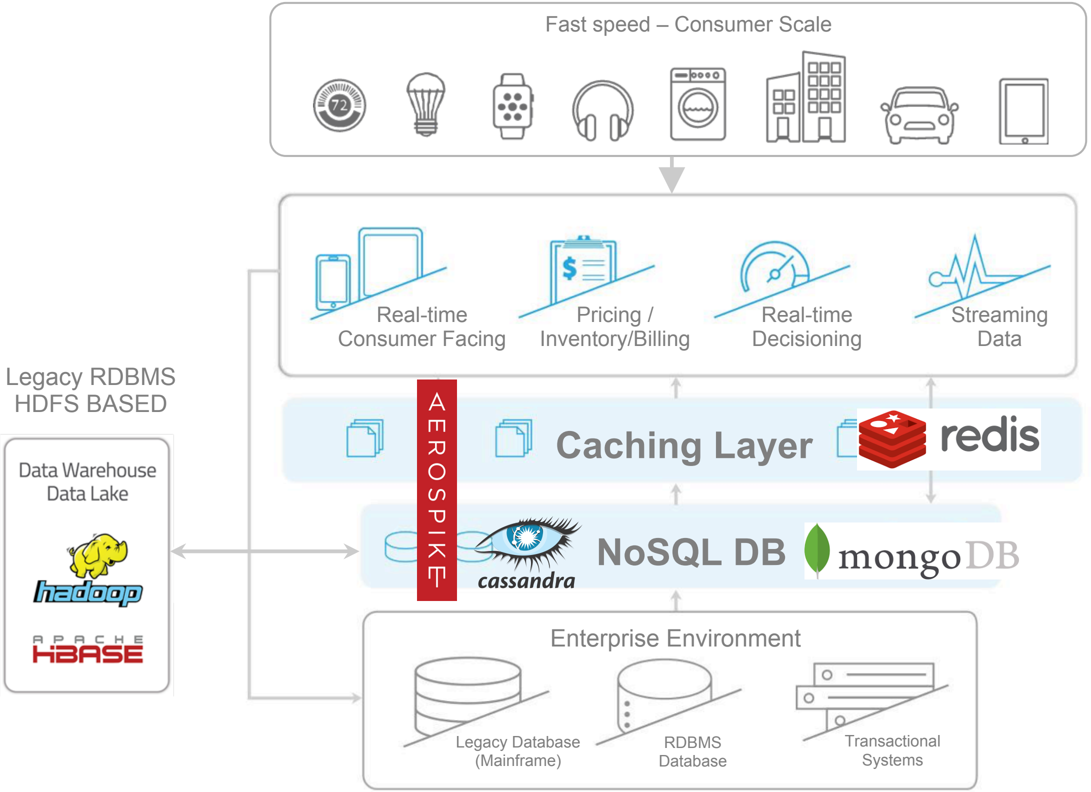
Traditional architecture has significant limitations



Challenges

- Complex
- Maintainability
- Durability
- Consistency
- Scalability
- Cost ($)
- Data Lag

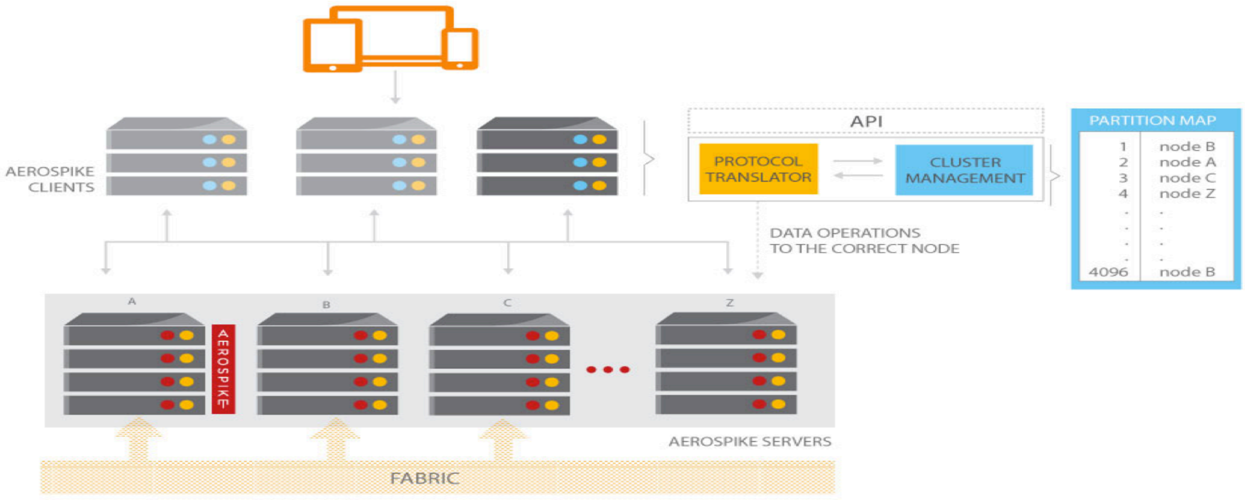
Hybrid Memory Architecture

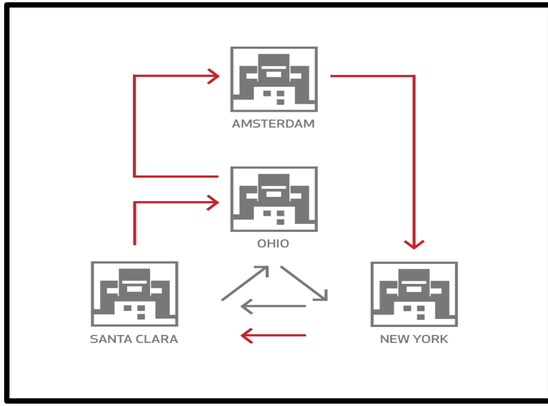


Benefits

- Fast App Development
- Richer data schema
- No need for SQL
- In-memory performance
- High scale
- Lower latency
- Distributed
- Tradeoff: Consistency versus Availability

