

ORACLE®

# Oracle NoSQL Database

## Product Overview

Robert Greene  
Product Manager / Strategist  
Nov 19, 2014

## Safe Harbor Statement

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

- ▶ **Oracle NoSQL Database Overview**
- ▶ Use cases and customer highlight
- ▶ Product features, Performance, API walk thru
- ▶ Costs and Roadmap

# What is Oracle NoSQL Database

less is more

Simple



Fast



Flexible

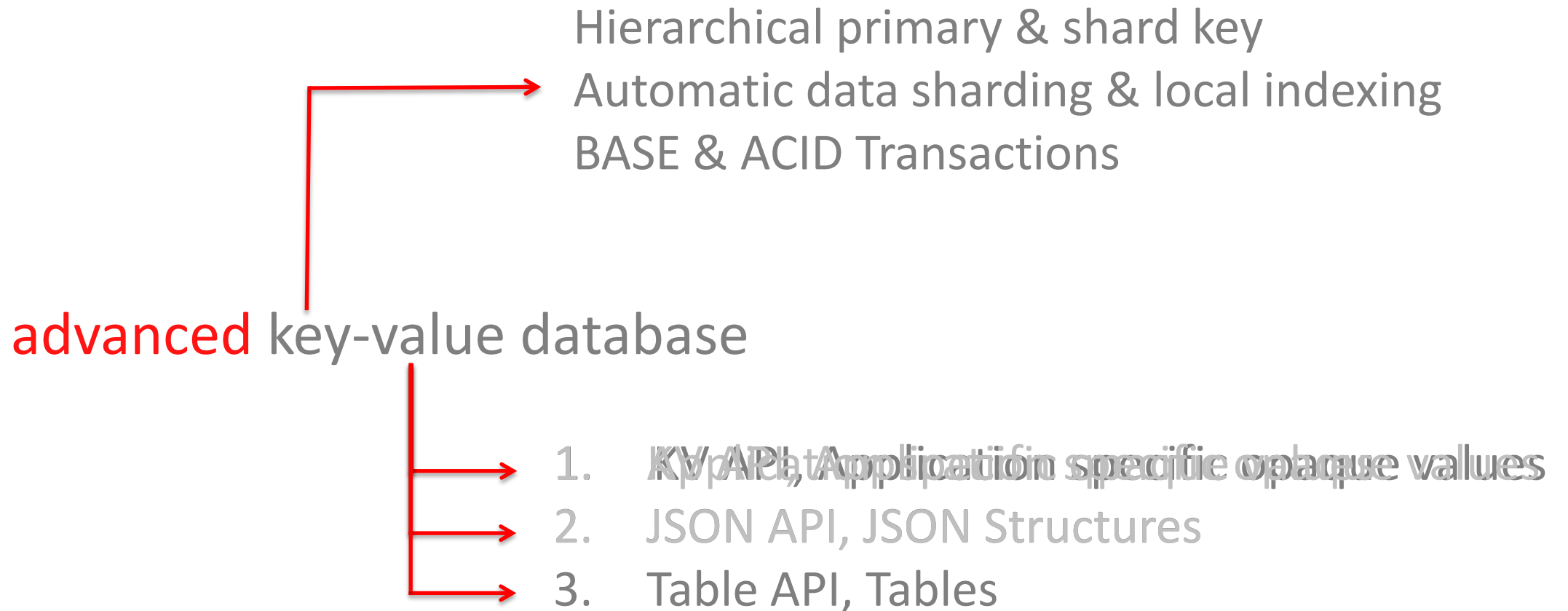


Reliable

**advanced** Key-Value database designed as **cost effective, high performance solution** for **simple operations** on collections of data with **built in high availability** and elastic **scale-out**.

# Oracle NoSQL Database

## Flexible Data Model



# Oracle NoSQL Database

## NoSQL for Developers *and* IT

Simple:

**Setup, Admin, API & Integration**

Fast:

**Parallel Access & Scale-out**

Flexible:

**Flexible schema & Agile development**

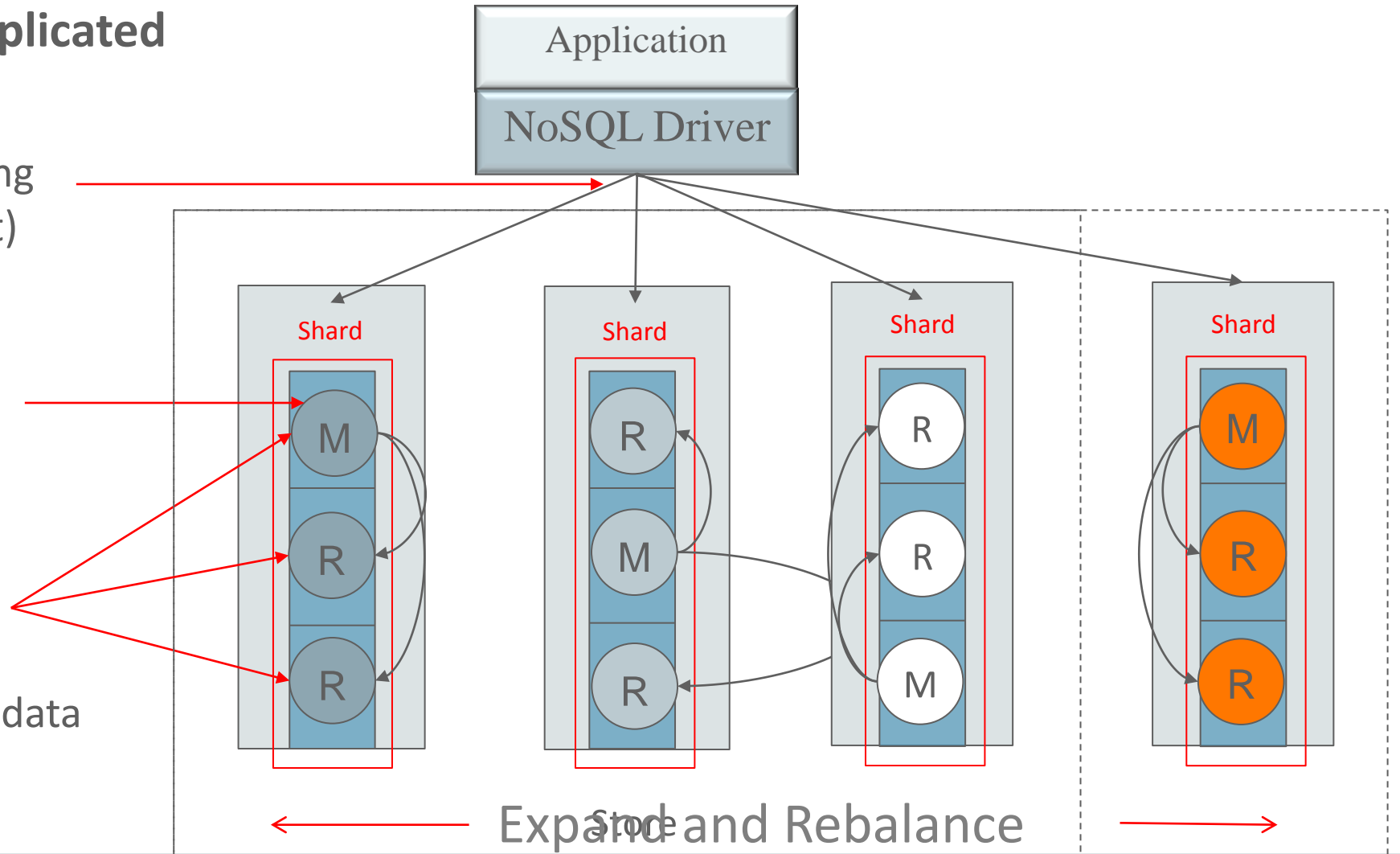
Reliable:

