**BỘ TÀI NGUYÊN VÀ MÔI TRƯỜNG**

**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG TP. HCM**

**KHOA HỆ THỐNG THÔNG TIN & VIỄN THÁM**

----------🕮 🙞🙜----------



**BÁO CÁO ĐỒ ÁN MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**ORACLE NOSQL DATABASE**

Thành viên của nhóm:

1. Nguyễn Nhật Minh ( 0950080131)
2. Nguyễn Minh Thuận (0950080141)
3. Trương Thanh Phúc (0950080136)

Lớp : 09\_ĐH\_CNTT4

Thành phố Hồ Chí Minh, tháng 11, năm 2022

LỜI MỞ ĐẦU

Trong thời đại internet hiện nay, mạng internet có mặt ở khắp nơi, hàng tỉ thiết bị trên toàn cầu đang được kết nối với nhau. Năm 2015, con số thiêt bị được ước tính khoảng 14,4 tỉ. Mỗi thiết bị đều có một nhu cầu trao đôi dữ liệu nhất định qua internet đến các server hay đến các “đám mây” dữ liệu. Với con số 14,4 tỉ thiết bị, cùng với thời gian thì lượng lưu trữ ngày càng lớn. Theo ước tính của IDC thì năm 2020 chúng ta sẽ có 4 tỉ người kết nối với nhau, với hơn 25 triệu ứng dụng internet cùng 50 nghìn tỉ Gb dữ liệu được lưu trữ trên toàn thế giới. Từ sự phát triển của internet và nhu cầu trao đổi, xử lý dữ liệu ngày càng lớn trong xã hội, chúng ta mới được biết tới rộng rãi hơn về khái niệm “Big Data”. Cùng với “Big Data” thì loại CSDL NoSQL cũng được ứng dụng nhiều hơn ( người ta thường gọi là Not only SQL ). Nhận thấy vấn đề này, Oracle đã ra mắt HQT CSDL NoSQL của mình một phần dựa trên BerkeleyDB vào tháng 10 năm 2011.Chúng ta có thể kể tới các hệ quản trị CSDL như: MySQL, Oracle, SQL Server, PostgreSQL và mỗi loại trên có những tính năng, lợi ích riêng. Và Oracle NoSQL là một trong những hệ quản trị cơ sở dữ liệu mà chúng ta sẽ cùng tìm hiểu trong bài viết này.

Mục lục

[LỜI MỞ ĐẦU 2](#_Toc121869898)

[CHƯƠNG 1: GIỚI THIỆU 1](#_Toc121869899)

[I. Oracle NoSQL Database là gì? 1](#_Toc121869900)

[CHƯƠNG 2: Tổng quan về cơ sở dữ liệu NoSQL 2](#_Toc121869901)

[II. Cơ sở dữ liệu NoSQL là gì? 2](#_Toc121869902)

[III. Ưu điểm 2](#_Toc121869903)

[IV. Nhược điểm 3](#_Toc121869904)

[CHƯƠNG 3: Tổng quan chức năng 4](#_Toc121869905)

[I. Mô hình dữ liệu kiến trúc 4](#_Toc121869906)

[1. Tùy chọn mô hình hóa dữ liệu 4](#_Toc121869907)

[2. Mô hình Sharding 5](#_Toc121869908)

[3. Tính nhất quán và mô hình độ bền 6](#_Toc121869909)

[4. Mô hình lập trình 7](#_Toc121869910)

[5. Mô hình bảng 7](#_Toc121869911)

[6. Truy cập dữ liệu bảng qua SQL 8](#_Toc121869912)

[7. Làm việc với tài liệu JSON 10](#_Toc121869913)

[CHƯƠNG 4: Khả năng phục hồi thất bại 12](#_Toc121869914)

[CHƯƠNG 5: Ngành kiến trúc 13](#_Toc121869915)

[CHƯƠNG 6: Cài đặt 15](#_Toc121869916)

[II. Storage Nodes 15](#_Toc121869917)

[III. Trình điều khiển khách hàng 16](#_Toc121869918)

[CHƯƠNG 7: Bảo vệ 18](#_Toc121869919)

[CHƯƠNG 8: Hội nhập 19](#_Toc121869920)

[CHƯƠNG 9: DEMO 21](#_Toc121869921)

[CHƯƠNG 10: Kết luận 27](#_Toc121869922)

[LỜI CẢM ƠN 28](#_Toc121869923)

CHƯƠNG 1: GIỚI THIỆU

# **Oracle NoSQL Database là gì?**



Oracle NoSQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, với giấy phép GNU Affero General Public. Các bạn có thể tải bộ mã nguồn từ trang chủ của Oracle với bản Community Edition để xem mã nguồn. Oracle NoSQL hỗ trợ đa nền tảng : từ linux, windows cho đến OS x. Oracle NoSQL ban đầu xây dựng dựa trên mô hình cơ sở dữ liệu đơn giản nhất trong NoSQL là key/value : mỗi lưu trữ key/value đều tồn tại cho một key nhất định. Tuy nhiên đến tháng 4/2014 , với phiên bản 3.0 thì Oracle NoSQL đã mở rộng mô hình dữ liệu key/value stores đến mô hình dữ liệu hướng tài liệu document-oreinted.

- Oracle NoSQL hỗ trợ giao dịch với các tính chất Automicity – nguyên tố, Consistency – nhất quán, Isolation – tách biệt, Durability – bền vững, hỗ trợ bảng và tất nhiên như nhiều CSDL NoSQL khác, Oracle NoSQL hỗ trợ cấu trúc kiểu dữ liệu JSON.

- Oracle NoSQL tiến hành mã hóa SSL các phiên và cổng mạng bảo vệ hạn chế cho sự xâm nhập mạng và tích hợp với các sản phẩm Oracle Wallet cho phép bảo vệ khỏi những truy cập trái phép vào dữ liệu nhạy cảm.

- Do dự bùng nổ của Big Data, Oracle đã xây dựng một hệ thống gọi là Big Data Appliance, trong đó Oracle NoSQL là một thành phần không thể thiếu cùng với framework Hadoop trong hệ thống này. Oracle Big Data Appliance tích hợp chặt chẽ với Oracle Exadata thông qua sử dụng Big Data Connector, cho phép kết nối phân tích các dữ liệu trong các cơ sở dữ liệu truyền thống như Oracle Database.

- Các nhà phát triển có thể tạo ra các ứng dụng sáng tạo bằng cách sử dụng một số ngôn ngữ lập trình phổ biến và có thể lập mô hình dữ liệu theo một số cách khác nhau.

CHƯƠNG 2: Tổng quan về cơ sở dữ liệu NoSQL

# **Cơ sở dữ liệu NoSQL là gì?**

Cơ sở dữ liệu NoSQL đại diện cho một sự phát triển gần đây trong kiến trúc ứng dụng doanh nghiệp, tiếp tục sự phát triển của hai mươi năm qua. Vào những năm 1990, các ứng dụng tích hợp theo chiều dọc đã nhường chỗ cho kiến trúc máy chủ-máy khách và gần đây hơn, kiến trúc máy khách-máy chủ đã nhường chỗ cho kiến trúc ứng dụng web ba tầng. Song song đó, nhu cầu của các dịch vụ quy mô web đã bổ sung khả năng truy cập thời gian thực có độ trễ rất thấp cũng như xử lý giảm thiểu bản đồ ngoại tuyến vào hỗn hợp. Các kiến trúc sư dữ liệu bắt đầu tránh né tính nhất quán trong giao dịch để đổi lấy khả năng mở rộng gia tăng và phân phối quy mô lớn. Phong trào NoSQL nổi lên từ hệ sinh thái thứ hai này.

