ORACLE®

Oracle NoSQL Database – A Distributed Key-Value Store

Marie-Anne Neimat

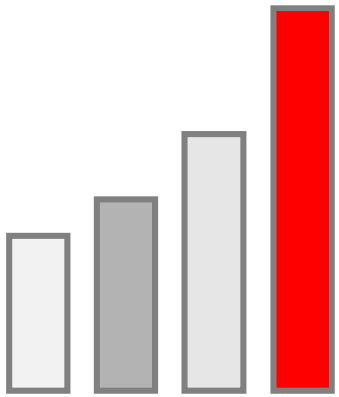
November 9, 2011

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

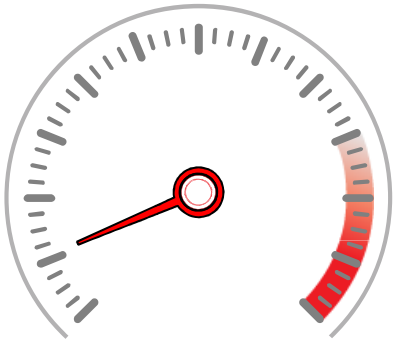
Agenda

- Big Data Overview
- Oracle and NoSQL
- Oracle NoSQL Database
  - Architecture
  - Technical Overview
  - Benchmark Results

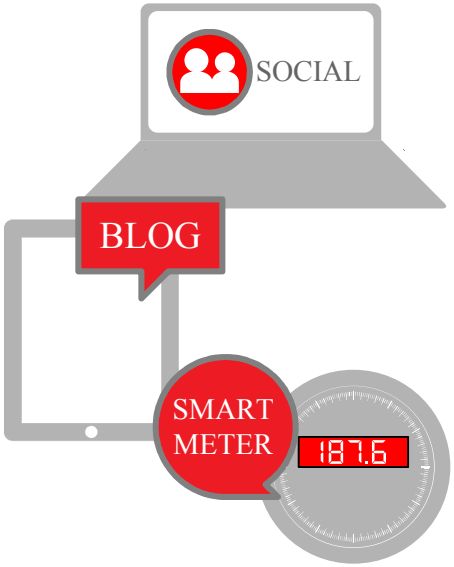
What is Big Data ?