**Built-in HA, Predictable Performance**

# Logical Architecture

## Linear scaling and replicated

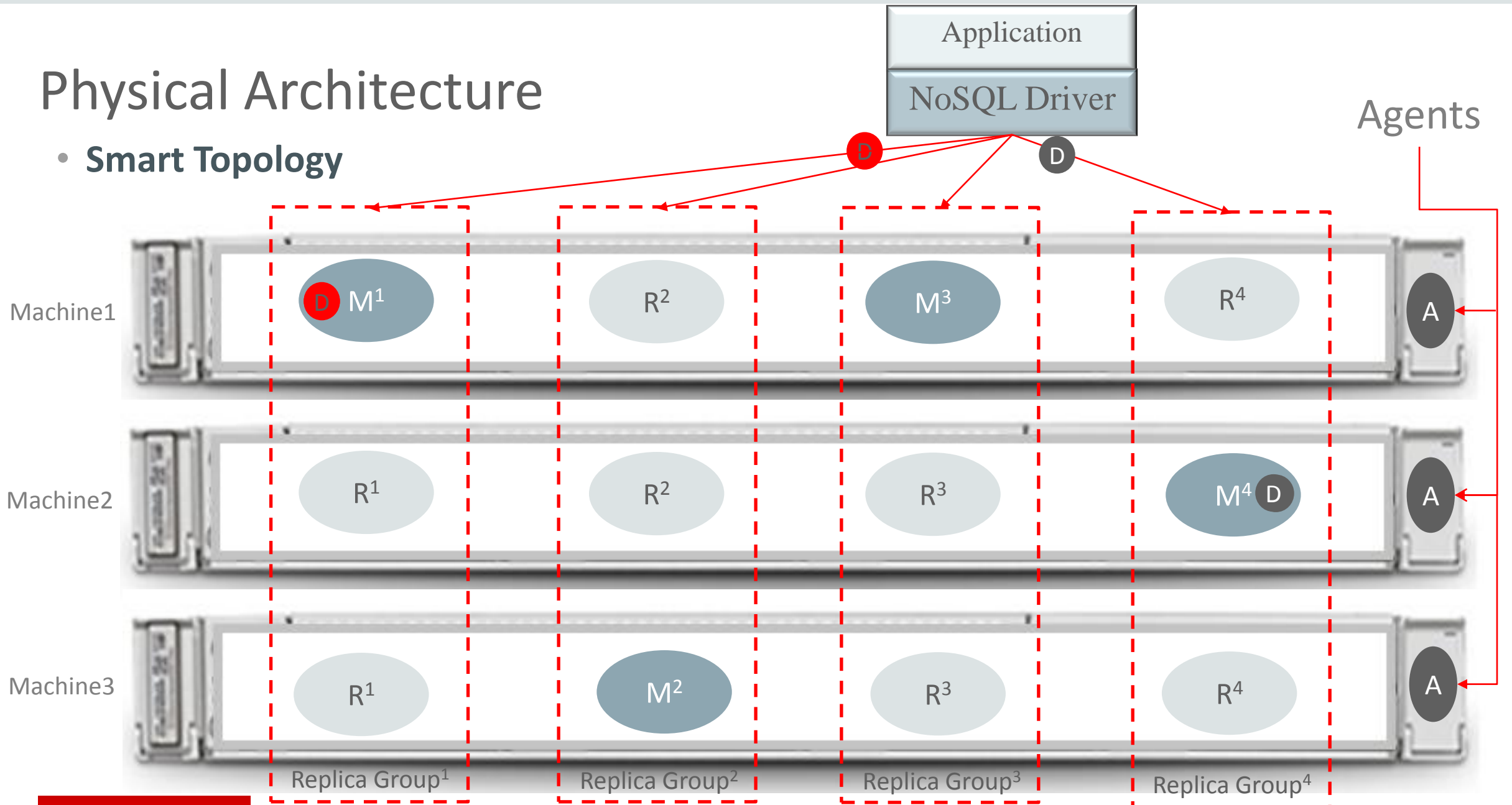
- Elastic Auto Sharding (split, add, contract)
- Writes to elected node
- Reads from any node in system
- Auto re-balance of data on expansion





# Physical Architecture

- Smart Topology



# Oracle NoSQL Database Releases

2011

## Release 1.x

- Major/Minor Key-Value storage
- Java API
- ACID & BASE Transactions
- Replication
- Distributed Data & Queries
- Web Admin Console
- Admin CLI

