

Oracle NoSQL Database vs. Cassandra

ORACLE COMPETITIVE | MAY 2017





Introduction and Overview

Oracle NoSQL Database is a scalable and distributed key-value data store. It leverages Berkeley DB in its storage. It provides highly reliable, flexible and available data management across a configurable set of storage nodes. Oracle NoSQL Database is a Java based key-value store implementation that supports Tables, JSON schema or key-value data models. The implementation uniquely supports a full range of transaction semantics from ACID to relaxed eventually consistent and has built in support for online elasticity. Developers can choose from multiple programming language APIs including REST to store and retrieve data. It is designed to scale to large clusters and geographically distributed data centers for disaster recovery. This is complimented by its secondary index support and the support for full text search. Oracle NoSQL DB offers comprehensive SQL-based declarative query capability. Oracle NoSQL DB provides enterprise-class security. In addition, the NoSQL Database is integrated with Oracle RDBMS, several other Oracle products and related open source technologies like Hadoop / MapReduce. It can be deployed on the Oracle Big Data Appliance which is an engineered system as well as commodity servers. Oracle NoSQL Database Enterprise Edition server is available under commercial license and Oracle NoSQL Database Community Edition server is available under an open source license.

Oracle NoSQL Database is available under Apache 2.0 license. Cassandra is a highly available, distributed database for managing large structured data across many commodity servers. Cassandra is a key-value store that supports a single value abstraction known as table-structure. It configures nodes over a ring based architecture where every node in the system can handle any read-write request, so nodes become coordinators of requests when they do not actually hold the data involved in the requested operation. Cassandra supports drivers for popular programming languages. It supports SQL-like query language called CQL, and the support for secondary index and other indexing patterns enable users to retrieve data effectively. Cassandra can be integrated well with the Hadoop/MapReduce environment. Cassandra Community edition is available under Apache license and it is also available with commercial license as Datastax Enterprise Edition.

Comparison

Feature	Oracle NoSQL Database	Cassandra
Data Model	<p>Oracle NoSQL Database has a flexible key-value data model. It also supports Table and JSON data models. The Table and JSON data models also support secondary indices and schema evolution.</p> <p>Data Models</p>	<p>Cassandra provides a Table based data model where rows are partitioned. The first component of a table's primary key is the partition key; within a partition, rows are clustered by the remaining columns of the key. Other columns can be indexed separately from the primary key.</p>
Storage Model	<p>Oracle NoSQL Database leverages BerkeleyDB as a storage engine for each node. BerkeleyDB is a log structured implementation proven in millions of deployments. It is an append only implementation that enables efficient and extremely high write throughput.</p>	<p>Cassandra uses log-structured engine that uses sequential IO to update data. During writes, Cassandra stores the data in memory, and appends the same to the commit log on disk. The data stored in memory is eventually flushed to the disk.</p>
Data Access and APIs	<p>Oracle NoSQL Database has client library API's for Java, Python, Node.js, C/C++ and REST. Oracle NoSQL DB also supports parallelized bulk inserts and bulk retrieval APIs for high performance data access with very large data-sets.</p> <p>Client APIs</p>	<p>Data in Cassandra can be accessed using CQL – the query shell and also multiple client drivers like – Java, Python, C#, Ruby, Node.js, PHP, C++, Apache Spark. Cassandra Bulk loader (SSTableLoader) provides the ability to bulk load external data into the cluster.</p>
Query	<p>Oracle NoSQL Database provides key based access methods (put, get, delete) including multi-key variations with large result set streaming support. Oracle NoSQL DB provides support for SQL for read access via shell as well as programmatic API. The data can also be retrieved using secondary indices. The Oracle NoSQL DB data can also be accessed from Oracle relational database or Hive using SQL queries. This is further optimized and</p>	<p>The Cassandra Query Language (CQL) is the primary language for communicating with the Cassandra database. Users can interact with Cassandra using the CQL shell, cqlsh. With cqlsh, users can create keyspaces and tables, insert and query tables, Nested user-defined types and indexes. Syntactically the Query language is very much similar to standard RDBMS SQL. Cassandra has support for hash based</p>