Aerospike System Overview





1) **No Hotspots** – Distributed Hash Table simplifies data partitioning

2) Smart Client – **1 hop** to data, no load balancers

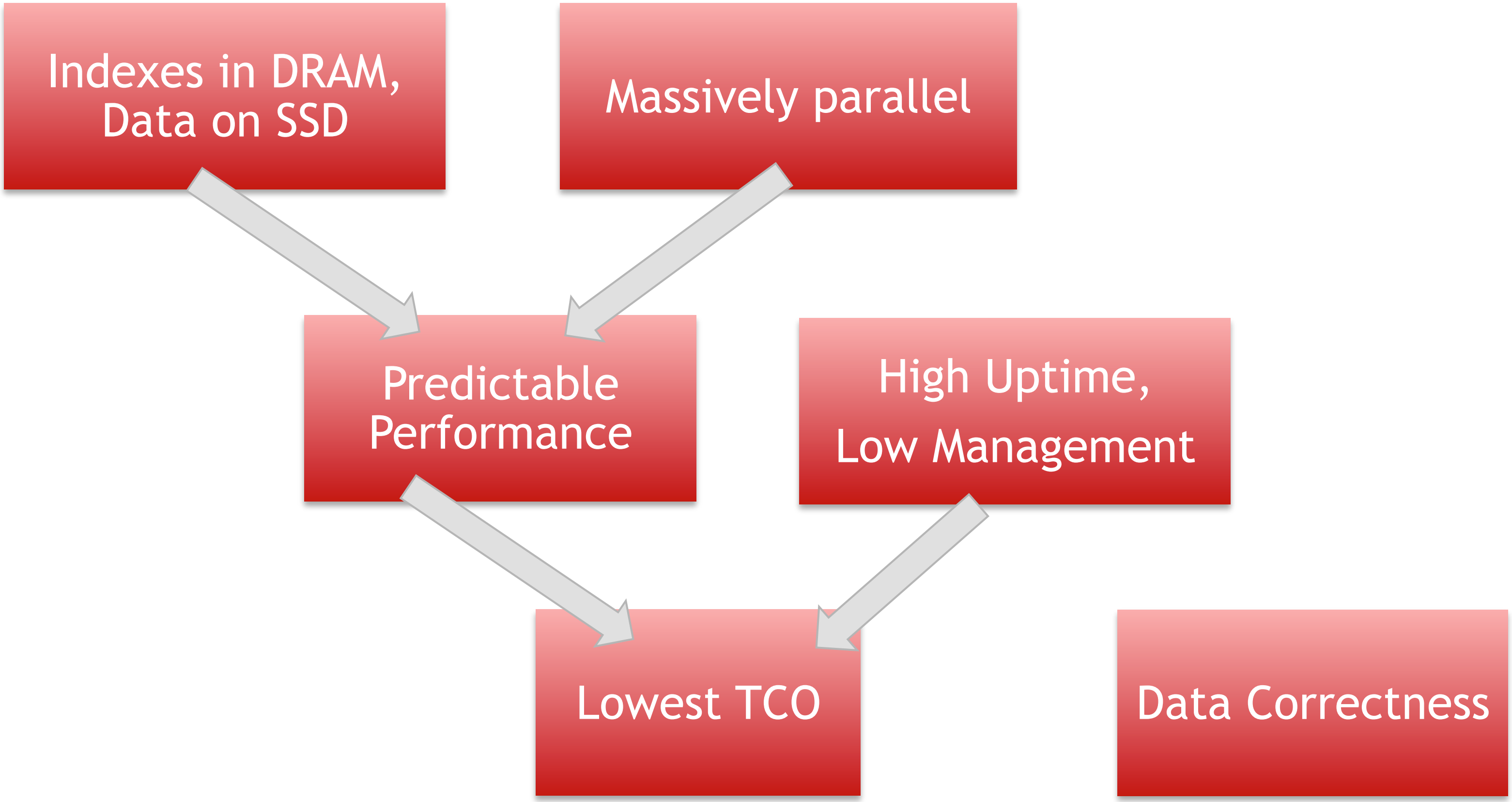
3) **Shared Nothing Architecture,** every node is identical

4) Smart Cluster, **Zero Touch** – auto-failover, rebalancing, rack aware, rolling upgrades