## Integration

- Hadoop KVInputFormat
- Oracle Loader for Hadoop

2012

## Release 2.0

- JSON schemas & AVRO API
- Streaming Large Object API
- C API
- NoSQL Cluster Elasticity
- Smart Topology

## Integration

- Oracle Database
- Oracle Event Processing
- JMX & SNMP

2013

## Release 2.1 & 2.2

- Data Centers
- Online Rolling Upgrades
- Index Views
- Parallel Scans
- Data CLI
- BDA as NoSQL appliance
- Community Edition Support

## Integration

- Oracle Coherence
- Oracle RDF Graph

2014

## Release 3.0 & 3.1

- Unified CLI
- Secondary Data Centers
- Table Data Model
- Secondary Indices
- Encryption
- Authentication
- Authorization
- Client-side DDL API
- Thrift-based C API
- REST API

## Integration

- Oracle Enterprise Manager
- Oracle Big Data SQL

# Agenda

- ▶ Oracle NoSQL Database Overview
- ▶ **Use cases and customer highlight**
- ▶ Product features, Performance, API walk thru
- ▶ Costs and Roadmap

# Oracle NoSQL Database Use Case Summary

## Web-Scale Personalization & Transaction Processing

Guaranteed low latency lookups and data capture

Advertising, Product Recommendations, Online Catalogs, Social Media, Profile Management, Personalization Web browsing, Shopping Carts

## Real-Time Event Processing

Real-time events trigger rules that perform low latency lookups for context

Fraud Detection, Risk Analysis, Financial Trading, Medical Monitoring, Factory Automation, Utilities Mgmt, Geo-location based content

## Time Series & Sensor Data Management

Capture and management of large volumes of regular and irregular time-based information

Stock Quotes, Manufacturing Sensor Data Mgmt (QC), Testing Data, Utilities, HW/SW Performance

# New Recommendation Platform



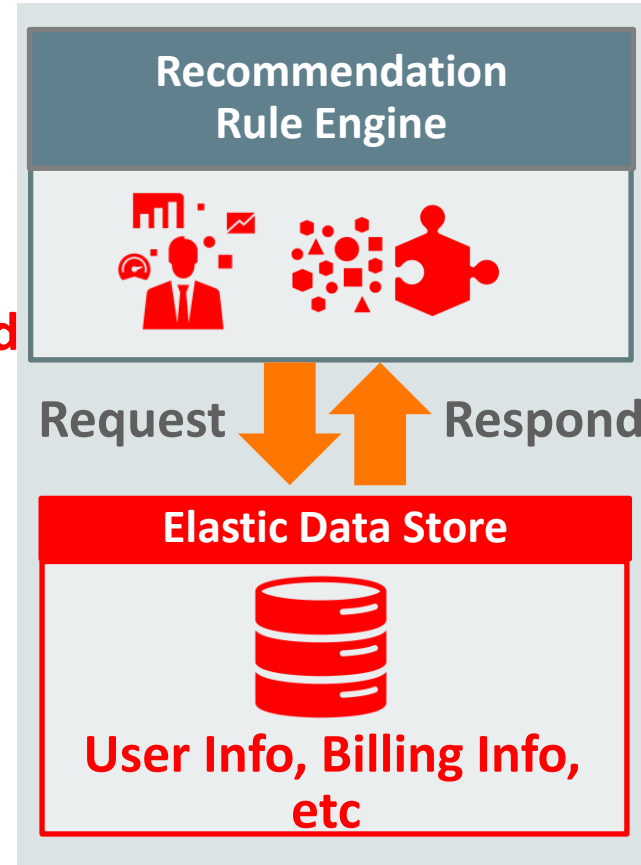
Login



Corporate Web Page

Recommend

User ID



Recommendation Platform

Rules



Customer Info



# Technical Challenges and requirements

Scalability	<ul style="list-style-type: none"> <li>• <b>Scale-out capability</b> by adding new nodes, when new recommendation services are added.</li> </ul>
Flexibility	<ul style="list-style-type: none"> <li>• <b>No additional coding is required</b>, when new user information attributes are added.</li> </ul>
Performance	<ul style="list-style-type: none"> <li>• Recommendation for <b>20 million users</b> using <b>350</b> user information attributes.</li> <li>• <b>Performance capability</b> to handle sharply rising requests.</li> </ul>
Support	<ul style="list-style-type: none"> <li>• <b>Enterprise-grade Support</b></li> <li>• <b>Clear Product Roadmap</b></li> </ul>