	<p>parallelized across the cluster to achieve improved query speed and overall better performance.</p> <p>SQL Support</p> <p>Query Key/Value API</p> <p>Oracle RDBMS integration</p>	<p>secondary index.</p>
Transactions	<p>Oracle NoSQL DB support ACID transactions over multiple records that share the same shard key. It provides choice over for consistency and durability on a per-operation basis. This allows developers to make appropriate tradeoff between performance and durability/ consistency.</p> <p>Consistency Explained</p>	<p>Cassandra does not support ACID transactions but offers atomic, isolated, and durable transactions with eventual/tunable consistency. Cassandra trades transactional isolation and atomicity for high availability and fast write performance. Writes in Cassandra are durable but consistent reads can be expensive, since it must contact several replicas to find the “latest” version of the data.</p>
Concurrency	<p>Oracle NoSQL DB achieves concurrency with fine grained locking. The locking is performed at database record level.</p> <p>Concurrent Processing</p>	<p>Cassandra does not implement any locking mechanism. It relies on Coordinator nodes to serve concurrent requests.</p>
Replication	<p>Oracle NoSQL DB supports Replication for both availability and scalability with auto failover. The client driver is topology aware that hashes insert operations to elected master in the replica group and the data is replicated to replica nodes. For every write operation, the user can choose the number of replicas to be updated before responding to client and whether the data should be written to the memory or disk on each node. Reads can be serviced from any node in a replication group. To maintain write availability when replication factor is two, Oracle NoSQL DB provides support for arbiter node. The arbiter does not</p>	<p>Cassandra is designed as a peer-to-peer system that makes copies of the data and distributes the copies among nodes in a group. The replication uses the ring to determine nodes that would hold copies of data. Number of replicas of data is configurable with each keyspace can have independent replication factor. Cassandra supports 2 replication strategies. SimpleStrategy that is the default which blindly writes the data in subsequent nodes along the ring. NetworkTopologyStrategy is primarily useful when deploying to multiple data centers. It ensures that data is replicated across data centers.</p>



	<p>hold any data but holds only the state information and participates in primary election process.</p> <p>Replication Groups</p>	
Sharding/Scalability	<p>Oracle NoSQL DB is sharded (shared-nothing) system that distributes data uniformly and automatically across the cluster based on the hashed value of the primary key. It uses MD5 hashing algorithm over a fixed, highly granular, partition definition. Oracle NoSQL DB supports online elasticity by redistribution of data to newly added hardware.</p> <p>Topology Changes</p>	<p>In Cassandra data distribution and replication go together. Virtual nodes assign the data ownership to physical machines by allowing each node to own a large number of small data range. Cassandra offers the following 3 partitioners: Murmer3 (default), Random and ByteOrder partitioners. Cassandra provides online cluster expansion to achieve scalability and elasticity</p>
Security	<p>Oracle NoSQL DB supports following security features – Authentication and authorization, Audit Logging, Role based access control with support for custom defined roles, SSL Encryption on network data, External Password Storage with Oracle Wallet and password store, Kerberos Support. Data at rest can be encrypted using file-system-based encryption. Oracle NoSQL DB provides security configuration utility to administer security for the cluster.</p> <p>Security Guide</p>	<p>Apache Cassandra provides the following Security features:</p> <ol style="list-style-type: none"> 1. SSL Encryption from Client to Database. 2. Authentication and Authorization DataStax Enterprise Advanced Security provides additional extensions on top of the above- <ol style="list-style-type: none"> 1. Inter-node Encryption 2. Encryption at Rest 3. Data Auditing
Administration	<p>Oracle NoSQL DB is simple and very easy to use. It can be set up to load and query data within just 5 minutes. Oracle NoSQL DB provides tools for capacity planning. Multi-node configuration is made easy with single administration interface. It supports node upgrades and migrations as well as backup and Recovery using file-system-based snapshots. NoSQL DB also supports high speed data export and import.</p>	<p>Cassandra administration requires good understanding of its concepts. The cluster can be setup via a configuration file, hence subsequent configuration changes/ upgrades could become error prone in absence of any admin tool. Datastax Enterprise Edition offers Ops Center tool that provides web based UI for monitoring and administrating Cassandra cluster. Apache Cassandra does not have native</p>