NoSQL thường được đặc trưng bởi những gì nó không - tùy thuộc vào người bạn yêu cầu, nó không chỉ là một hệ thống quản lý cơ sở dữ liệu quan hệ dựa trên SQL hoặc đơn giản không phải là một Hệ thống quản lý cơ sở dữ liệu quan hệ dựa trên SQL (RDBMS). Trong khi các định nghĩa đó giải thích NoSQL không phải là gì, chúng không giải thích được NoSQL là gì. Hãy xem xét các nguyên tắc cơ bản đã hướng dẫn quản lý dữ liệu trong bốn mươi năm qua. Hệ thống RDBMS và quản lý dữ liệu quy mô lớn được đặc trưng bởi các thuộc tính ACID giao dịch của Tính nguyên tử, Tính nhất quán, Tính cách ly và Độ bền. Ngược lại, NoSQL đôi khi được đặc trưng bởi từ viết tắt BASE:

# **Ưu điểm**

Có một số lợi thế, điểm mạnh khi làm việc với cơ sở dữ liệu NoSQL như MongoDB và Cassandra. Những ưu điểm chính của nosql là khả năng mở rộng và tính sẵn sàng cao. - NoSQL giải quyết được các vấn đề dữ liệu lớn(big data) về các hệ thống thông tin hoặc là phân tán dữ liệu. - Việc mở rộng phạm vi là mềm dẻo: NoSQL thay thế câu thần chú cũ của các nhà quản trị CSDL về việc ‘mở rộng phạm vi’ với một thứ mới: ‘mở rộng ra ngoài’. Thay vì phải bổ sung thêm những máy chủ lớn hơn để tải nhiều dữ liệu hơn, thì CSDL NoSQL cho phép một công ty phân tán tải qua nhiều máy chủ khi tải gia tăng. - High availability: Khả năng tự động sao chép trong MongoDB làm cho nó rất tốt trong mọi trường hợp vì trong trường hợp có bất kỳ lỗi nào, dữ liệu sẽ tự động sao chép về trạng thái nhất quán trước đó.

# **Nhược điểm**

Bên cạnh những ưu điểm của nó thì NoSQL Database cũng có những nhược điểm sau: - Quản lý dữ liệu: Mục đích của các công cụ dữ liệu lớn là làm cho việc quản lý một lượng lớn dữ liệu trở nên đơn giản nhất. Nhưng quản lý dữ liệu trong NoSQL phức tạp hơn nhiều so với cơ sở dữ liệu quan hệ. Đặc biệt, NoSQL nổi tiếng là khó cài đặt và thậm chí là để quản lý nó hằng ngày cũng tốn khá nhiều thời gian. - Sao lưu dữ liệu: Sao lưu là một điểm yếu lớn đối với một số cơ sở dữ liệu NoSQL như MongoDB. Nó không có cách tiếp cận để làm sao lưu dữ liệu một cách nhất quán. - Thiếu tính nhất quán: NoSQL đánh đổi sự nhất quán để ưu tiên tốc độ, hiệu suất hiệu quả hơn. - Trọng tâm hẹp: Cơ sở dữ liệu NoSQL có trọng tâm rất hẹp vì nó chủ yếu được thiết kế để lưu trữ nhưng nó cung cấp rất ít chức năng. - Mã nguồn mở: NoSQL là cơ sở dữ liệu mã nguồn mở và không có tiêu chuẩn đáng tin cậy cho NoSQL được nêu ra. - Không có lược đồ: Ngay cả khi bạn lấy dữ liệu ở dạng tự do, bạn hầu như luôn cần áp đặt các ràng buộc để làm cho nó hữu ích. Với NoSQL, trách nhiệm sẽ được chuyển từ cơ sở dữ liệu sang nhà phát triển, lập trình ứng dụng. - Kỹ năng NoSQL: Một hạn chế khác đối với NoSQL là người sử dụng có thể sẽ thiếu các kỹ năng chuyên môn ở mức tương đối vì hệ thống này còn khá mới và không phải ai cũng biết sử dụng nó một cách thành thạo.

CHƯƠNG 3: Tổng quan chức năng

Cơ sở dữ liệu Oracle NoSQL tận dụng công cụ lưu trữ tính khả dụng cao của Cơ sở dữ liệu Oracle Berkeley Java Edition để cung cấp lưu trữ bảng giá trị và khóa phân tán, có tính khả dụng cao cho các ứng dụng hoặc dịch vụ web có khối lượng lớn, nhạy cảm với độ trễ. Trong khi Oracle Berkeley DB cung cấp chức năng quản lý lưu trữ cơ bản, thì các lớp trên của Cơ sở dữ liệu Oracle NoSQL cung cấp các tính năng quan trọng như mở rộng và mở rộng linh hoạt trực tuyến, hỗ trợ cho nhiều mô hình dữ liệu (khóa giá trị, bảng, JSON (Ký hiệu đối tượng JavaScript) tài liệu và đồ thị), Truy vấn SQL, tìm kiếm toàn văn, xác thực, ủy quyền và khôi phục thảm họa nhiều trung tâm dữ liệu.

# **Mô hình dữ liệu kiến trúc**

Tổng quan về Cơ sở dữ liệu Oracle NoSQL về cốt lõi, Cơ sở dữ liệu Oracle NoSQL triển khai bản đồ khóa, giá trị từ các khóa do người dùng xác định đến các mục dữ liệu không rõ ràng. Nó ghi lại số phiên bản cho các cặp khóa, dữ liệu nhưng vẫn duy trì phiên bản mới nhất duy nhất trong cửa hàng. Các ứng dụng không cần quan tâm đến việc điều chỉnh các phiên bản không tương thích vì Cơ sở dữ liệu Oracle NoSQL sử dụng tính năng sao chép dựa trên người dẫn đầu; nút chính Paxos luôn có giá trị cập nhật nhất cho một khóa nhất định, trong khi các bản sao không phải chính có thể có các phiên bản cũ hơn một chút. Các ứng dụng có thể sử dụng số phiên bản để đảm bảo tính nhất quán cho các hoạt động đọc-sửa-ghi.

## **Tùy chọn mô hình hóa dữ liệu**

Là một kho dữ liệu đa mô hình thực sự, Cơ sở dữ liệu Oracle NoSQL cung cấp một số tùy chọn khác nhau để lập mô hình dữ liệu:

Key/value: Oracle NoSQL thực hiện một bản đồ key từ người dùng định nghĩa ( định dạng như string ). Nó có ghi lại số phiên bản cho cặp key/value, nhưng vẫn duy trì các phiên bản duy nhất trong store. Ứng dụng có thể sử dụng các số phiên bản để đảm bảo tính thống nhất trong các hoạt động đọc chỉnh sửa ghi.

Bảng: Sử dụng tùy chọn mô hình hóa bảng này, các nhà phát triển chỉ định các định nghĩa bảng bằng ngôn ngữ dữ liệu SQL, rất giống với cơ sở dữ liệu quan hệ. SQL dòng lệnh để truy vấn Cơ sở dữ liệu Oracle NoSQL hoặc API Cơ sở dữ liệu Oracle NoSQL có thể được sử dụng để truy cập dữ liệu bảng.