5) Transactions and long-running tasks **prioritized in real-time**

6) XDR – async replication across data centers ensures **Zero Downtime**

Attributes of a Hybrid Memory Architecture



Indexes in DRAM, Data on SSD

Massively parallel

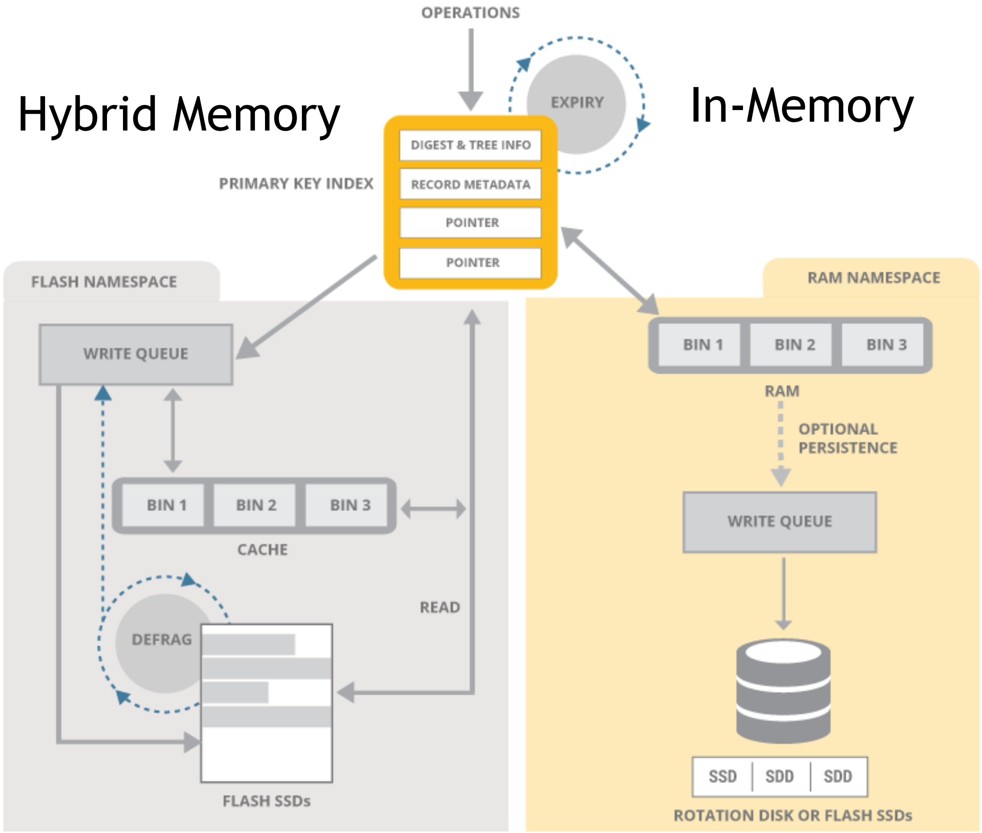
Predictable Performance

High Uptime, Low Management

Lowest TCO

Data Correctness

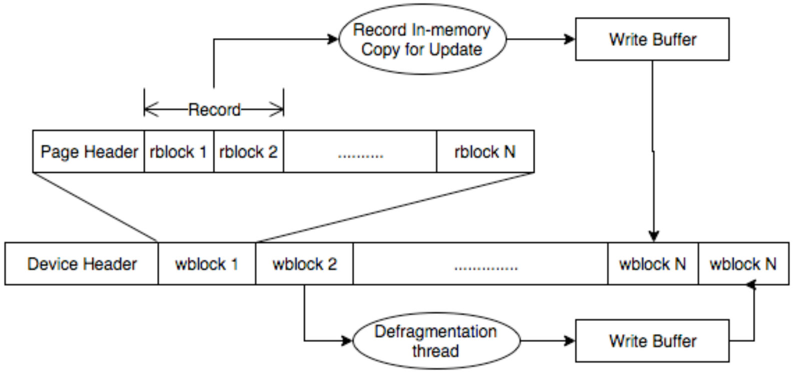
Aerospike Storage Architecture (HMA+)



Highlights

1. Direct device access
2. Large Block Writes
3. Indexes in DRAM
4. Highly Parallelized
5. Log-structured FS “copy-on-write”
6. Fast restart with shared memory