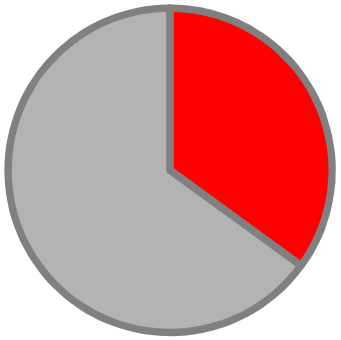
VOLUME



VELOCITY



VARIETY



VALUE

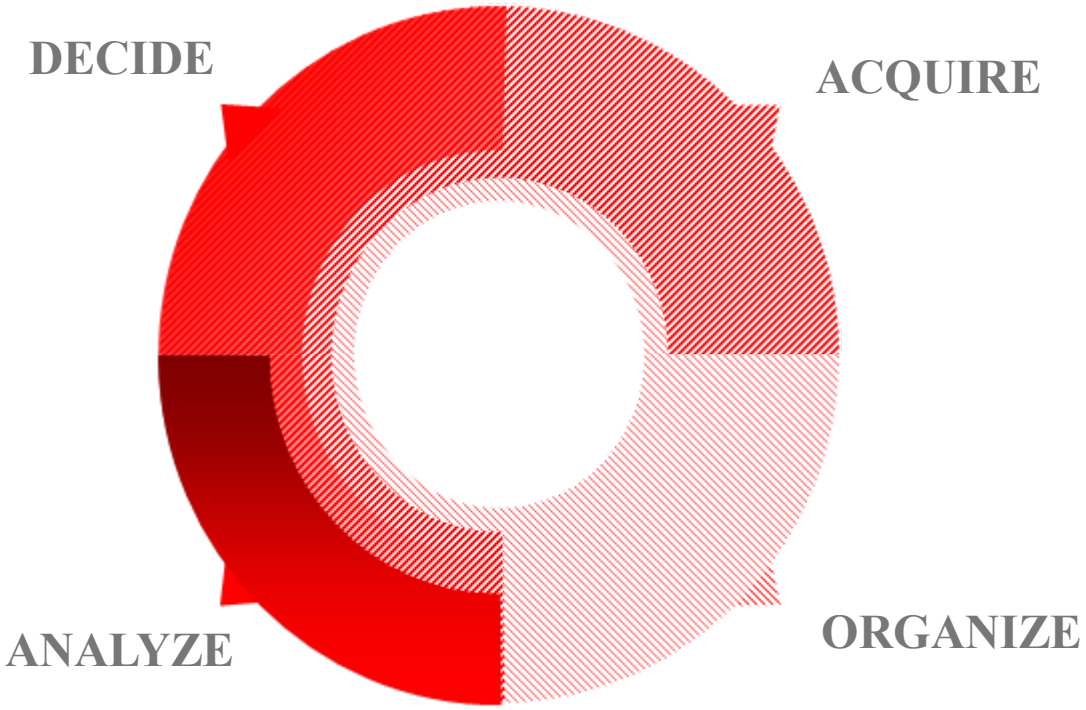
Why Is Big Data Important?

US HEALTH CARE	US RETAIL	MANUFACTURING	GLOBAL PERSONAL LOCATION DATA	EUROPE PUBLIC SECTOR ADMIN
Potential Annual Value	Potential Increase in Retailers' Operating Margins	Potential decrease in cost of dev & assembly	Potential increase in service provider revenue	Potential Annual Value
$300 B	60%	50%	$100 B	€250 B

"In a big data world, a competitor that fails to sufficiently develop its capabilities will be left behind."

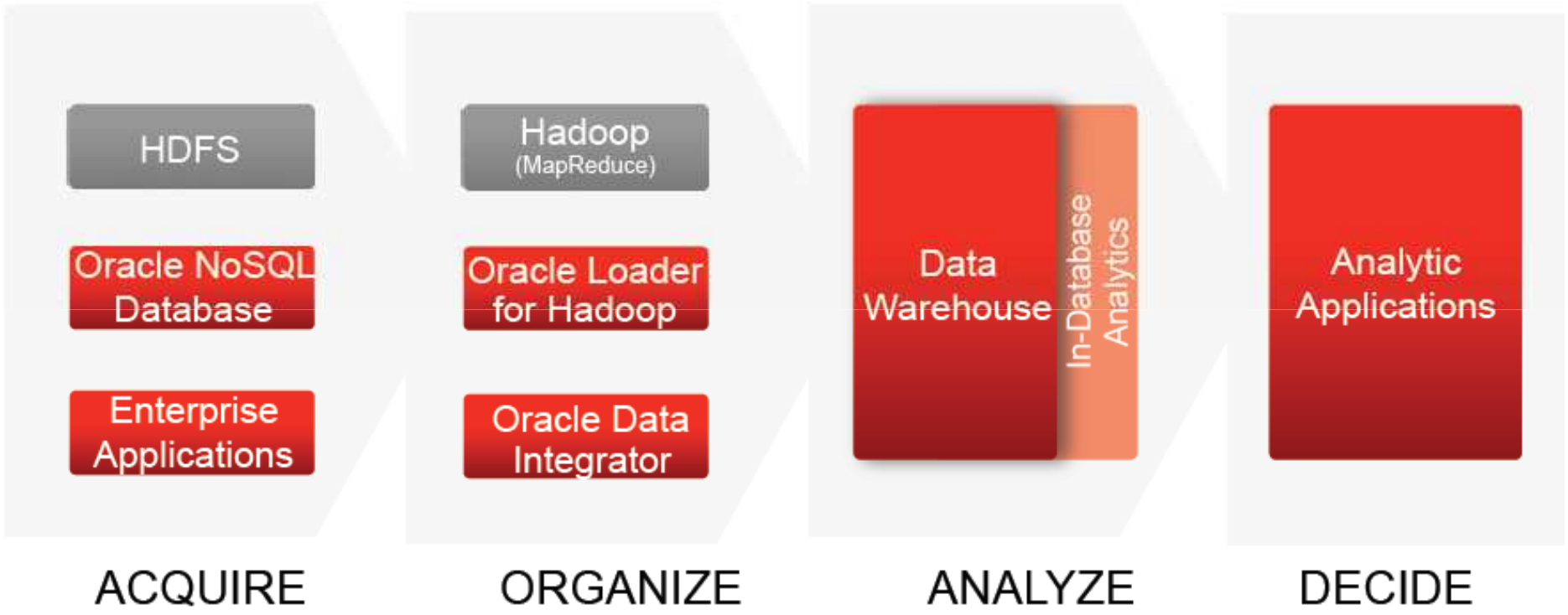
Source: * McKinsey Global Institute: Big Data – The next frontier for innovation, competition and productivity (May 2011)