Graph-based hay cơ sở dữ liệu kiểu đồ thị có thể lưu trữ các thực thể và các mối quan hệ giữa các thực thể đó. Những thực thể này được lưu dưới dạng một node với mối quan hệ là các cạnh. Mỗi một cạnh sẽ cho biết một mối quan hệ giữa các node. Mỗi node và cạnh đều chỉ có một mã định danh.

Loại cơ sở dữ liệu dạng document – based (hay còn gọi là cơ sở dữ liệu dạng tài liệu) có vai trò lưu trữ, truy xuất dữ liệu dưới dạng một cặp khoá giá trị (key value) nhưng phần giá trị sẽ được lưu trữ ở dạng tài liệu. Phần tài liệu ở đây sẽ có định dạng XML hoặc JSON.

Cơ sở dữ liệu kiểu key-value có một số hạn chế như:

Không có mối quan hệ với Multiple data.

Nếu bạn đang lưu trữ số lượng lớn khóa và không thể lưu một trong các khóa, bạn không thể quay về các phần còn lại của thao tác thực hiện (multi operation transactions).

Kết quả tìm kiếm “khóa” dựa vào một số thông tin được tìm thấy ở phần “giá trị” của các cặp key-value (query data by “value”).

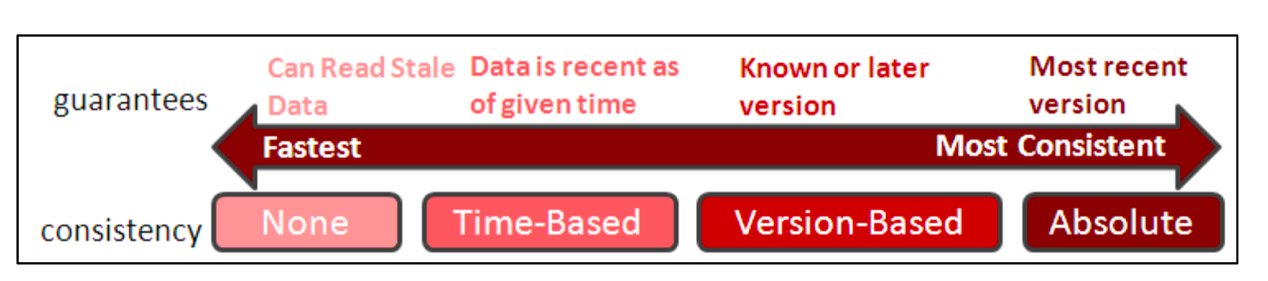
Do các hoạt động bị giới hạn trong một khóa tại một thời điểm nhất định nên không có cách để chạy nhiều khóa trong cùng một khoảng thời gian (operation by groups).

## **Mô hình Sharding**

Cơ sở dữ liệu Oracle NoSQL băm các khóa thành các phân đoạn để cung cấp phân phối qua tập hợp các nút lưu trữ cung cấp khả năng lưu trữ cho cơ sở dữ liệu. Tuy nhiên, các ứng dụng có thể tận dụng khả năng của khóa con (còn được gọi là bảng con) để đạt được vị trí dữ liệu. Khóa chính là sự ghép nối tiềm năng của khóa phân đoạn và khóa không phân đoạn, cả hai đều được ứng dụng chỉ định. Tất cả các bản ghi chia sẻ khóa phân đoạn đều được định vị chung để đạt được dữ liệu địa phương. Trong một bộ sưu tập các khóa phân đoạn được đặt cùng vị trí, khóa chính đầy đủ, bao gồm cả phần phân đoạn và không phân đoạn, cung cấp tra cứu được lập chỉ mục, nhanh chóng. Ví dụ: một ứng dụng lưu trữ email của người dùng có thể sử dụng ID người dùng làm khóa phân đoạn và sau đó có một số bảng con cho các thư mục email khác nhau do người dùng sở hữu, chẳng hạn như hộp thư đến, đã xóa, bản nháp, v.v. Khi hiển thị giao diện người dùng khi đăng nhập, Trang đầu tiên của các thư trong hộp thư đến của người dùng (hoặc bất kỳ thư mục nào khác cho người dùng đó) có thể được đọc bằng cách sử dụng một lệnh gọi API từ ứng dụng. Nói cách khác, một cuộc gọi mạng duy nhất có thể được thực hiện tới Cơ sở dữ liệu Oracle NoSQL để truy xuất tất cả các thông báo cho một người dùng cụ thể.

## **Tính nhất quán và mô hình độ bền**

Trong khi nhiều cơ sở dữ liệu NoSQL cung cấp tính nhất quán cuối cùng, thì Cơ sở dữ liệu Oracle NoSQL cung cấp một số chính sách nhất quán khác nhau. Ở một đầu của phổ, các ứng dụng có thể chỉ định tính nhất quán tuyệt đối, điều này đảm bảo rằng tất cả các lần đọc đều trả về giá trị được ghi gần đây nhất cho một khóa được chỉ định. Ở đầu kia của phổ, các ứng dụng có khả năng dung nạp dữ liệu không nhất quán có thể chỉ định tính nhất quán yếu, cho phép cơ sở dữ liệu trả về một giá trị hiệu quả ngay cả khi nó không hoàn toàn được cập nhật. Ở giữa hai thái cực này, các ứng dụng có thể chỉ định tính nhất quán dựa trên thời gian để hạn chế mức độ cũ của bản ghi đối với phiên bản mới nhất ở người dẫn đầu hoặc tính nhất quán dựa trên phiên bản để hỗ trợ cả tính nguyên tử cho các hoạt động đọc-sửa-ghi và đọc ít nhất là gần đây nhất như phiên bản được chỉ định. Hình 2 cho thấy phạm vi của các chính sách nhất quán linh hoạt cho phép các nhà phát triển dễ dàng tạo ra các giải pháp kinh doanh cung cấp sự đảm bảo về dữ liệu trong khi vẫn đáp ứng các yêu cầu về độ trễ và khả năng mở rộng của ứng dụng.



Hình 2: Mô hình CAP

Cơ sở dữ liệu Oracle NoSQL cũng cung cấp một loạt các chính sách về độ bền chỉ định những gì đảm bảo hệ thống thực hiện sau sự cố. Ở một mức độ nào đó, các ứng dụng có thể yêu cầu chặn yêu cầu ghi đó cho đến khi bản ghi được ghi vào bộ nhớ ổn định trên tất cả các bản sao. Điều này có ý nghĩa rõ ràng về hiệu suất và tính khả dụng nhưng đảm bảo rằng nếu ứng dụng ghi dữ liệu thành công, dữ liệu đó sẽ tồn tại và có thể được khôi phục ngay cả khi tất cả các bản sao tạm thời không khả dụng do nhiều lỗi đồng thời. Ở một khía cạnh khác, các ứng dụng có thể yêu cầu các hoạt động ghi quay trở lại ngay sau khi hệ thống ghi nhận sự tồn tại của việc ghi, ngay cả khi dữ liệu không liên tục ở bất kỳ đâu. Chính sách như vậy cung cấp hiệu suất ghi tốt nhất nhưng không đảm bảo độ bền. Bằng cách chỉ định thời điểm cơ sở dữ liệu ghi các bản ghi vào đĩa và phần nào trong số các bản sao của bản ghi đó phải liên tục (không có, tất cả hoặc phần lớn đơn giản), các ứng dụng có thể thực thi một loạt các chính sách về độ bền và tạo ra sự cân bằng quan trọng giữa độ trễ và độ bền.