Storage Layout



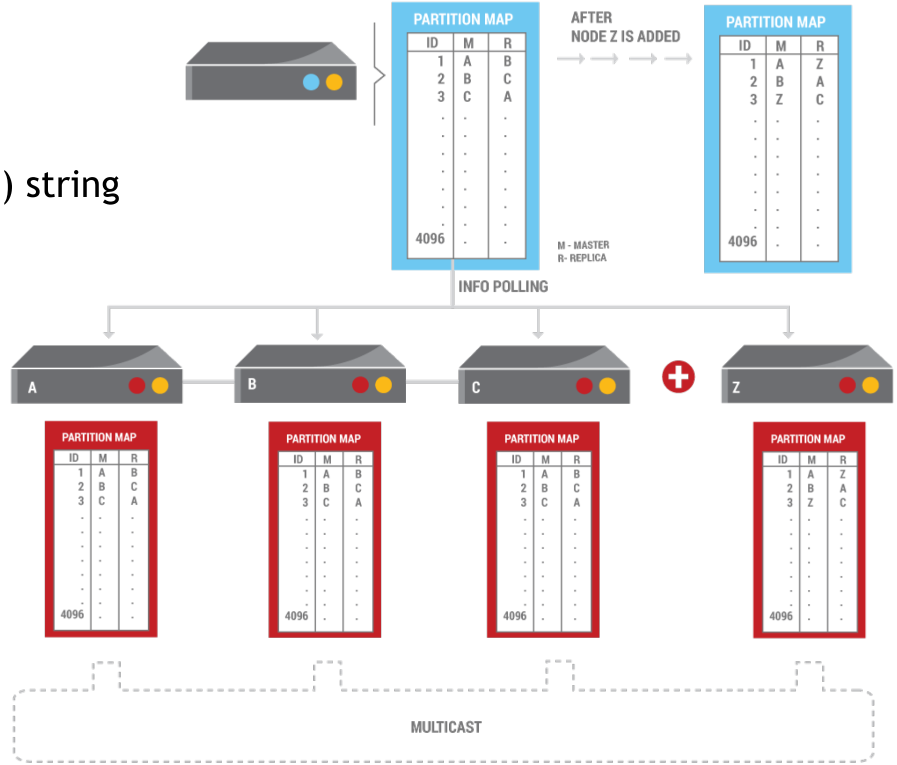
Hybrid Memory Characteristics

Indexes in DRAM, Data on SSD

- **Small amount of DRAM**
  - Avoid cost and server sprawl
- **No cache, so no cache misses**
  - Predictable, low latency performance on NVMe/SSD
- **Optimized for SSDs**
  - Reads done in parallel
  - Writes done optimally for SSD to reduce wear-and-tear

Distributed Hash Based Partitioning

- Distributed Hashing with No Hotspots
  - Every key **hashed** with **RIPEMD160** into an ultra efficient 20 byte (fixed length) string
  - Hash + additional (fixed 64 bytes) data forms **index entry** in RAM
  - **Some bits** from hash value are used to calculate the **Partition ID** (4096 partitions)
  - Partition ID maps to Node ID in the cluster



PARTITION MAP

ID	M	R
1	A	B
2	B	C
3	C	A
.	.	.
4096	.	.

PARTITION MAP

ID	M	R
1	A	Z
2	B	A
3	Z	C
.	.	.
4096	.	.

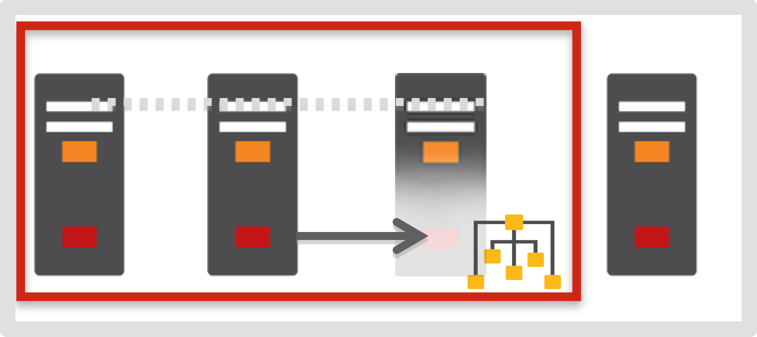
Automatic rebalancing

Adding, or removing a node, the cluster **automatically rebalances**

1. Cluster discovers new node via **gossip** protocol
2. **Paxos** vote determines new data organization
3. **Partition migrations** occur

After migration is complete, the cluster is **evenly** balanced.

Clients keep working during rebalancing.



AEROSPIKE

Massively Parallel

Massively parallel

- **Take full advantage of all the hardware**
  - Scaling up
- **Scale linearly with number of nodes**
  - Scaling out