Big Data in Action

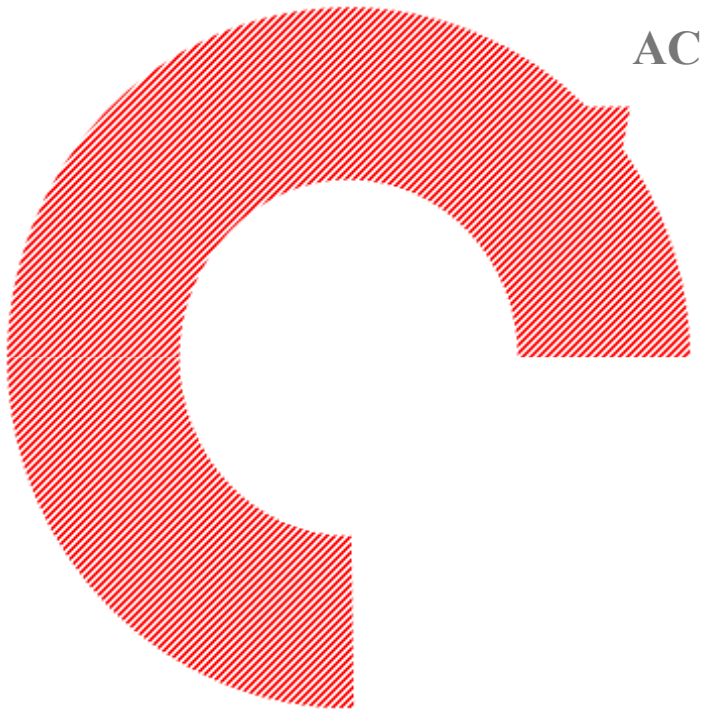


Make Better Decisions Using Big Data

Oracle Integrated Solution Stack for Big Data



Acquiring Big Data Challenge



- Application will need to change frequently
- Need to process high volume, low density information from various data-sets
- Must deliver sub-millisecond velocity
- Must scale out to meet aggressive roll out plan

Agenda

- Big Data Overview
- Oracle and NoSQL
- Oracle NoSQL Database
  - Architecture
  - Technical Overview
  - Benchmark Results

Characteristics of NoSQL System

- Distributed and scalable
- Large data (Terabyte – Petabyte range)
- Two categories
  - OLTP ← Our focus is here
  - Batch Processing (M/R & Hadoop) ← … and we integrate here
- Data Models
  - Key-Value ← Our focus is here
  - Document (e.g., CouchDB, MongoDB)
  - Columnar (e.g., Cassandra)
  - Graph (e.g., Neo4J)

Target Use Cases

- Large schema-less data repositories
  - Web applications (click-through capture)
  - Online retail
  - Sensor/statistics/network capture (factory automation for example)
  - Backup services for mobile devices
  - Scalable authentication
  - Personalization
  - Social Networks