## **Mô hình lập trình**

Như đã nêu trước đây, các nhà phát triển có nhiều lựa chọn về cấu trúc mô hình hóa trong Cơ sở dữ liệu Oracle NoSQL và những cấu trúc mô hình hóa dữ liệu này dẫn trực tiếp đến mô hình lập trình. Cơ sở dữ liệu Oracle NoSQL cho phép các nhà phát triển vừa lập mô hình dữ liệu của họ vừa sử dụng một API tự nhiên cùng nhau. Ví dụ: nếu bạn cần sử dụng JSON làm mô hình dữ liệu của mình, thì API Oracle NoSQL Database JSON đã được phát triển để đơn giản hóa việc phát triển ứng dụng của bạn. Tương tự như vậy, việc sử dụng các bảng để lập mô hình ứng dụng của bạn sẽ dẫn đến việc sử dụng tự nhiên API Bảng Cơ sở dữ liệu của Oracle NoSQL. Trong bài báo này, chúng tôi sẽ đưa ra các ví dụ về hai mô hình cụ thể này.

## **Mô hình bảng**

Một trong những mô hình dữ liệu được Cơ sở dữ liệu Oracle NoSQL đưa ra là mô hình bảng. Mặc dù mô hình này có thể quen thuộc hơn với các nhà phát triển có kinh nghiệm với cơ sở dữ liệu quan hệ, nhưng mô hình bảng Cơ sở dữ liệu Oracle NoSQL cung cấp các kiểu dữ liệu duy nhất không tìm thấy trong thế giới quan hệ, chẳng hạn như mảng, bản đồ và bản ghi nhúng. Việc tạo các bảng trong Cơ sở dữ liệu Oracle NoSQL được thực hiện thông qua Ngôn ngữ Định nghĩa Dữ liệu SQL (DDL), rất giống với SQL quan hệ. Dưới đây là một câu lệnh DDL ví dụ sẽ tạo một bảng có tên Người dùng với các cột sau:

ID – Đây sẽ là khóa chính cũng như khóa phân đoạn (trừ khi được chỉ định, khóa phân đoạn mặc định là khóa chính)

- firstName và lastName – Chuỗi để lưu trữ họ và tên của người dùng.

Một mảng các bản ghi địa chỉ – Chúng tôi sẽ lưu trữ nhiều địa chỉ cho người dùng này. Mỗi bản ghi địa chỉ sẽ chứa các thuộc tính sau:

Street, City, State – Đường, thành phố và tiểu bang cho địa chỉ này

zip – Mã zip số nguyên cho địa chỉ này

addrType – Một phép liệt kê có thể chứa “cơ quan” hoặc “nhà riêng”

Trong Cơ sở dữ liệu Oracle NoSQL, DDL để tạo bảng này có thể được chỉ định là:

Statement result res = KVStore.executeSync(“CREATE TABLE user(id INTEGER, firstName STRING, lastName String,  
addresses ARRAY(RECORD (street STRING, city String, state STRING,

zip INTEGER, addrType ENUM(home, work), primary key(id))))”);

Việc kết hợp Cơ sở dữ liệu Oracle NoSQL vào các ứng dụng rất đơn giản. API cho các thao tác Tạo, Đọc, Cập nhật và Xóa (CRUD) cơ bản, cũng như truy cập SQL, được đóng gói trong một tệp jar duy nhất. Các ứng dụng có thể sử dụng API hoặc truy vấn SQL từ một hoặc nhiều quy trình máy khách truy cập quy trình máy chủ Cơ sở dữ liệu Oracle NoSQL độc lập, giúp giảm bớt nhu cầu thiết lập cấu hình đa hệ thống để phát triển và thử nghiệm ban đầu.

## **Truy cập dữ liệu bảng qua SQL**

Đối với các truy vấn không phù hợp với kiểu truy cập từ khóa đơn giản/giá trị đơn giản hoặc quét phạm vi đơn giản bằng khóa phụ, Cơ sở dữ liệu Oracle NoSQL cung cấp quyền truy cập SQL gốc vào dữ liệu. Mặc dù Cơ sở dữ liệu Oracle NoSQL cung cấp một phương ngữ SQL khá phong phú, SQL này có thể được coi là một tập hợp con đơn giản hơn của SQL mạnh hơn và toàn diện hơn do Oracle RDBMS cung cấp. Đối với những ứng dụng yêu cầu mức truy cập SQL này, Cơ sở dữ liệu Oracle NoSQL được tích hợp với Oracle RDBMS, cho phép RDBMS truy vấn dữ liệu trực tiếp từ các bảng Cơ sở dữ liệu Oracle NoSQL.

Oracle NoSQL của SQL là khái niệm về khả năng truy vấn SQL đối với dữ liệu phi quan hệ như mảng và bản đồ. Phương ngữ Một lĩnh vực khác biệt đáng chú ý giữa các phương ngữ Cơ sở dữ liệu Oracle RDBMS và SQL trong Cơ sở dữ liệu Oracle NoSQL cung cấp các cấu trúc mạnh mẽ để cắt qua các mảng và bản đồ cũng như cung cấp các biểu thức lọc trên các cấu trúc này. Các ví dụ dưới đây minh họa một tập hợp con nhỏ của khả năng này.

VD1: Trả về họ và tên của tất cả người dùng ở thành phố New York:

Select

firstName,

lastName

from

user

Where

addresses.phones[$element.city = “New York”]

VD2: Trả về họ và tên của tất cả người dùng có địa chỉ nhà riêng trong mã zip 94107 và sắp xếp kết quả theo thuộc tính lastName:

Select

firstName,

lastName

from

user

where

addresses.phones[$element.zip=94107]and addresses.phones[$element.attrType = “home”]

order by lastName

VD3: Hiển thị trang đầu tiên (25 kết quả mỗi trang) của tất cả người dùng ở New Haven, Newark và Thành phố New York:

select \*

From

user

where

addresses.phones[$element.city = “New York”] or

addresses.phones[$element.city=“NewHaven”]or addresses.phones[$element.city = “Newark”]

order by lastName limit 25 offset 1

## **Làm việc với tài liệu JSON**

JSON đã phát triển đáng kể trong vài năm qua dưới dạng định dạng dữ liệu thuận tiện cho các ứng dụng web, trả về kết quả từ các cuộc gọi API dịch vụ và trao đổi dữ liệu giữa các ứng dụng.Là một kiểu dữ liệu tự mô tả, dữ liệu được định dạng JSON cung cấp tính linh hoạt cao nhất cho những ứng dụng muốn lưu trữ nội dung đặc biệt mà không cần phải chỉ định trước lược đồ.

Các lược đồ cố định được lưu trữ một lần, trong khi mọi tài liệu JSON chứa một bản sao của lược đồ.

Cơ sở dữ liệu Oracle NoSQL cung cấp một sự kết hợp độc đáo của các bảng lược đồ cố định cũng như hỗ trợ không cần lược đồ đặc biệt cho các tài liệu JSON. Các nhà phát triển có thể chọn phần nào của ứng dụng sẽ tận dụng tính linh hoạt của JSON không cần lược đồ và những phần nào của ứng dụng nên tối ưu hóa dung lượng lưu trữ và tính toán để đổi lấy lược đồ cố định.