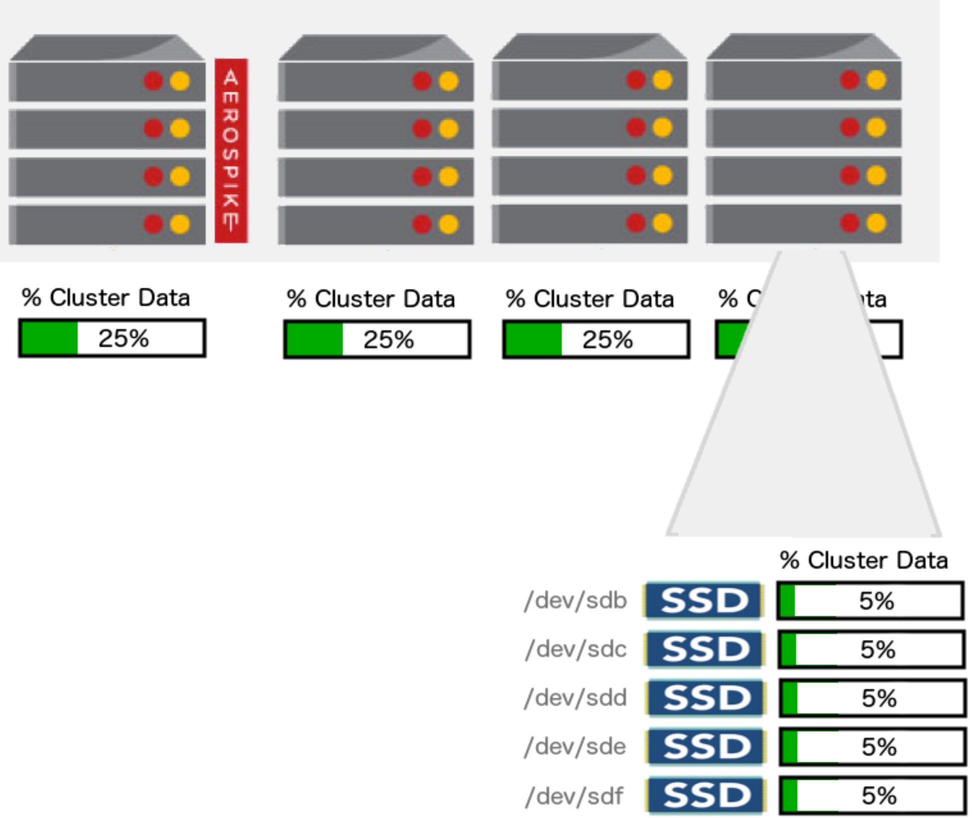
Massively Parallel

Automatic Distribution of Data using Smart Partitions™ algorithm

- Even amount data on every node and on every flash device
- All hardware used equally
- Load on all servers is balanced
- No “hot spots”
- No configuration changes as workload or use case changes

Smart Clients

- Single “hop” from client to server
- Cluster-spanning operations (scan, query, batch) sent to all processing nodes for parallel processing.



/dev/sdb	SSD	5%
/dev/sdc	SSD	5%
/dev/sdd	SSD	5%
/dev/sde	SSD	5%
/dev/sdf	SSD	5%

Aerospike’s Predictable Performance

Performance Built In

- Written in C with memory-optimized libraries => No garbage collection
- Continual defragmentation of storage => No compactions
- Known master for any piece of data => No quorum reads
- Designed as a distributed database => Networking primary consideration

Storage Optimizations

- Writes done to memory buffer => Avoid storage slowdown
- Storage used in “block” mode => No file system overhead
- Reads and writes striped across devices => Concurrent use of hardware

Smart Clients

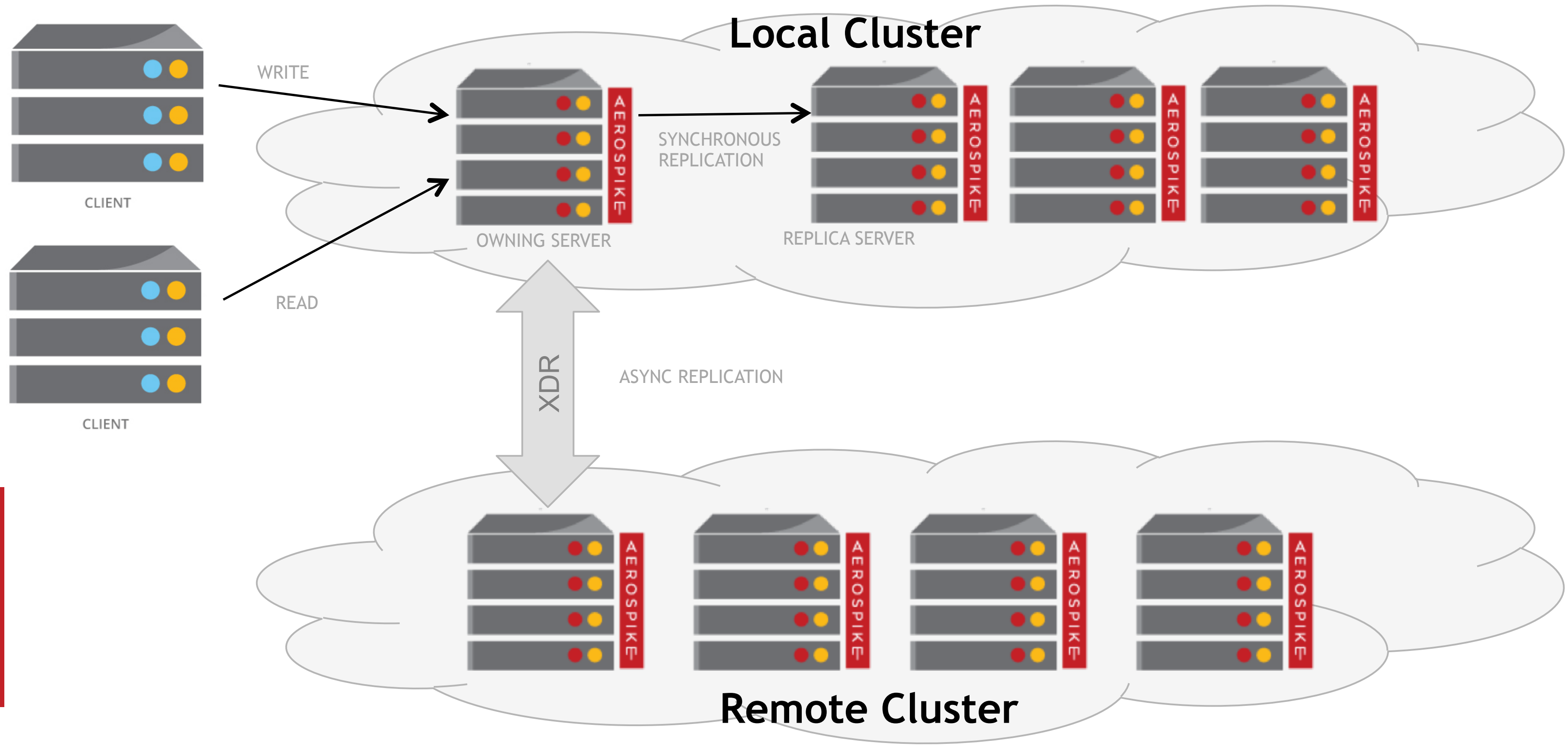
- Single “hop” from client to server
- Partition map stored on client
- Automatic load balancing – no external load balancers!