Design Requirements

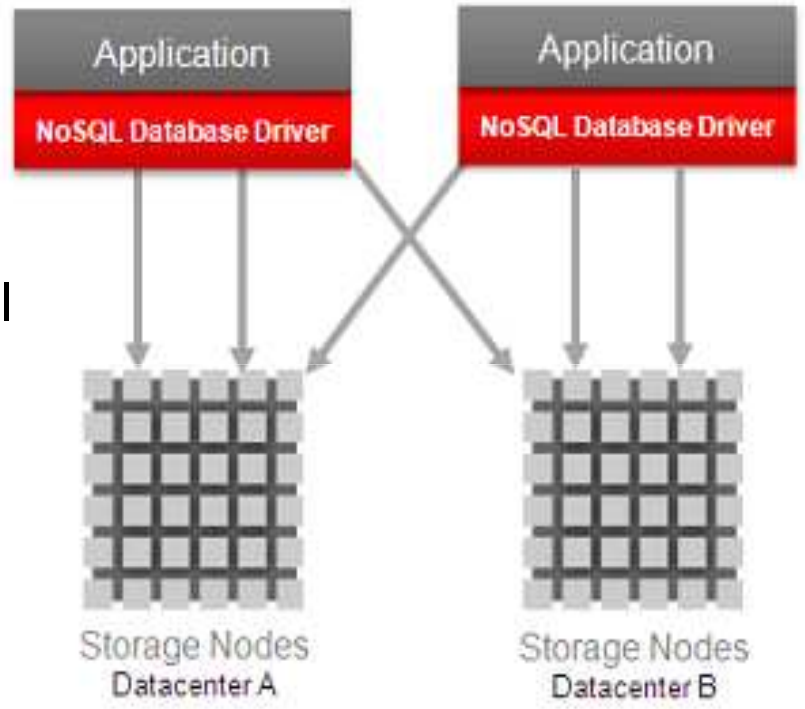
- Terabytes to Petabytes of data
- 10K's to 1M's ops/sec
- No single point of failure
- Elastic scalability on commodity hardware
- Fast, bounded response time to simple queries
- Flexible ACID transactions
- Unstructured or semi-structured data, with clustering capability
- Simple administration, enterprise support
- **Commercial-grade** NoSQL solution

Agenda

- Big Data Overview
- Oracle and NoSQL
- Oracle NoSQL Database
  - Architecture
  - Technical Overview
  - Benchmark Results

Oracle NoSQL Database

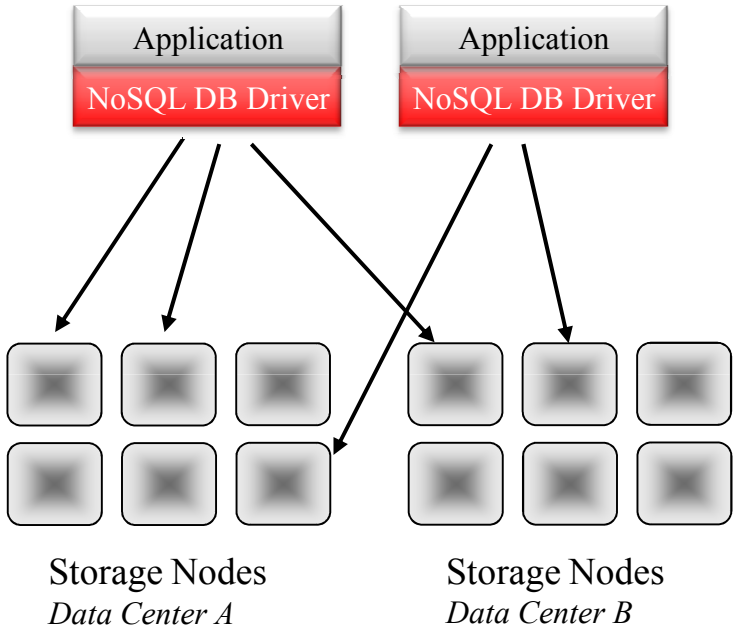
- Distributed key-value database
- Data distributed based on hashed value of primary key
- Storage nodes replicated for high availability, rapid failover and optimal load balancing of queries.
- Easy-to-use Java API.
- Rapid access to nodes and optimal load balancing with NoSQL DB Intelligent Driver
- Easy administration via either a web console or command line interface.