Phiên bản lược đồ cố định từ ví dụ trước của chúng ta có thể được biểu diễn dưới dạng JSON không có lược đồ bên dưới:

Statement result res = KVStore.executeSync(“CREATE TABLE user(id INTEGER, userData JSON primary key(id)))”);

Dữ liệu JSON có thể được chèn bằng cách sử dụng:

Row r = Table.createRow().put(“id”,123456).put("userData", FieldValueFactory.createValueFromJson(

“{

“firstName” : “John”,

“lastName” : “Doe”

"addresses": [ {

“street” : “127 Spring St”, “city” : “New York”, “state” : “NY”,

“zip” : 10012, ”addrType” : “work”},

{“street” : “625 Ridgewood Rd”, “city” : “Paramus”, “state” : “NJ”,

“zip” : 07675, ”addrType” : “home”}

] }”);

));

Dữ liệu JSON cũng có thể được truy vấn bằng cách sử dụng cùng một ngôn ngữ SQL hoạt động trên Bảng. Ngoài ra, đối với các tài liệu JSON đặc biệt , một số toán tử mới và biểu thức "trường hợp" được giới thiệu để hỗ trợ xử lý các tài liệu có thể có định dạng rời rạc hoặc không xác định.

VD4: Tìm tất cả người dùng ở thành phố New York. Trong trường hợp này, nếu tài liệu có một mảng địa chỉ thì hãy áp dụng vị ngữ cho mảng, nếu không, hãy áp dụng vị ngữ trực tiếp cho một thuộc tính được gọi là "thành phố" trong chính tài liệu.

select

firstName,

lastName

from

User u

where

(case when u.addresses instanceof ARRAY then

u.addresses[$element.city = “New York”] else

u.city = “New York” end)

VD5: Tìm tất cả người dùng ở Thành phố New York khi tài liệu JSON chứa thuộc tính thành phố, trong tài liệu dành cho vị ngữ truy vấn

select \*

from User u

where

(case when exists(u.address.city) then u.address.city = “New York”

Việc kết hợp Cơ sở dữ liệu Oracle NoSQL vào các ứng dụng rất đơn giản. Các API cho các thao tác Tạo, Đọc, Cập nhật và Xóa (CRUD) cơ bản, cũng như truy cập SQL, được đóng gói trong một tệp jar duy nhất. Các ứng dụng có thể sử dụng các API hoặc truy vấn SQL từ một hoặc nhiều quy trình máy khách truy cập vào quy trình máy chủ cơ sở dữ liệu Oracle NoSQL Database độc lập, giảm bớt nhu cầu thiết lập cấu hình đa hệ thống để phát triển và thử nghiệm ban đầu.

CHƯƠNG 4: Khả năng phục hồi thất bại

Cơ sở dữ liệu Oracle NoSQL đưa ra khái niệm về các vùng khả dụng dưới dạng các thùng chứa cách ly lỗi. Vùng khả dụng có thể ở dạng giá đỡ máy chủ, công tắc trên cùng của giá đỡ, sàn của tòa nhà hoặc toàn bộ trung tâm dữ liệu. Các khu vực cũng được mô tả là chính hoặc phụ, có các khu vực chính dành cho bầu cử tổng thể cũng như xác nhận số đại biểu cần thiết.Cơ sở dữ liệu Oracle NoSQL tiếp tục cho phép cấu hình bất kỳ số vùng chính và phụ nào. Sau khi cấu hình, các bản sao cho từng phân đoạn trong kho lưu trữ dữ liệu được bố trí trên các vùng này sao cho lỗi của bất kỳ vùng nào sẽ hạn chế (hoặc chứa) lỗi này để mỗi phân đoạn trong kho dữ liệu luôn duy trì đủ số đại biểu cần thiết.

Cơ sở dữ liệu Oracle NoSQL triển khai một Dịch vụ quản trị riêng có tính khả dụng cao. Nhất quán với triết lý “Không có điểm lỗi duy nhất” của Cơ sở dữ liệu Oracle NoSQL, hoạt động liên tục của một cài đặt không phụ thuộc vào tính khả dụng của Dịch vụ quản trị. Do đó, cả cơ sở dữ liệu và Dịch vụ quản trị vẫn khả dụng trong quá trình thay đổi cấu hình.

CHƯƠNG 5: Ngành kiến trúc

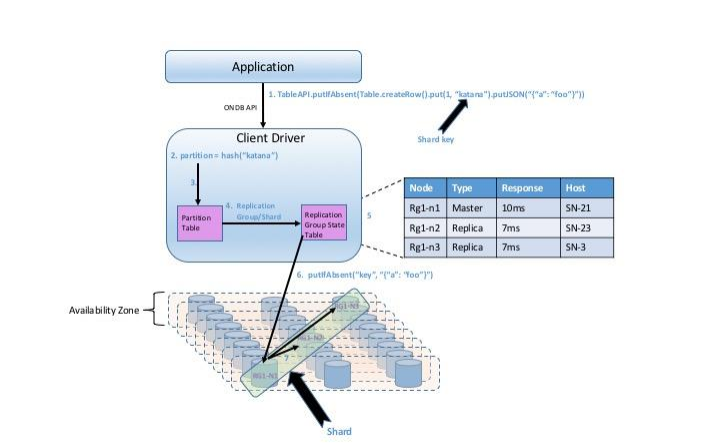
Chúng tôi trình bày kiến trúc Cơ sở dữ liệu Oracle NoSQL bằng cách theo dõi quá trình thực hiện thao tác ghi thông qua các thành phần logic của hệ thống và sau đó thảo luận về cách các thành phần đó ánh xạ tới hoạt động của phần cứng và phần mềm thực tế. Chúng tôi sẽ tạo một bản ghi với khóa “Katana” và một đối tượng JSON là giá trị “{“a”: “foo”}”. Hình 3 mô tả lời gọi phương thức putIfAbsent(“Katana”, “{“a”: “foo”}

Figure 3 – Architecture - Oracle NoSQL Database – Write Request Flow

Ứng dụng đưa ra phương pháp putIfAbsent trình điều khiển khách hàng (bước 1). Trình điều khiển máy khách băm khóa “Katana” để chọn một trong số các phân vùng (bước 2). Mỗi phân vùng được gán cho một nhóm sao chép cụ thể (phân đoạn). Trình điều khiển tham khảo bảng phân vùng (bước 3) để ánh xạ số phân vùng vào một phân đoạn.

Một phân đoạn bao gồm một số (có thể định cấu hình) số lượng nút sao chép, và mỗi nút sao chép nằm trong một vùng sẵn sàng riêng biệt (để ngăn chặn lỗi). Số lượng nút sao chép trong một phân đoạn quyết định số lượng lỗi mà từ đó hệ thống có khả năng phục hồi; một hệ thống có ba nút trên mỗi phân đoạn có thể chịu được hai lỗi trong khi tiếp tục phục vụ các yêu cầu đọc dịch vụ. Khả năng chịu đựng các lỗi ghi của nó dựa trên chính sách độ bền được yêu cầu trong lệnh gọi API ghi. Nếu ứng dụng không yêu cầu số người tham gia xác nhận một lần ghi, thì hệ thống cũng chỉ có thể chịu được tối đa hai lỗi ghi. Một nhóm năm nút có thể chịu được tối đa đến bốn lỗi đối với mỗi lần đọc và tối đa hai lần lỗi đối với hai lần ghi, ngay cả khi ứng dụng yêu cầu chính sách độ bền và yêu cầu phần lớn các trang web xác nhận thao tác ghi.