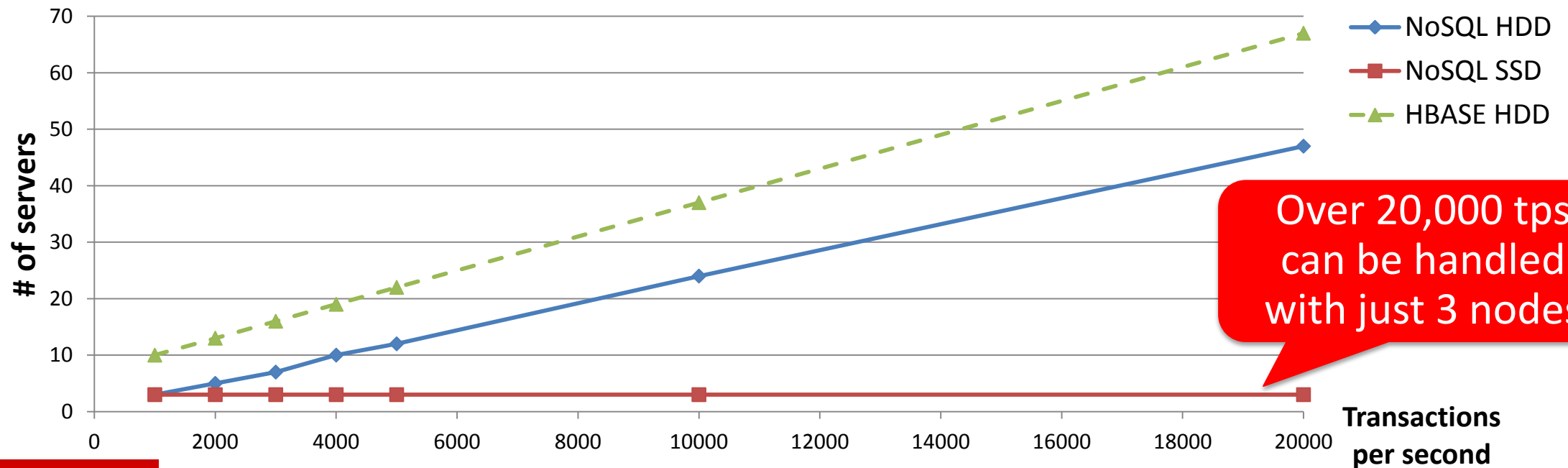
Key Value Store



Oracle  
NoSQL Database

# Performance Comparison

1. Oracle NoSQL Database with SSDs achieves **26000 tps** with **3 nodes**.
2. Compared to HBase, Oracle NoSQL Database achieves **high performance goals with much smaller number of nodes**.
3. Considering high performance goals combined with additional future requirements, **the total cost of Oracle NoSQL Database is estimated to be much lower** than HBase.



# Agenda

- ▶ Oracle NoSQL Database Overview
- ▶ Use cases and customer highlight
- ▶ **Product features, Performance, API walk thru**
- ▶ Costs and Roadmap



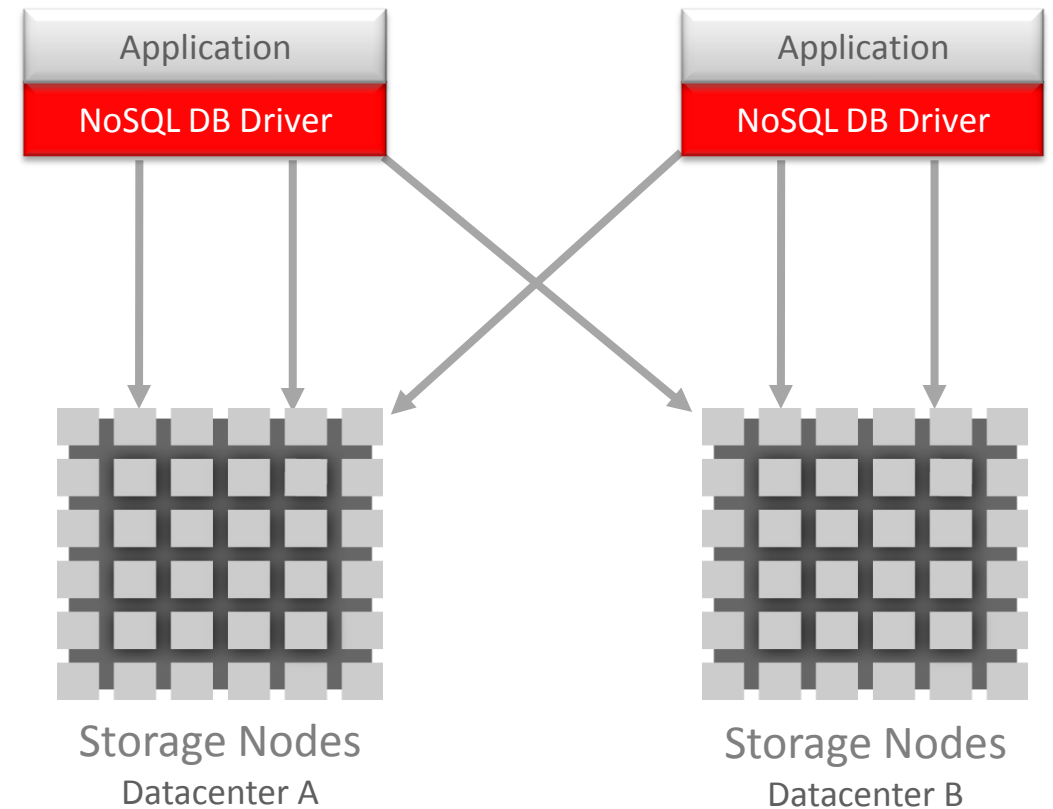
# Enterprise Ready

- **Features**

- Elastic
- BASE Operations
- Tables / JSON / Binary
- Online management
- Data Center Support
- Secondary Indexes
- Secure Access
- Flexible schema

- **Differentiators**

- ACID transactions
- Online rolling upgrades
- Streaming large object support
- Oracle technology integrated
- Engineered Systems and Commodity HW



# Enterprise ready -- Integrated out of the box

- Query NoSQL data from **Oracle Database**
- Access NoSQL data from **Hadoop** for DW and analytics
- Share data with **Oracle Coherence** for extensible in-memory cache grid
- Persist history & event streams for processing with **Oracle Event Processing**
- Store & query RDF data using **Oracle RDF** for NoSQL
- Replicate changes in Oracle Database to NoSQL DB using **Oracle Golden Gate**
- Monitor your NoSQL cluster using **Oracle Enterprise Manager**