Oracle NoSQL DB Overview

A Distributed, Scalable Key-Value Database

- Simple Data Model
  - Key-value pair with major+minor-key paradigm
  - Read/insert/update/delete
- Scalability
  - Dynamic data partitioning and distribution
  - Optimized data access via intelligent driver
- High availability
  - One or more replicas
  - Resilient to partition master failures
  - No single point of failure
  - Disaster recovery through location of replicas
- Transparent load balancing
  - Reads from master or replicas
  - Driver is network topology & latency aware



Oracle NoSQL Database Building Blocks

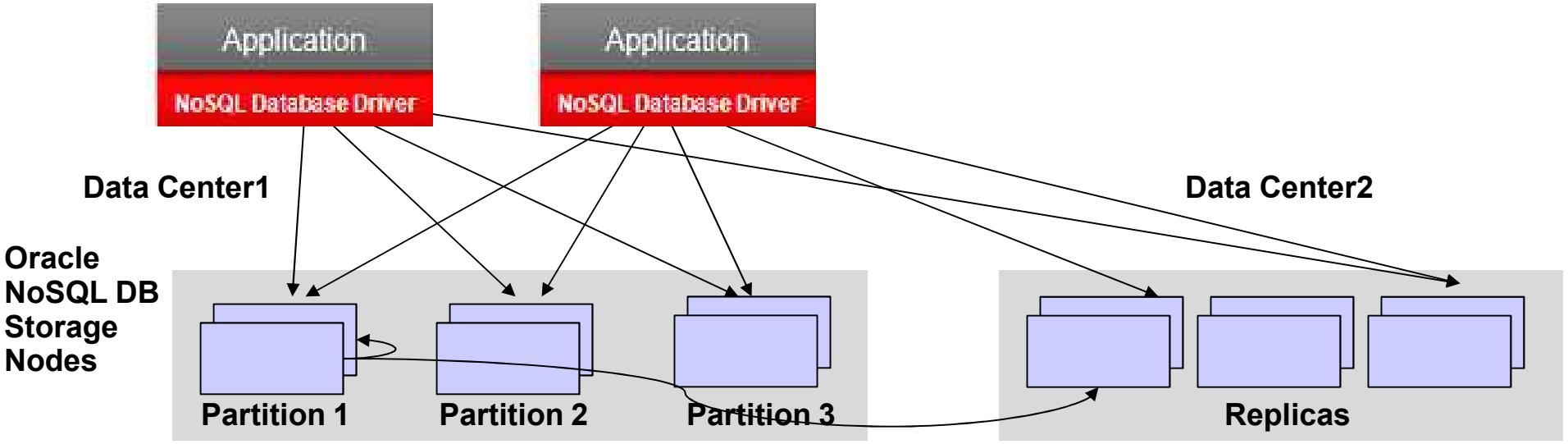
Berkeley DB

- Robust storage for a distributed key-value database
  - ACID transactions
  - Persistence
  - High availability
  - High throughput
  - Large capacity
  - Simple administration
- Already used in
  - Amazon Dynamo
  - GenieDB
  - Yammer

"The learning curve was steep, but the view from the top of the hill is pretty nice."
- Yammer

Oracle NoSQL DB Typical Topology

- NoSQL Database Driver is linked to each Application Process
- Data Nodes are kept current (underlying BDB-JE HA technology)
- Nodes may live in multiple Data Centers
- Node failure handling
  - Graceful degradation until node is fixed or replaced
  - Automatic recovery



Agenda

- Big Data Overview
- Oracle and NoSQL
- Oracle NoSQL Database
  - Architecture
  - Technical Overview
  - Benchmark Results