Được cung cấp một phân đoạn, Trình điều khiển máy khách tiếp theo sẽ tham khảo Bảng trạng thái nhóm sao chép (RGST) (bước 4). Đối với mỗi phân đoạn, RGST chứa thông tin về từng nút sao chép bao gồm nhóm các nút sao chép trong phân đoạn (bước 5). Dựa trên thông tin trong RGST, chẳng hạn như danh tính của máy chủ và tải trên các nút khác nhau trong nhóm sao chép, Trình điều khiển Máy khách chọn nút để gửi yêu cầu và chuyển tiếp yêu cầu đến nút thích hợp (bước 6 ). Trong trường hợp này, vì chúng tôi đang thực hiện thao tác ghi, nên yêu cầu phải đi đến nút chính.

Sau đó, nút sao chép sẽ áp dụng thao tác. Trong trường hợp putIfAbsent, nếu khóa tồn tại, thao tác không có có hiệu lực và trả về lỗi, cho biết rằng mục nhập được chỉ định đã có trong cửa hàng. Nếu chìa khóa không tồn tại, nút sao chép sẽ thêm cặp khóa/giá trị vào cửa hàng và sau đó truyền cặp khóa/giá trị mới tới các nút khác trong nhóm sao chép (bước 7).

CHƯƠNG 6: Cài đặt

Một bản cài đặt Cơ sở dữ liệu Oracle NoSQL bao gồm hai phần chính: Trình điều khiển máy khách và bộ sưu tập các Nút lưu trữ. Như được hiển thị trong Hình 3, trình điều khiển máy khách triển khai bản đồ phân vùng và RGST, trong khi các nút lưu trữ triển khai các nút sao chép bao gồm các phân đoạn. Trong phần này, chúng ta sẽ xem xét kỹ hơn từng thành phần này

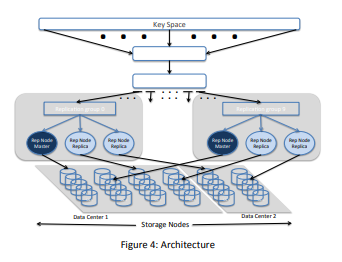
# **Storage Nodes**

Nút lưu trữ (SN) thường là một máy vật lý có bộ lưu trữ liên tục cục bộ, dạng đĩa hoặc trạng thái rắn, CPU có một hoặc nhiều lõi, bộ nhớ và địa chỉ IP. Hệ thống có nhiều nút lưu trữ hơn sẽ cung cấp thông lượng tổng hợp hoặc dung lượng lưu trữ lớn hơn so với hệ thống có ít nút hơn và các hệ thống có mức độ sao chép lớn hơn trong các nhóm sao chép có thể giảm độ trễ yêu cầu đối với các cài đặt có mức độ sao chép nhỏ hơn. Ngoài ra, nhiều SN hơn dẫn đến tính khả dụng cao hơn trên toàn bộ hệ thống.

Tác nhân nút lưu trữ (SNA) chạy trên mỗi nút lưu trữ, theo dõi hành vi của nút đó. SNA vừa nhận cấu hình vừa báo cáo thông tin giám sát cho Dịch vụ quản trị. SNA liên tục thu thập dữ liệu vận hành từ nút lưu trữ và sau đó cung cấp dữ liệu đó cho Dịch vụ quản trị khi được yêu cầu.

Một nút lưu trữ phục vụ một hoặc nhiều nút sao chép. Mỗi nút sao chép thuộc về một nhóm sao chép duy nhất. Các nút trong một nhóm sao chép duy nhất đều phục vụ cùng một dữ liệu. Mỗi nhóm có một nút chính được chọn động để xử lý tất cả các hoạt động sửa đổi dữ liệu (tạo, cập nhật và xóa). Các nút khác là bản sao chỉ đọc nhưng có thể đảm nhận vai trò chính nếu nút chính bị lỗi. Một cài đặt điển hình sử dụng hệ số sao chép là ba trong các nhóm sao chép, để đảm bảo rằng hệ thống có thể tồn tại ít nhất hai lỗi đồng thời và tiếp tục phục vụ các hoạt động đọc. Các ứng dụng yêu cầu độ tin cậy cao hơn hoặc thấp hơn có thể điều chỉnh thông số này cho phù hợp.

Hình 4 cho thấy một cài đặt với 10 nhóm sao chép (0-9) (còn được gọi là phân đoạn). Mỗi nhóm sao chép có hệ số sao chép là 3 (một bản chính và hai bản sao) trải rộng trên hai trung tâm dữ liệu. Lưu ý rằng chúng ta có thể đặt hai trong số các nút sao chép vào trung tâm dữ liệu lớn hơn và nút sao chép cuối cùng vào trung tâm dữ liệu nhỏ hơn. Kiểu sắp xếp này có thể phù hợp với ứng dụng sử dụng trung tâm dữ liệu lớn hơn để truy cập dữ liệu chính, duy trì trung tâm dữ liệu nhỏ hơn trong trường hợp trung tâm dữ liệu chính gặp sự cố nghiêm trọng.



Hình 4: Kiến trúc

10 nhóm sao chép được lưu trữ trên 30 nút lưu trữ, trải rộng trên hai trung tâm dữ liệu.

Các nút sao chép hỗ trợ API cơ sở dữ liệu Oracle NoSQL thông qua các lệnh gọi RMI từ máy khách và lấy dữ liệu trực tiếp từ hoặc ghi dữ liệu trực tiếp vào hệ thống lưu trữ có cấu trúc nhật ký, cung cấp khả năng ghi hiệu suất cao, đồng thời duy trì cấu trúc chỉ mục mang lại hiệu suất đọc có độ trễ thấp.

Cơ sở dữ liệu Oracle NoSQL sử dụng bản sao để đảm bảo tính khả dụng của dữ liệu trong trường hợp lỗi. Kiến trúc một chủ của nó yêu cầu ghi được áp dụng tại nút chính và sau đó được truyền tới các bản sao. Trong trường hợp nút chính bị lỗi, các nút trong nhóm sao chép sẽ tự động tổ chức một cuộc bầu cử đáng tin cậy (sử dụng giao thức Paxos), chọn một trong các nút còn lại làm nút chính. Master mới sau đó đảm nhận trách nhiệm ghi.