Data Correctness

Data Correctness

- **Reads should return the latest copy of the data**
  - With no latency penalty
- **Caches should not be necessary**
  - Eliminates stale data reads
- **Mixed workloads should not cause issues**
  - True concurrent reads/writes

Data Correctness



High Uptime, Low Management

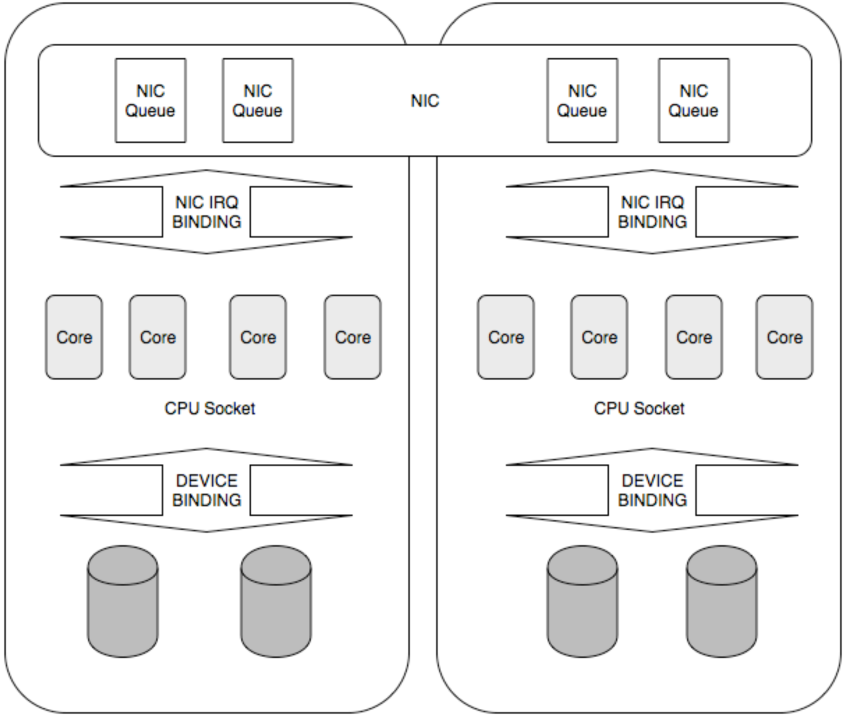
High Uptime,
Low Management

- **High Uptime**
  - “Shared Nothing” Architecture
  - No single points of failure
  - No cascading failures
  - Seamless loss of nodes with self-heal capability

- **Low Management**
  - Automatic sharding of data
  - No re-tuning of cluster for use-case changes
  - No requirement for caches
  - Smaller number of nodes for easier management
  - “Set and forget” DevOps management

Designed for Wire-Line Speed

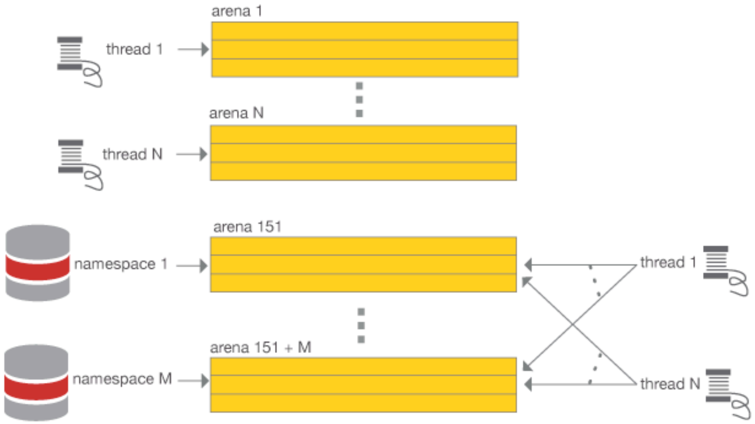
Multi-core architecture



Optimized C based DB kernel

1. Multi-threaded data structures
2. Nested locking model for synchronization
3. Lockless data structures
4. Partitioned single threaded data structures
5. Index entries are aligned to cache line (64 bytes)
6. Custom memory management (arenas)