Oracle NoSQL Database

From Programmer point of view

- Simple data model – key-value pair (major/minor key paradigm)
- Simple operations – CRUD, RMW (CAS), iteration
- ACID transactions for operations on minor keys within a major key, single API call
- Unordered scan of all data (non-transactional)
- Ordered iteration across minor keys within a major key

Replication Group 1

```
sue.bd→[billingdata]
sue.hist→[historydata]
sue.promo→[promodata]
george.bd->[billingdata]
george.hist->[historydata]
george.promo->[promodata]
```

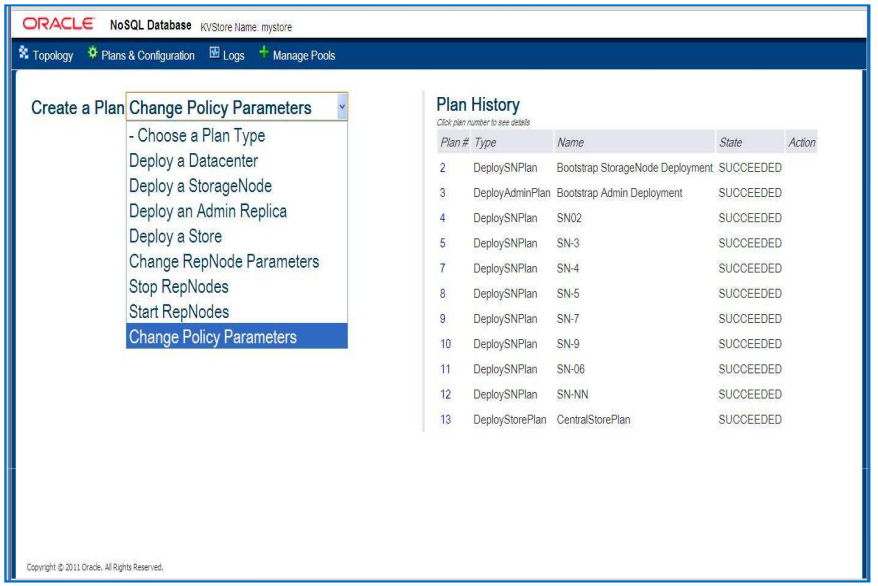
Replication Group 2

```
joe.bd→[billingdata]
joe.hist→[historydata]
joe.promo→[promodata]
linda.bd->[billingdata]
linda.hist->[historydata]
linda.promo->[promodata]
```

Oracle NoSQL Database

Easy Management

- Administrative Service from Web console & CLI
- Manages DB instances
  - Start / stop
  - Topology
  - Configuration changes
- Monitors
  - *Load*: Number of operations, data size
  - *Performance*: Latency, throughput. Min, max, average, …
  - *Events*: Failover, recovery, load distribution
  - *Alerts*: Failure, poor performance, …



Oracle NoSQL Database
Flexible Durability Options

- Specified on per-operation basis, default can be changed
- Two dimensions to durability
  - Writing to stable storage
    - Policies: Commit to RAM, Commit to Disk
  - Updating replicas
    - Policies: All, Simple Majority, Local copy
- Durability policies specify the guarantees that the system makes after a crash

Oracle NoSQL Database
Flexible Consistency Options

- Specified on per-operation basis, default can be changed
- Consistency policy options
  - Absolute (read from Master)
  - Time-based
  - Version
  - None (read from any node)

Oracle NoSQL Database Differentiation

Integrates seamlessly with Oracle Stack (ODI, CEP, OLH)

Commercial Grade Software and Support

- General Purpose
- Reliable – Based on proven Berkeley DB JE HA
- Easy to Install & Configure

Scalable throughput and bounded Latency

- Intelligent Oracle NoSQL DB Driver
  - Evenly distributes Data
  - Sends operation to fastest node
  - Bounded network hops for all operations

Simple Programming and Operation Model

- Simple Major + Minor key and Value data structure
- ACID transactions
- Configurable consistency and durability

Easy Management

- Web-based Console and CLI commands
- Manages and Monitors:
  - Topology
  - Load
  - Performance
  - Events
  - Alerts