**ORACLE**  
DATABASE

**ORACLE**  
FUSION MIDDLEWARE  
COHERENCE

**ORACLE**  
NOSQL DATABASE

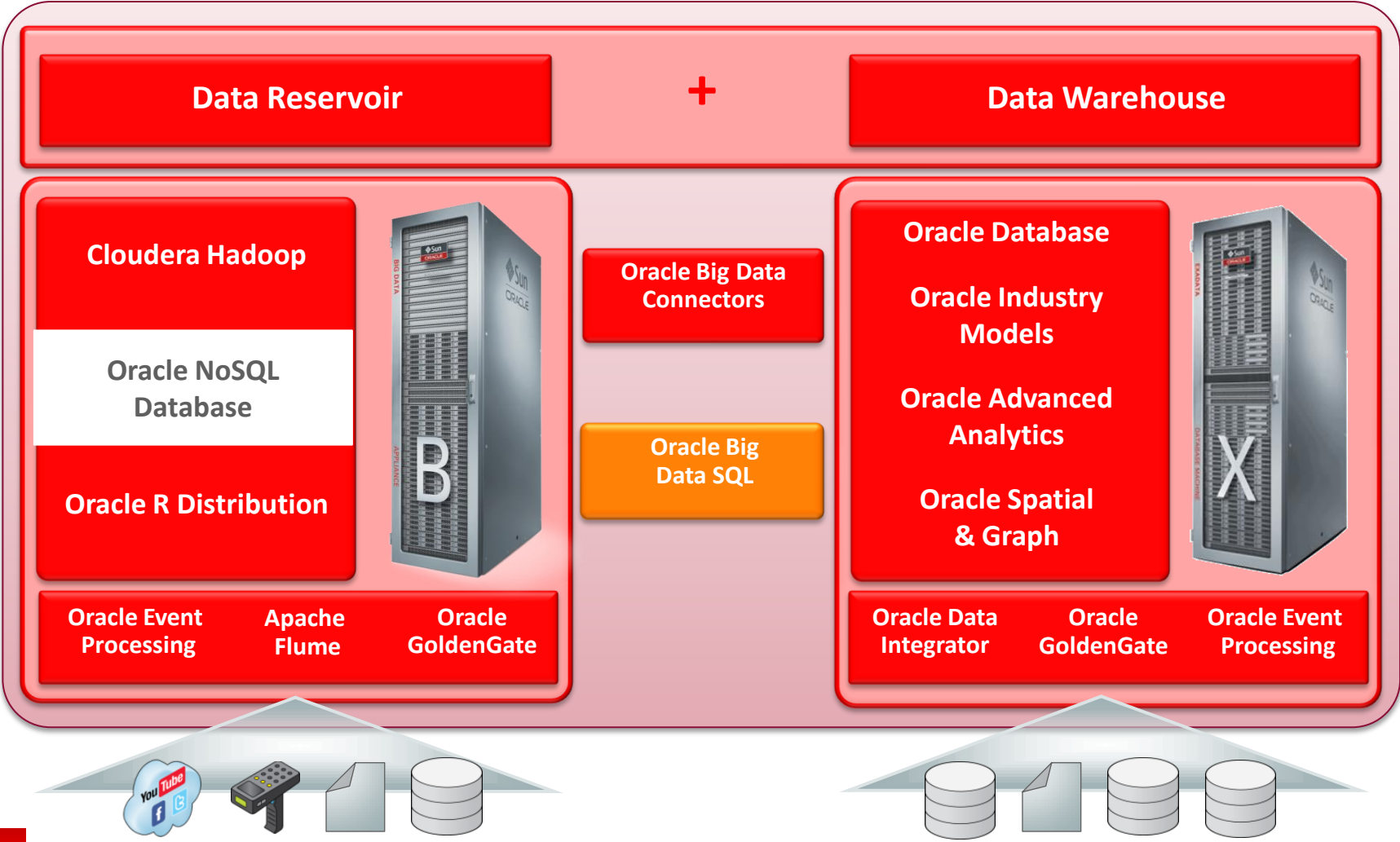
**ORACLE**  
FUSION MIDDLEWARE  
EVENT PROCESSING



**ORACLE**  
ENTERPRISE MANAGER

**ORACLE**  
FUSION MIDDLEWARE  
GOLDENGATE

# Oracle Big Data Management System



# Developer and Admin Tools

- **Standards based tooling**
  - SNMP / JMX metrics
  - **O**racle **E**nterprise **M**anager
  - Cloud ready HTML5 browser admin
  - Command line interface
    - Scripting
    - Query prototyping
    - Data load
  - Easy to use developer API
    - Java, C, REST (Javascript / Python FY15)
    - R, JRuby, Jython community drivers

The image displays two screenshots of Oracle Enterprise Manager (OEM) and Oracle NoSQL Database (NSD) interfaces. The top screenshot shows the 'Create a Plan Change Policy Parameters' dialog box with options like 'Choose a Plan Type', 'Deploy a Datacenter', 'Deploy a StorageNode', 'Deploy an Admin Replica', 'Deploy a Store', and 'Change RepNode Parameters'. The bottom screenshot shows the 'Performance Overview for Replication Node rgl-ml' with a line graph and a table of incidents and problems.

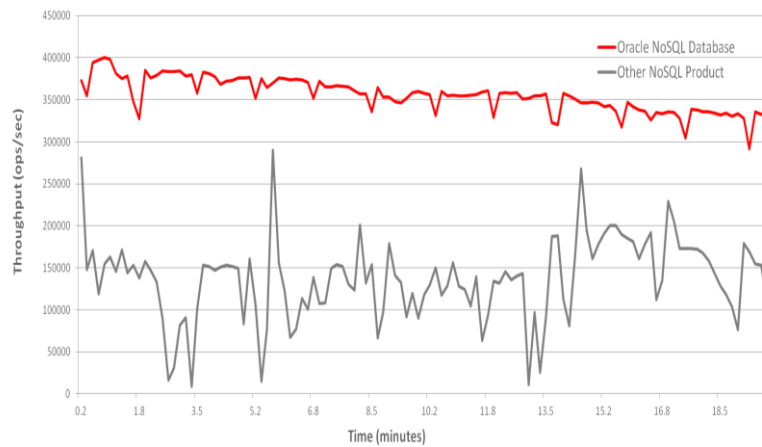
Overlaid on the screenshots are several text labels representing data formats and operations:

- multiGet
- putIfPresent GET
- getLOB AVRO
- JSON delete
- putIfAbsent PUT
- cli commands