Memory Arena Assignment

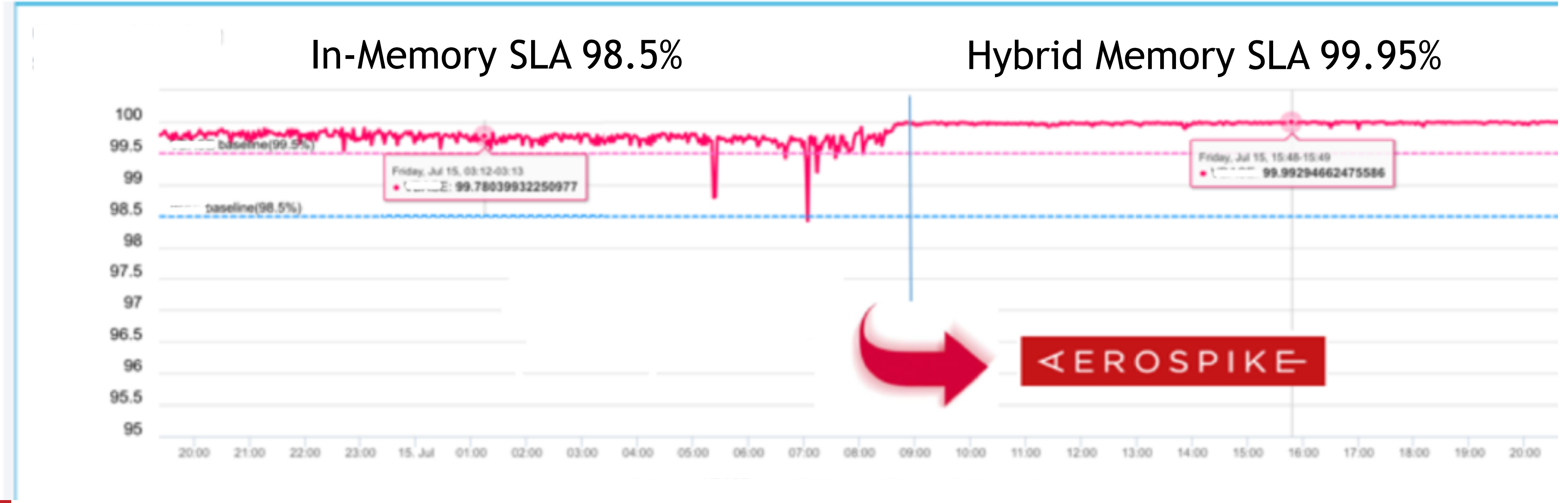


AEROSPIKE

Hybrid Memory Benefits

In Memory vs. Hybrid Memory – SLA in Actual Deployment

30X Improvement in SLA

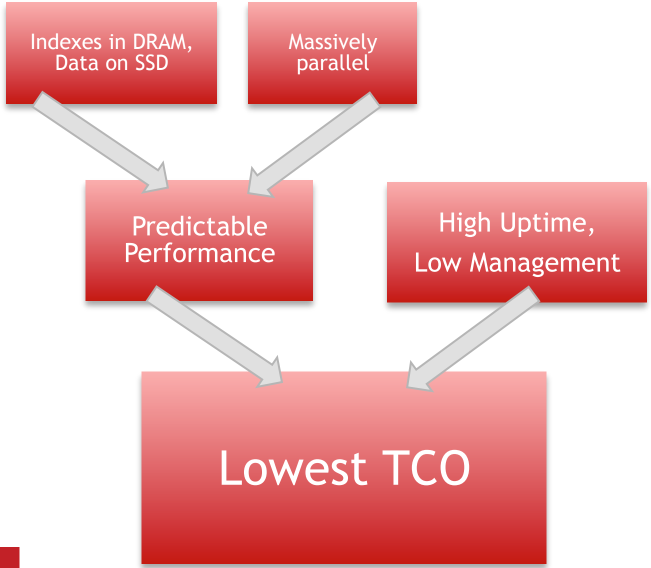


In-Memory SLA 98.5%

Hybrid Memory SLA 99.95%

Missed SLA is lost Revenue!!!

Lowest TCO



- **Hybrid Memory Architectures offer**
  - Cacheless, consistent performance using NVMe/ Optane.
  - Server count reduced (3x or more)
  - Significant reduction in TCO (10x documented)

In-Memory vs Hybrid Memory – Low TCO @ Scale

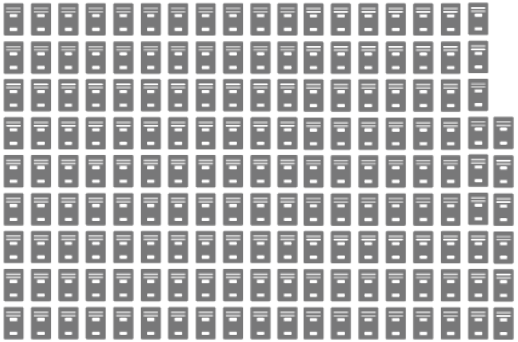
UP TO

10x FASTER

10x FEWER

ACTUAL CUSTOMER ANALYSIS

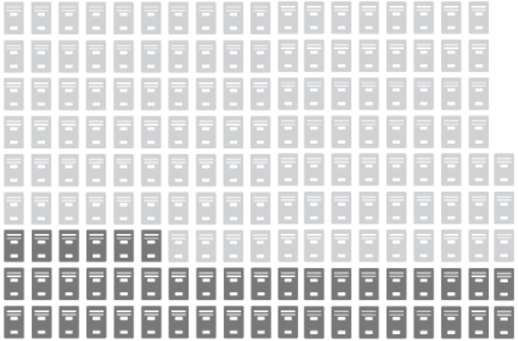
168 SERVERS



OTHER DATABASE

DRAM & HDD

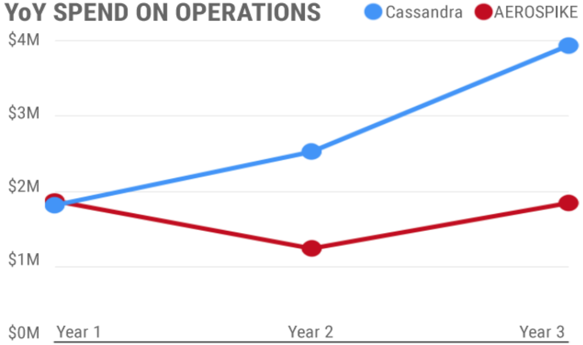
ONLY 44 SERVERS



SSD & DRAM

	Year 1 (Millions USD)	Year 2 (Millions USD)	Year 3 (Millions USD)	Total (Millions USD)
Cassandra	$1.82	$2.53	$3.94	$8.28
Aerospike	$1.88	$1.24	$1.85	$4.97
Total OpEx Savings	**-$0.06**	**$1.28**	**$2.09**	**$3.32**