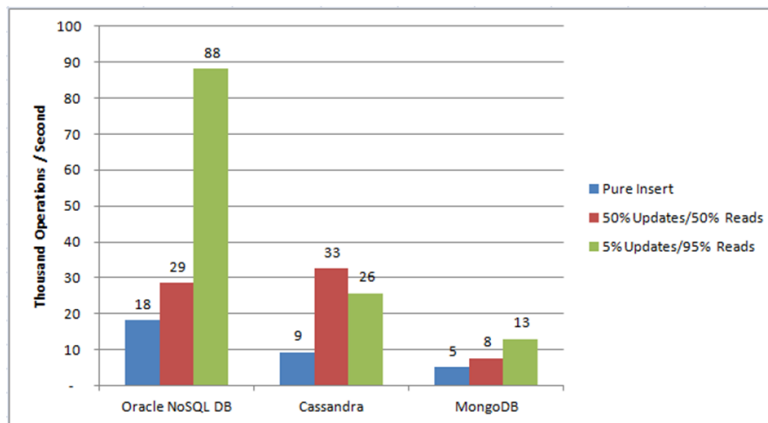


	<p>Oracle NoSQL DB simplifies maintenance with comprehensive support for</p> <p>Rolling Upgrades</p> <p>Administration</p>	<p>support for online rolling upgrades.</p>
Multi DataCenter Awareness	<p>Oracle NoSQL DB supports multiple DataCenters through non-electable replica group strategy. Read requests use local nodes to satisfy latency demands and Write availability is achieved in a local quorum in primary data center and data is replicated to non-electable nodes in secondary data centers</p>	<p>Cassandra allows multiple workloads to be run across multiple datacenters using a snitch. Data can be replicated across the data centers automatically.</p>
Monitoring and Administration	<p>Oracle NoSQL DB provides a variety of protocols for monitoring the cluster. The proprietary protocols are supported in both browser based and CLI interfaces. JMX facilitates integration with monitoring tools like Enterprise Manager, BMC tools etc.</p> <p>Standardized Monitoring</p>	<p>Cassandra provides JMX support for monitoring operations. Cassandra offers command line tools for administration and monitoring. Datastax Enterprise Edition provides Ops Center that is graphical user interface to for monitoring and administrating the cluster.</p>
Hadoop Integration	<p>Oracle NoSQL DB is integrated with and can participate in MapReduce operations from Hadoop environment. Oracle NoSQL DB also supports running Hive queries.</p> <p>Hadoop Integration</p> <p>Hive Integration</p>	<p>Cassandra is integrated with Hadoop environment such that MapReduce jobs can retrieve data from and output to Cassandra. Cassandra includes native support for Apache Pig and Hive.</p>
Large Object Support	<p>Oracle NoSQL DB provides stream API support to read and write Large Objects (LOBs).</p> <p>LOB Support</p>	<p>Cassandra isn't optimized for large file or BLOB storage and single blob values is always read and send to the client entirely. It is advised to manually split large blobs (>16MB) into smaller chunks.</p>
Text Search	<p>Oracle NoSQL DB is integrated with ElasticSearch.</p> <p>Full Text Search</p>	<p>There is no native support for Full Text Search in Apache Cassandra. Datastax Enterprise Edition offers search support to query data using</p>



		complex, sub-string, fuzzy and full text search queries.
Spatial and Graph Data	Oracle NoSQL DB supports graph data management by integrating with Oracle Big Data Spatial and Graph. Together, they can manage networks of linked data as vertices, edges, and properties of vertices that can be used to model, store and analyze relationships found in social networks, cyber security and knowledge networks.	There is no native support for managing Spatial and Graph data in Apache Cassandra. Datastax offers Graph database based on Datastax Enterprise Edition that is inspired by open source Titan graph database.
Oracle Integration	Oracle NoSQL DB is well integrated with Oracle products including: <ul style="list-style-type: none">• Oracle Relational Database• Oracle Coherence• Oracle Database Mobile Server Oracle Enterprise Manager• Oracle Golden Gate• Oracle SQL Developer• Oracle Stream Explorer among others	Cassandra is not integrated with any Oracle products
Time To Live	The Time To Live feature in Oracle NoSQL DB limits the lifetime of data in the store. TTL can be specified in hours and days using the Table API as part of DDL queries. Oracle NoSQL DB supports TTL at the row level.	Cassandra supports data to have an optional expiration period, i.e. Time to Live. It can have a precision of a second.

Performance comparison Oracle NoSQL DB, Cassandra, Mongo DB



Benchmark specifications

- Yahoo! Cloud Serving Benchmark (YCSB)
[13 byte key; 1100 byte value]
- 60 Million records in database (~60 GB of data)
- 3 Node system
(X4170 m2; 2 Xeon(R) X5670
2.93GHz; 72 GB; SSD drives)

ORACLE

Copyright © 2017, Oracle and/or its affiliates. All rights reserved. |





Figure 1 - Oracle NoSQL Database Performance Compared to Cassandra and MongoDB



Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Integrated Cloud Applications & Platform Services

Copyright © 2017, Oracle and/or its affiliates. All rights reserved. This document is provided *for* information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0116

Oracle NoSQL Database vs Cassandra
May 2017
Author: Schulman/Nirhali



Oracle is committed to developing practices and products that help protect the environment

Integrated Cloud Applications & Platform Services