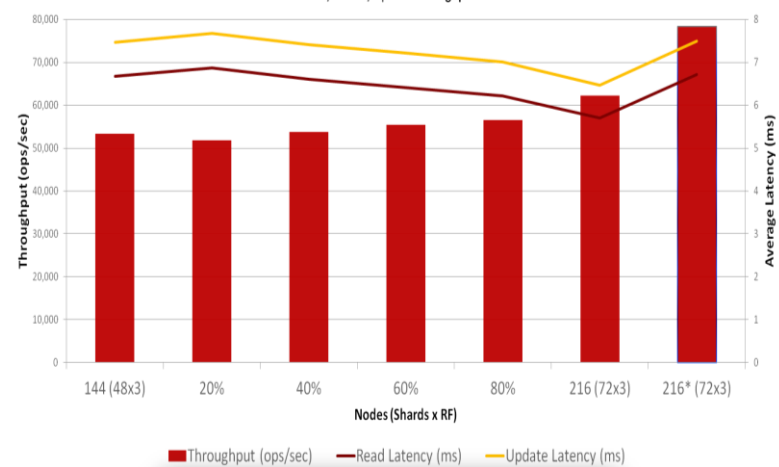


# Predictability, Reliability & Support

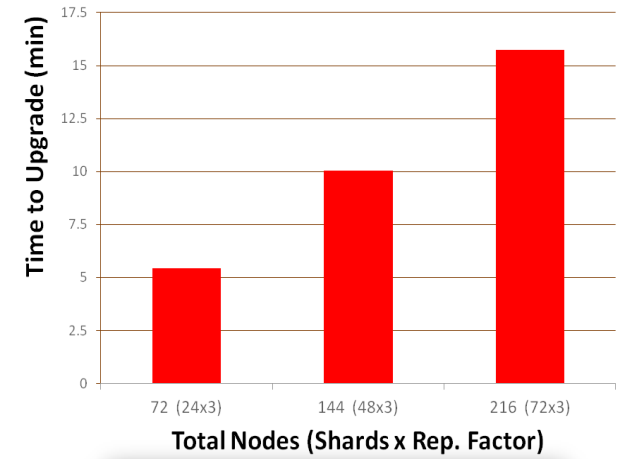
- Global, mission-critical application deployment experience
- Decades of enterprise-grade non-relational database technology
- Oracle Support available for both Enterprise **and** Community Edition
- Designed for Predictability and Manageability



**Bulk Insert Test**



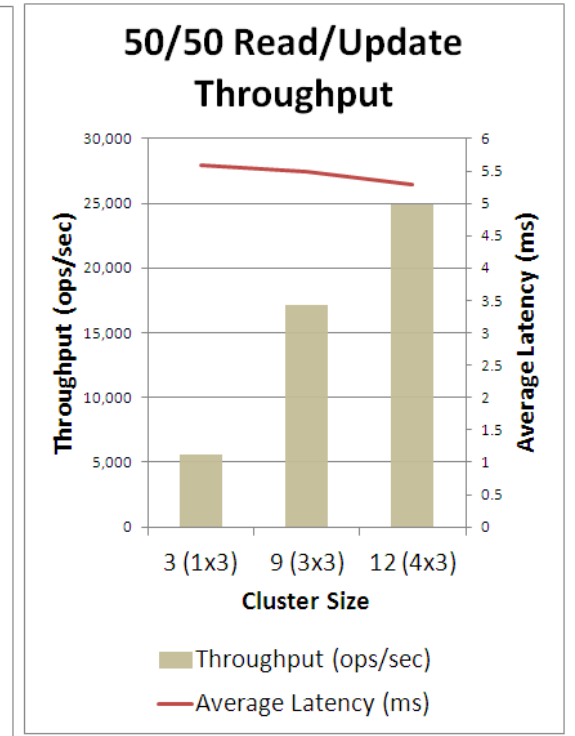
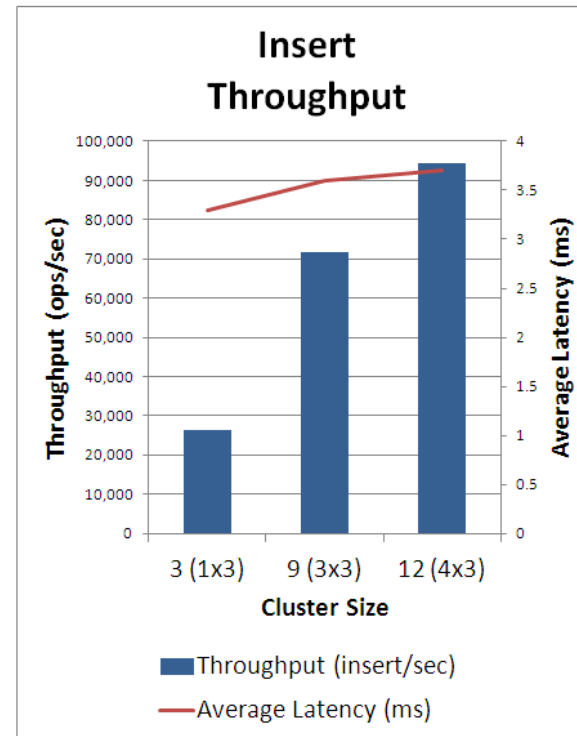
**Cluster Expansion Test**