YoY SPEND ON OPERATIONS

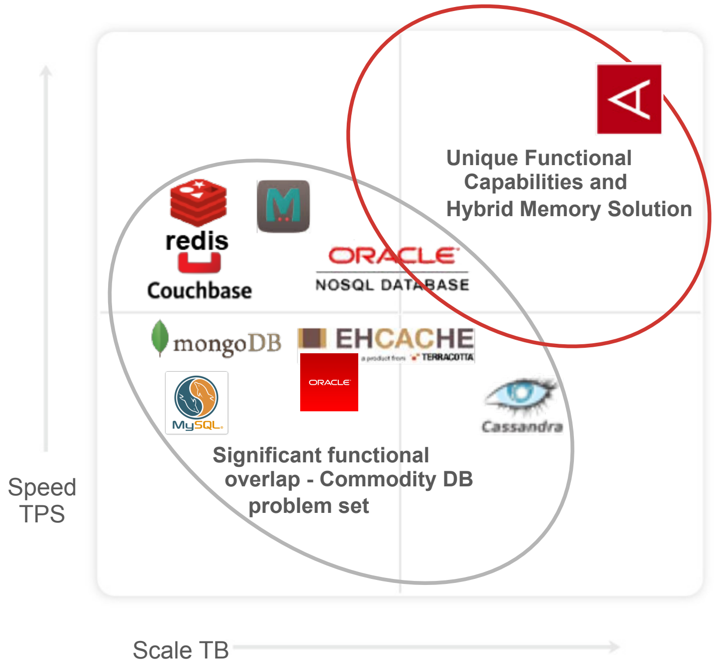


Case Studies: HMA - Lower TCO & better SLA

Customer	Situation	Problem	Hybrid Memory System
Trading Account Risk Management	DB2+Gemfire cache	150 Servers growing to 1000	Single cluster – 12 servers
Payments Fraud Detection	2 ORCL RAC clusters + Terracotta cache	System Stability & missing SLA's	3 Clusters – 20 Servers each
User Integrity Checking for Internet Transactions	DataStax/Cassandra	168 DataStax Servers growing to 450+	30 Servers – 2 clusters
Telco Device and User Access	ORCL Coherence / DataStax Cassandra	Existing SOE solutions unstable & Costly	5 successful POC's
Telco Revenue Assurance	DataStax/Cassandra PostgreSQL + cache	Hundreds of cache & Cassandra Servers Scalability challenges	Significant reduction of server footprint – global deployment

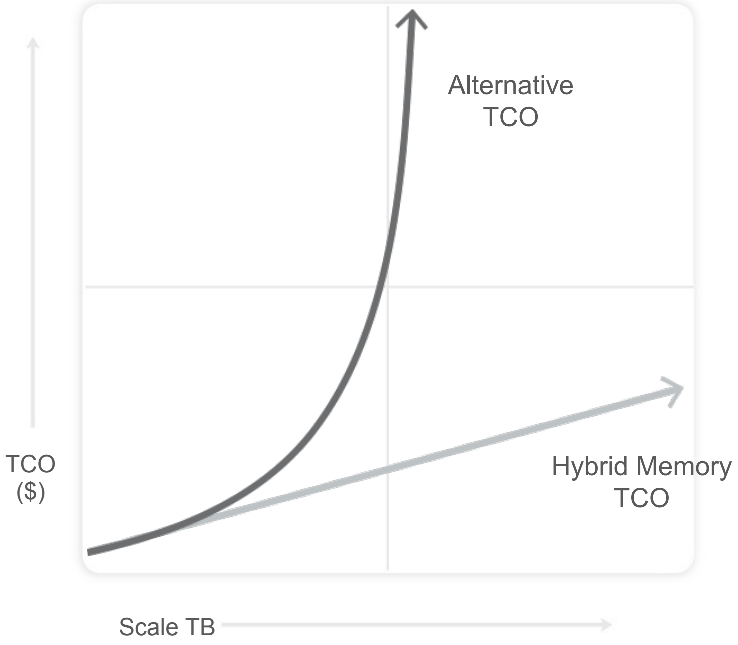
Next Generation Systems of Engagement –
An Emerging Market with Multiple Technologies

Systems of Engagement – Many Choices



Unique Functional Capabilities and Hybrid Memory Solution
redis
Couchbase
ORACLE NOSQL DATABASE
mongoDB
EHCACHE
TERRACOTTA
ORACLE
MySQL
Cassandra
Significant functional overlap - Commodity DB problem set
Speed TPS
Scale TB

Systems of Engagement - TCO



Hybrid Memory Architecture Delivers Predictable Performance, Highest Availability, and Lowest TCO

Thank You

Questions?

AEROSPIKE