# **Trình điều khiển khách hàng**

Trình điều khiển máy khách là một tệp jar Java xuất API sang các ứng dụng. Trình điều khiển máy khách là cấu trúc liên kết nhận biết bằng cách sử dụng bảng trạng thái nhóm sao chép (RGST). Cấu trúc liên kết ánh xạ hiệu quả các khóa đến phân vùng và từ phân vùng đến các nhóm sao chép. Đối với mỗi nhóm sao chép, nó bao gồm tên máy chủ của nút lưu trữ lưu trữ từng nút sao chép trong nhóm, tên dịch vụ được liên kết với các nút sao chép và trung tâm dữ liệu chứa từng nút lưu trữ. Sau đó, máy khách sử dụng RGST cho hai mục đích chính: xác định nút chính của nhóm sao chép, để nó có thể gửi yêu cầu ghi tới nút chính và cân bằng tải trên tất cả các nút trong một nhóm sao chép để đọc. Vì RGST là một cấu trúc dữ liệu được chia sẻ quan trọng nên mỗi máy khách và nút sao chép sẽ duy trì bản sao của nó, do đó tránh được bất kỳ điểm lỗi đơn lẻ nào. Cả máy khách và nút sao chép đều chạy RequestDispatcher sử dụng RGST để (tái) hướng các yêu cầu ghi tới máy chủ và đọc các yêu cầu tới thành viên thích hợp của nhóm sao chép.

Cấu trúc liên kết được tải trong quá trình khởi tạo nút máy khách hoặc nút sao chép và sau đó có thể được cập nhật bởi quản trị viên nếu có thay đổi cấu trúc liên kết. RGST là động, yêu cầu bảo trì liên tục. Mỗi lần sao chép nút chạy một luồng, được gọi là luồng Cập nhật trạng thái nút sao chép, chịu trách nhiệm duy trì liên tục RGST. Luồng cập nhật, cũng như RequestDispatcher, thu thập thông tin theo cơ hội trên điều khiển từ xa các nút sao chép bao gồm trạng thái hiện tại của nút trong nhóm sao chép của nó, một chỉ báo về cách cập nhật nút là, thời gian tương tác thành công cuối cùng với nút, thời gian phản hồi trung bình của nút và độ dài hiện tại của hàng đợi yêu cầu nổi bật của nó. Ngoài ra, luồng cập nhật duy trì các kết nối mạng và thiết lập lại những cái bị hỏng. Việc bảo trì này được thực hiện bên ngoài chu trình yêu cầu/phản hồi của RequestDispatcher đối với giảm thiểu tác động của các kết nối bị hỏng đối với độ trễ.

CHƯƠNG 7: Bảo vệ

Cơ sở dữ liệu Oracle NoSQL có thể được cấu hình an toàn. Trong một cấu hình an toàn, giao tiếp mạng giữa Các ứng dụng khách, tiện ích và thành phần máy chủ NoSQL được mã hóa bằng SSL/TLS và tất cả các quy trình phải xác thực bản thân với các thành phần mà chúng kết nối.

Có hai cấp độ bảo mật cần lưu ý. Đây là những bảo mật mạng, cung cấp một lớp bên ngoài của bảo vệ ở cấp độ mạng và xác thực ủy quyền người dùng. An ninh mạng được cấu hình tại file mức hệ thống thông thường trong quá trình cài đặt, trong khi xác thực ủy quyền người dùng được quản lý thông qua Tiện ích NoSQL.

Cơ sở dữ liệu Oracle NoSQL cung cấp các khả năng sau để bảo mật kho lưu trữ và thiết lập độ phức tạp của mật khẩu:

Tiện ích cấu hình bảo mật. Cho phép cấu hình để thêm bảo mật cho một Oracle NoSQL mới hoặc hiện có Cài đặt cơ sở dữ liệu.

Phương thức xác thực. Cơ sở dữ liệu Oracle NoSQL cung cấp xác thực mật khẩu cho người dùng và hệ thống. Các Phiên bản Enterprise Edition của Cơ sở dữ liệu Oracle NoSQL cũng hỗ trợ xác thực Kerberos.

Mã hóa. Dữ liệu được mã hóa trên mạng để ngăn chặn truy cập trái phép vào dữ liệu đó.

Lưu trữ mật khẩu bên ngoài. Cơ sở dữ liệu Oracle NoSQL cung cấp hai loại phương pháp lưu trữ mật khẩu bên ngoài có thể thao tác (một loại dành cho triển khai CE). Các loại là văn bản rõ ràng và sử dụng Oracle Wallet (có sẵn với Phiên bản doanh nghiệp cơ sở dữ liệu Oracle NoSQL).

Chính sách bảo mật. Cơ sở dữ liệu Oracle NoSQL cho phép thiết lập các hành vi để đảm bảo một môi trường an toàn.

Ủy quyền dựa trên vai trò. Cơ sở dữ liệu Oracle NoSQL cung cấp các vai trò hệ thống, đặc quyền và vai trò do người dùng xác định trước cho người dùng. Bạn có thể đặt các đặc quyền mong muốn cho người dùng bằng cách cấp vai trò.

CHƯƠNG 8: Hội nhập

Cơ sở dữ liệu Oracle NoSQL (phiên bản Enterprise Edition), khi được phân phối, đã sẵn sàng để tích hợp với nhiều Oracle khác nhau các công nghệ như Cơ sở dữ liệu Oracle, Oracle Enterprise Manager, Oracle Coherence, Oracle Big Data Spatial và Graph, Oracle Rest Data Services, và các công nghệ mã nguồn mở như Hadoop, Hive và Spark.

Tích hợp cơ sở dữ liệu Oracle

Để đọc dữ liệu từ Cơ sở dữ liệu Oracle NoSQL vào Bảng bên ngoài của Cơ sở dữ liệu Oracle, hãy xác định Bảng bên ngoài với một hoặc nhiều tệp vị trí; các tệp vị trí sẽ không thực sự chứa bất kỳ dữ liệu nào mà thay vào đó chứa cấu hình thông tin liên quan đến việc kết nối với Cơ sở dữ liệu Oracle và Cơ sở dữ liệu Oracle NoSQL, các hạn chế truy vấn và định dạng.

Chỉ định nhiều tệp vị trí trong định nghĩa Bảng bên ngoài cho phép mức độ song song trong dữ liệu đọc hiểu. Khi Bảng bên ngoài được xác định, hãy chạy tiện ích Xuất bản để "xuất bản" thông tin về Bảng bên ngoài và cách truy cập dữ liệu trong Cơ sở dữ liệu Oracle NoSQL. Thông tin này được ghi dưới dạng tài liệu XML vào mỗi của (các) tệp vị trí.

Khi quá trình xuất bản hoàn tất, SQL SELECT có thể được thực thi đối với Bảng bên ngoài. Điều này sẽ gây ra

Tập lệnh <KVHOME>/exttab/bin/nosql\_stream được gọi, do đó khiến lớp Preproc được gọi. Các tiền xử lý:

Đọc cấu hình và thông tin truy cập Cơ sở dữ liệu NoSQL từ tệp vị trí,

Đọc dữ liệu từ Cơ sở dữ liệu NoSQL,

Định dạng nó bằng trình định dạng do người dùng xác định hoặc định dạng mặc định và

» Ghi nó vào thiết bị xuất chuẩn.

Tích hợp SQL Hadoop, Hive và Dữ liệu lớn

Cơ sở dữ liệu Hadoop to Oracle NoSQL được hỗ trợ bằng cách chạy các công việc Hadoop MapReduce đối với dữ liệu được được lưu trữ trong bảng Cơ sở dữ liệu Oracle NoSQL. Một lớp InputFormat tùy chỉnh xác định phương thức đọc và ghi dữ liệu đầu vào MapReduce. Ngoài ra còn có một triển khai tùy chỉnh của InputSplit xác định cách lấy danh sách "phân tách" hợp lý cho công việc và triển khai tùy chỉnh của lớp RecordRead thực hiện công việc định vị và trả về các cặp Khóa/Giá trị được sử dụng bởi MapReduce và SerDe.