**Rolling Upgrade Test**

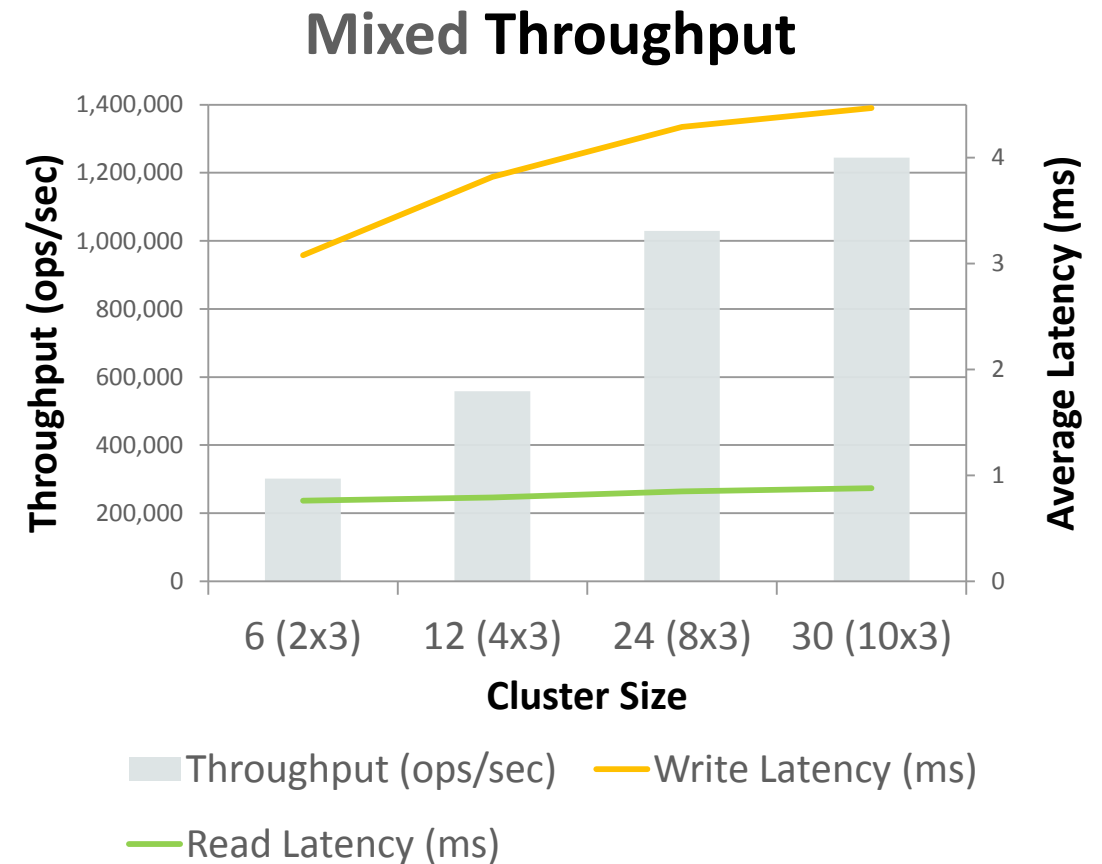
# YCSB on commodity servers

- **What's the big deal**
  - 1.6 billion records
  - 94K insert/sec
  - 25K read/update/sec
  - Low latency
  - Linear scalability



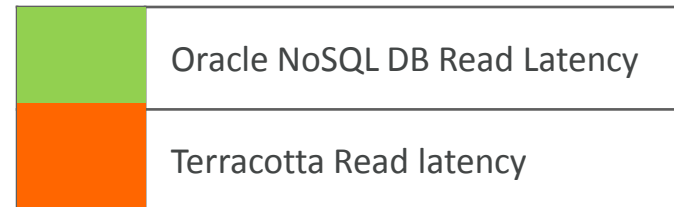
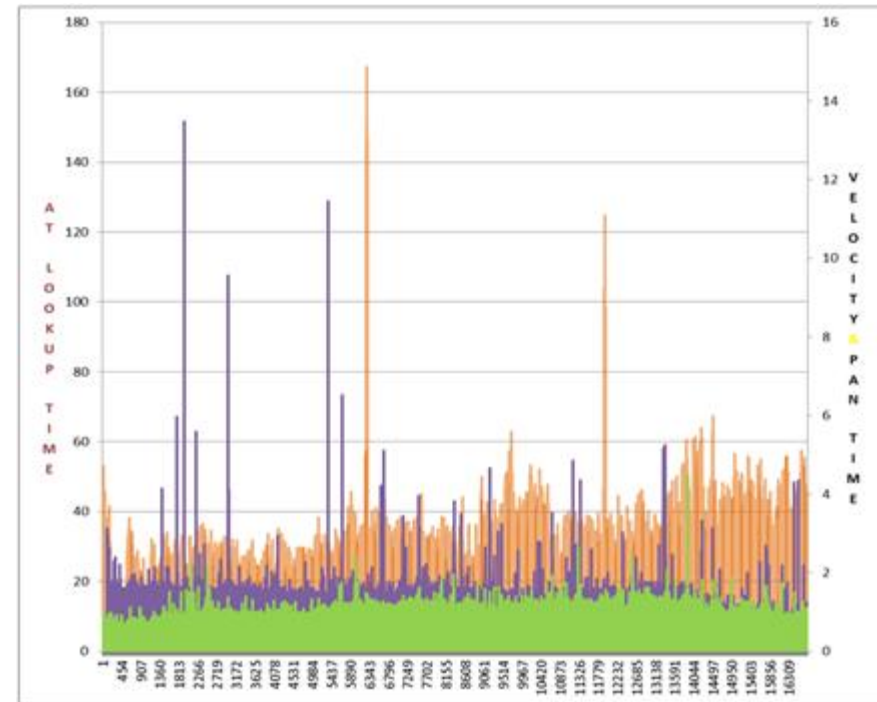
# YCSB on SSD-backed commodity servers

- **What's the big deal**
  - Twitter sees ~500M tweets/day
    - This is 75M a minute
    - Capture all tweets with 3 commodity servers
  - 1.25M ops/sec
  - 2 billion records
  - 2 TB of data
  - 95% read, 5% update
  - Low latency, High Scalability



# Customer benchmarking on SSD

- Oracle NoSQL beats in-memory grid
  - 10's of thousands of txn / sec
  - Lower latency than grid cache
  - Persistent storage
  - Predictable SLA's





# Developer Centric APIs

- Language specific client drivers
  - Java, C, \*javascript, python, C#
- REST API available through Oracle Rest Data Services
  - Deployments using Oracle Web Logic Server, Glassfish, Tomcat
  - Directs Web Service calls to NoSQL DB
  - Marshals data returned into JSON format
  - Full CRUD operations across NoSQL Cluster
- Client-side DDL API
  - Create and Alter tables and schemas inside an application

# Simple Data Modeling

## DDL

```
kv-> table create -name User -desc "A sample user table"
User-> add-field -type Integer -name userId
User-> add-field -type String -name firstName
User-> add-field -type String -name lastName
User-> primary-key -name userId
User-> exit
Table User built.
```

```
kv-> plan add-table -name User --wait
kv-> plan add-index -name UserSecondary -table
      User -field firstName --field lastName
```

Create Table with id, firstName, lastName columns.

Add index on firstName, lastName columns.

