INTRODUCTION

Oracle NoSQL Database Cloud Service is designed for today’s most demanding workloads with high volume, high velocity, and high variety. It meets the modern application requirements for predictable low latencies, flexible data models, elastic capacity scaling, and fast production deployment.

It is a serverless cloud service fully managed by Oracle. Developers focus on application development without dealing with the hassle of managing servers, storage devices, cluster expansion, software installation or updates, and high availability configurations.

To help developer get started right away, Oracle provides a simulated version of NoSQL Database Cloud Service for developers to install on their computers. Applications can be developed and tested by connecting to this localized, simulated Oracle NoSQL Database cloud environment before deploying to Oracle NoSQL Database Cloud Service.

COMPONENTS AND ARCHITECTURE

Oracle NoSQL Cloud Simulator consists of two components:

- Oracle NoSQL Cloud Simulator
- Build/run scripts

Oracle NoSQL Java SDK contains the Java language driver, examples and Javadoc.

Figure 1 shows the relationship between the different components for Java only:

![Oracle NoSQL Cloud Components and the client side drivers](image-url)
Oracle NoSQL Cloud Simulator is not recommended to be used for production environments.

Oracle NoSQL language drivers, for example the Java SDK, contain the APIs which can be used to build an application that can run either against Oracle NoSQL Cloud Simulator or Oracle NoSQL Database Cloud Service.

PERSONAS

Application developers designing cloud applications

- Developers write code that can be built, debugged and tested in their local environments using Oracle NoSQL Cloud Simulator and an Oracle NoSQL language driver, for example Java SDK (referred as the “client driver” in the rest of this document). With minimal changes, the application can migrated to Oracle NoSQL Database Cloud Service.

- Developers use Oracle NoSQL Cloud Simulator to troubleshoot issues locally without impacting their production operations in the cloud.

Oracle NoSQL Cloud Simulator development process

- A developer would download Oracle NoSQL Cloud Simulator and Oracle NoSQL Java SDK (driver) from the Oracle NoSQL Database Cloud download page.

- After downloading and unbundling the .zip or .tar.gz files, developers can quickly start developing for Oracle NoSQL Cloud Simulator:
  - Start and run Oracle NoSQL Cloud Simulator as a background process.
  - Write an application in Java that uses classes from the client driver to connect to Oracle NoSQL Cloud Simulator and perform read/write operations to a local implementation of the data storage.
  - For reference, the Javadoc for the NoSQL client driver API is in the parent directory of the untarred SDK under /doc/javadoc.

Oracle NoSQL Database Cloud Service development process

- After downloading and unbundling the Oracle NoSQL Java SDK .zip or .tar.gz files, developers can begin developing for Oracle NoSQL Database Cloud Service:
  - Unlike an application written for the Cloud Simulator, an application running against Oracle NoSQL Database Cloud Service needs to handle authorization and authentication.
  - Refer to the instructions on the “Connecting Your Application” documentation to establish a secure connection with the cloud service. There are 2 ways: i) connecting using API, ii) connecting using a configuration file.
  - Similar information is available in the quick start tutorial.

For reference, consult the Javadoc for the NoSQL client driver APIs.
DATA FLOWS

Below is a logical diagram of how an application would connect to Oracle NoSQL Cloud Simulator.

Figure 2 - Relationship between components for Oracle NoSQL Cloud Simulator

Once the developer is confident running the application using Oracle NoSQL Cloud Simulator, they can run the application with the addition of authentication credentials/access tokens using Oracle NoSQL Database Cloud Service.

The application logic and code remain the same between using Oracle NoSQL Cloud Simulator and Oracle NoSQL Database Cloud Service.

Below is a logical diagram of how an application would connect to Oracle NoSQL Database cloud.
ORACLE NOSQL CLOUD SIMULATOR

Oracle NoSQL Cloud Simulator allows application developers to start developing Oracle NoSQL Database cloud applications without having to purchase or use any Oracle NoSQL Database cloud credits.

- Developers can take advantage of Oracle NoSQL Cloud Simulator and run the storage service as a background process in a developer's local environment.
- Oracle NoSQL Cloud Simulator supports all the client APIs that are required to communicate with Oracle NoSQL Database Cloud Service.
- Developers can design, build, and test all functionality using the Oracle NoSQL Cloud Simulator utility before running the application against Oracle NoSQL Database Cloud Service.

Assumptions and Requirements

- Oracle NoSQL Cloud Simulator can be used on standalone/local development environments on Windows, Mac, and Linux platforms for development and test.
- Developers should have Java Developer Kit (JDK) Version 10 or higher installed on their client systems.
- If there are questions regarding the API for Oracle NoSQL Cloud Simulator, send an email to:

  oraclenosql-info_ww@oracle.com with Oracle NoSQL Cloud Simulator in the subject line.
ORACLE NOSQL CLOUD SIMULATOR COMPARED TO ORACLE NOSQL DATABASE CLOUD SERVICE

The server instance created in Oracle NoSQL Cloud Simulator supports relatively limited aggregate throughput. The operations depend on the speed and capability of the machine running the Cloud Simulator. By comparison Oracle Database NoSQL Cloud Service includes scalability, availability, and durability features which make it ideal for production use.

- Oracle NoSQL Cloud Simulator is for development and testing purposes.
- Multiple instances of Oracle NoSQL Cloud Simulator must not be started against the same root directory. It assumes exclusive control over the data storage.
- Oracle NoSQL Cloud Simulator doesn’t support or require security-relevant configuration.
- No hard limits are enforced for number of tables, size of tables, number of indexes or maximum throughput specified for tables.
- By default, read operation consistency is *Eventual*, but might seem strongly consistent since the server runs locally.
- If *-throttle* option is not enabled some data operations may still appear throttled based on the limits provisioned for a table. For development purposes, it is recommended not to enable *–throttle* and instead assign very high table limits. Enable the *–throttle* option only if this type of testing is necessary for your application.
- DDL Operations such as table create/drop, index create/drop, are not throttled.
- Oracle NoSQL Cloud Simulator does not maintain an operation history.