Agenda

- Big Data Overview
- Oracle and NoSQL
- Oracle NoSQL Database
  - Architecture
  - Technical Overview
  - Benchmark Results

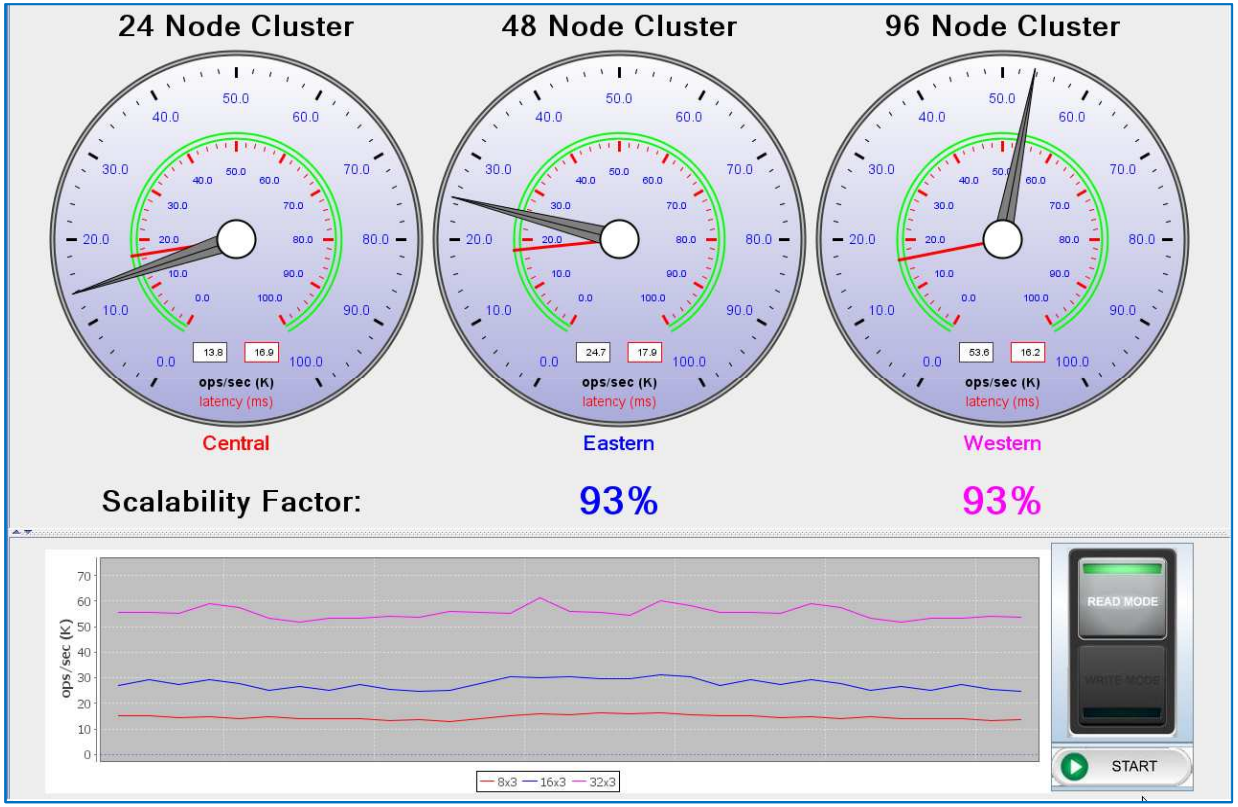
What We've Been Testing

Benchmarking

- YCSB-based QA/benchmarking
  - Key ~= 10 bytes, Data = 1108 bytes
- Configurations of 10-200 nodes
  - Typical Replication Factor of 3 (master + 2 replicas)
  - 100m to 2.1b records, 100m – 400m records per storage group (100 Gbyte – 400 Gbyte per storage group)
- Minimal I/O overhead
  - B+Tree fits in memory => one I/O per record read
  - Writes are buffered + log structured storage system == fast write throughput