## API DDL

```
CREATE TABLE complex (
  COMMENT "this comment goes into the table metadata"
  id INTEGER,
  PRIMARY KEY (id), # id is the pk. this comment is just syntax
  nested MAP(RECORD( m MAP(FLOAT), a ARRAY(RECORD(age INTEGER)))),
  address RECORD (street INTEGER, streetName STRING, city STRING, zip INTEGER),
  friends MAP (STRING),
  anArray ARRAY (FLOAT), fixedBinary BINARY(5),
  days ENUM(monday, tuesday, wednesday, thursday, friday, saturday,
            sunday) NOT NULL DEFAULT tuesday )
```

# API CRUD - Write

```
/* Connect to the NoSQL store */
    store = KVStoreFactory.getStore
        (new KVStoreConfig(storeName, hostName + ":" + hostPort));
    tableAPI = store.getTableAPI();

/* Insert row, if it is already in the table */
    Table table = tableAPI.getTable("myTable");
    Row row = table.createRow();
    row.put("userID", 2);
    row.put("firstName", "John");
    row.put("lastName", "Jameson");
    tableAPI.put(row, null (a return row), null(a write option)); .....put, putIfPresent, putIfAbsent, putIfVersion

/* Catch possible Exceptions*/
    Durability, Timeout, IllegalArgument, Fault Exceptions
```

# API CRUD – Keyspace Read

```
/* Read row, Create a primary key and assign the field value */
```

```
    PrimaryKey key = table.createPrimaryKey();
```

```
    key.put("userID", 1);
```

```
    Row row = tableAPI.get(key, new ReadOptions(null, 0, null));
```

..... or use multiGet,

```
    List<Row> myRows = tableAPI.multiGet(key, null, null);
```

```
/* Read row(s), in a shard or in parallel across all shards*/
```

```
    TableIterator<Row> iter = tableAPI.tableIterator(key, null ( a multiRow options) , null (iterator options));
```

```
    FieldRange fh = table.createFieldRange("familiarName");
```

```
    fh.setStart("Bob", true);
```

```
    fh.setEnd("Patricia", true);
```

```
    MultiRowOptions mro = fh.createMultiRowOptions();
```

```
/* Catch possible Exceptions*/
```

```
    Consistency, Timeout, IllegalArgument, Fault Exceptions
```

# API CRUD – Index Read

- Search secondary index where firstName = “Jane”

1. Create instance of table object we wish to read from
2. Create instance of index object that we will search
3. Set search key
4. Call iterator to scan index
5. Convert results to JSON object, take JSON object and convert back to Row

```
Table userTable = tableAPI.getTable("User", null);
Index index = userTable.getIndex("UserSecondary");
```

```
IndexKey indexKey = index.createIndexKey().put("firstName", "Jane");
TableIterator<Row> results = tableAPI.tableIterator(indexKey);
```

```
while (results.hasNext()) {
    Row row = results.next();
    /* Convert the row to a JSON object */
    String jsonString = row.toJsonString();

    /* Convert the JSON object back to a row */
    Row myRow = userTable.createRowFromJSON(jsonString);
}
```

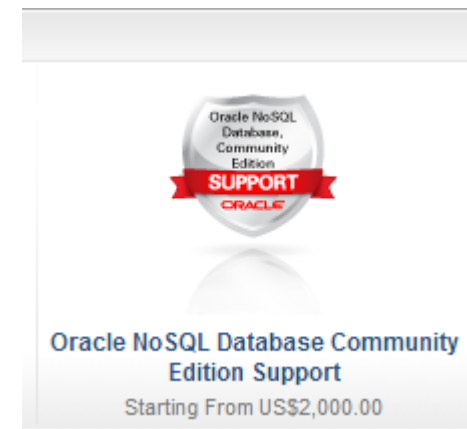
# Agenda

- ▶ Oracle NoSQL Database Overview
- ▶ Use cases and customer highlight
- ▶ Product features, Performance, API walk thru
- ▶ **Costs and Roadmap**

# Oracle NoSQL Database Subscription Model

- **Business-friendly support service**
  - Oracle NoSQL Database Community Edition
    - Open Source AGPL Edition
  - NoSQL Database Client Drivers
    - Open Source, Apache licensed
  - Subscription support for Community Edition
    - Price is \$2,000/year per server
    - No upfront license fee
    - Provides for full Oracle support policy response
    - Purchase online via the Oracle Store
  - Enterprise Edition
    - Closed Source. Standard Oracle License

**ORACLE®**  
**SUPPORT**



<https://shop.oracle.com/>

# Roadmap plan

Dec '14

2015

## Release 3.1 & 3.2

- Unified CLI
- Authorization
- Client-side DDL API
- Thrift-based C API
- Javascript, Python
- REST API

## Integration

- Oracle Enterprise Manager
- Oracle Big Data SQL

## APIs

- C#, Ruby
- SQL query language

## Server & Performance

- Full Text Indexing
- Admin Repair
- Diagnostic enhancements

## Security

- User defined Roles
- Authorization
- Kerberos Integration
- On-disk Encryption

## Administration

- Replication Factor of 2
- Improve admin Scriptability
- Batch Imp/Exp utility
- Supportability & Debug enhancements

## Integration

- More OEM Integration
- Oracle Distributed Query Integration
- Oracle Public Cloud
- KVOutput format
- Yarn Integration
- Spark Integration
- Docker Integration

The current plan is for quarterly releases



# Join NoSQL Database Community

Oracle.com/BigData



**Twitter**

<https://twitter.com/#!/OracleNoSQL>



**LinkedIn**

<http://www.linkedin.com/groups?gid=4147754>



**Oracle's NoSQL DB blog**

<https://blogs.oracle.com/nosql>



**Oracle Technology Network**

<http://bit.ly/1f0d8wU>



**Developer Webcast Series**

<http://bit.ly/1doV2jl>



A woman with long brown hair and glasses is sitting at a wooden table in a bright, modern office or cafe. She is wearing a brown leather jacket over a blue patterned scarf. She is holding a black smartphone to her ear with her left hand and looking down at a large open book or document on the table with her right hand. In the background, another person is sitting at a table, and there are large windows letting in natural light.

# Q&A

# **Hardware and Software Engineered to Work Together**

ORACLE®

# Appendix

# Data Center Support

- **Availability Zones**
  - Flexible configuration
  - Primary Zones
    - Durability guarantees
    - Low latency writes, HA
  - 2<sup>nd</sup>ary Read-Only Zones
    - Asynchronous replication
    - Analytic workloads
    - Report generation
  - Topology Aware Client Driver
  - Provides business continuity *and* distributed workload management