Benchmarking

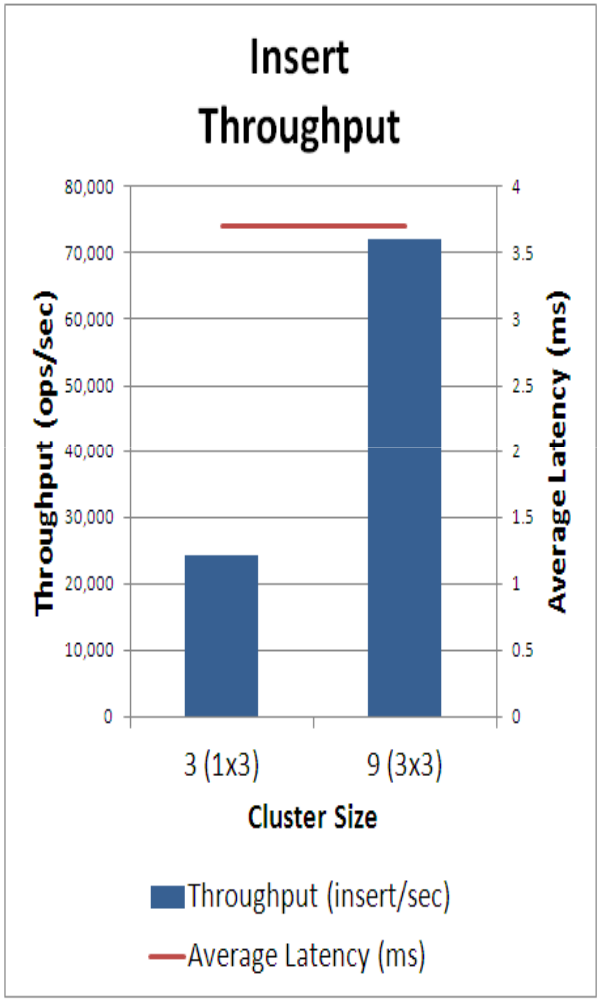
Scalability @ Intel

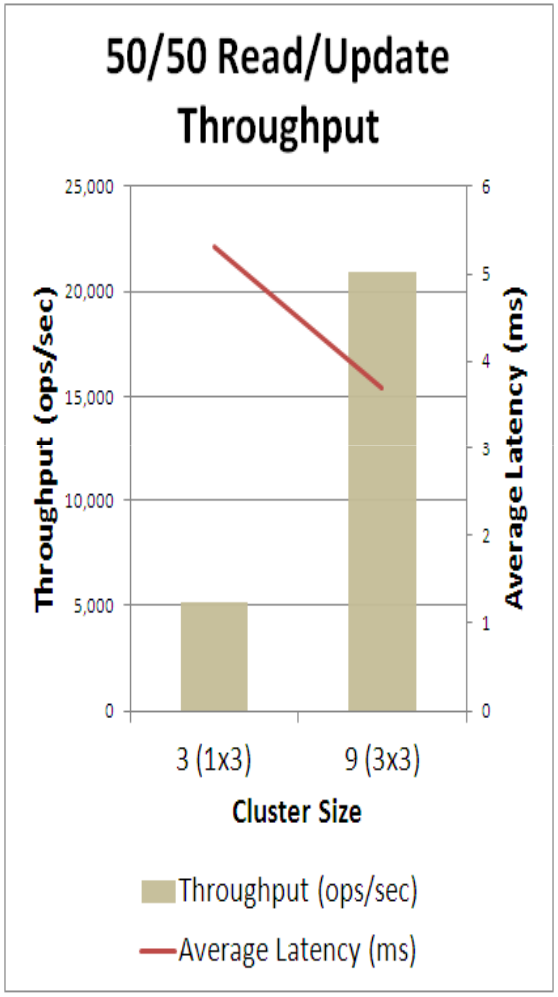


- ~93% of linear scalability
- Response time remains constant

Benchmarking

Results @ Cisco





- 1.2 billion records
- 72K insert/sec
- 21K read/update/sec
- Constant latency
- Linear scalability

Questions