Oracle Big Data SQL cho phép người dùng sử dụng ngôn ngữ SQL để quản lý và thao tác dữ liệu được lưu trữ trong một số của các vị trí khác nhau trong các cơ sở dữ liệu khác nhau. Oracle Big Data SQL đạt được điều này bằng cách trình bày Oracle NoSQL Dữ liệu cơ sở dữ liệu dưới dạng các bảng bên ngoài nâng cao cho Cơ sở dữ liệu Oracle. Điều này được thực hiện bằng cách lập bản đồ bên ngoài ngữ nghĩa để truy cập dữ liệu từ các nguồn này đến các cấu trúc bên trong Cơ sở dữ liệu Oracle.

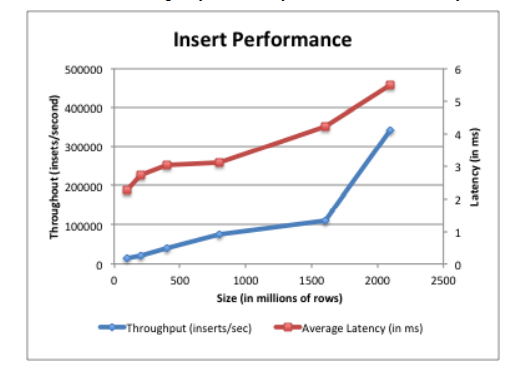
Oracle Big Data SQL không chỉ cho phép dễ dàng tích hợp dữ liệu từ các nguồn Hadoop và NoSQL, Big Data SQL cũng tận dụng các cơ chế lưu trữ cơ bản để cung cấp hiệu suất tốt nhất có thể. SQL dữ liệu lớn công nghệ đẩy xuống vị từ cho phép các vị từ trong các truy vấn được đưa ra trong Cơ sở dữ liệu Oracle được thực thi từ xa hệ thống và được đẩy vào các định dạng tệp nhất định.

Tích hợp với Hive đạt được bằng cách chạy các truy vấn Hive bằng ngôn ngữ giống như SQL có tên là HiveQL (HQL) trên dữ liệu được lưu trữ trong Cơ sở dữ liệu Oracle NOSQL. Hive TableStorageHandler tùy chỉnh xác định cơ chế được yêu cầu bởi cả Hive và Big Data SQL và được sử dụng khi thực hiện truy vấn đối với một bảng trong Oracle Cửa hàng cơ sở dữ liệu NoSQL. Ngoài ra, còn có lớp ObjectInspector tùy chỉnh được sử dụng để dịch các kiểu dữ liệu sang định dạng dữ liệu Cơ sở dữ liệu Oracle NoSQL tương ứng.

Để cải thiện hiệu suất truy vấn, cả Hive và Big Data SQL đều hỗ trợ một hình thức đẩy xuống vị từ nơi máy khách giao diện người dùng bên phân tách mệnh đề WHERE của truy vấn thành thông tin cột và so sánh tương ứng hoạt động. Điều này dẫn đến việc đẩy các thành phần kết quả sang phía máy chủ cơ sở dữ liệu để xử lý. SQL dữ liệu lớn sử dụng giao diện Hive để hỗ trợ tính năng đẩy xuống vị từ khi thực hiện truy vấn SQL dữ liệu lớn đối với Dữ liệu bảng KVStore. Để hỗ trợ tính năng đẩy xuống vị ngữ, có một triển khai tùy chỉnh của HiveStoragePredicateHandler hỗ trợ và xác định tiêu chí phân tách các vị từ được đẩy xuống quét bảng và quét chỉ mục.

CHƯƠNG 9: DEMO

Chúng tôi đã thử nghiệm với nhiều cấu hình Cơ sở dữ liệu Oracle NoSQL khác nhau và trình bày một số hiệu suất kết quả của Yahoo! Điểm chuẩn phục vụ đám mây (YCSB), thể hiện cách hệ thống thay đổi quy mô theo số lượng các nút trong hệ thống. Như với tất cả các phép đo hiệu suất, kết quả có thể thay đổi do một số yếu tố.

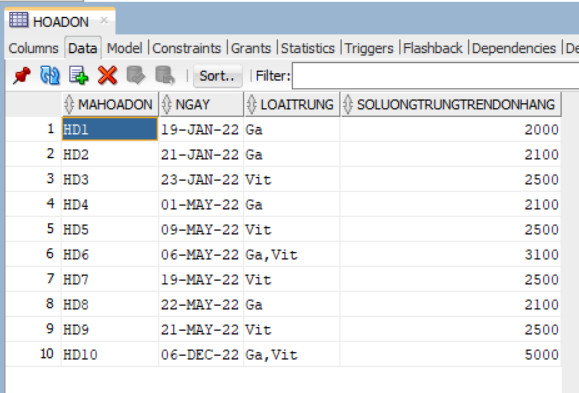
Chúng tôi đã áp dụng tải YCSB không đổi cho mỗi nút lưu trữ cho các cấu hình có kích thước khác nhau. Mỗi nút lưu trữ được bao gồm một máy ổ cắm kép Bộ xử lý Intel Xeon X5670 với 6 lõi/ổ cắm và bộ nhớ 24 GB. Mỗi máy có một đĩa cục bộ và chạy RedHat 2.6.18-164.11.1.el5.crt1. Ở mức 300 GB, kích thước ổ đĩa là tài nguyên giới hạn tỷ lệ trên mỗi nút, quyết định cấu hình tổng thể, vì vậy chúng tôi đã định cấu hình mỗi nút để chứa 100M bản ghi, với kích thước khóa trung bình là 13 byte và kích thước dữ liệu là 1108 byte.

Hình 5 – Kết quả YCSB – Hiệu suất khi kích thước của cơ sở dữ liệu tăng lên

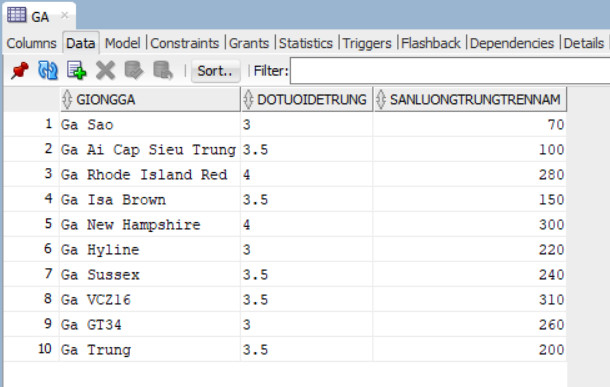
Hình 5 cho thấy hiệu năng chèn thô của Cơ sở dữ liệu Oracle NoSQL cho các cấu hình từ một hệ thống nhóm sao chép với ba nút lưu trữ 100 triệu bản ghi vào hệ thống có 32 nhóm sao chép trên 96 các nút lưu trữ 2,1 tỷ bản ghi (điểm chuẩn YCSB được giới hạn ở mức tối đa là 2,1 tỷ bản ghi). Đồ thị hiển thị cả thông lượng trong hoạt động mỗi giây (đường màu xanh lam và trục bên trái) và thời gian phản hồi tính bằng mili giây Tổng quan về cơ sở dữ liệu Oracle NoSQL 15 (đường màu đỏ và trục bên phải). Thông lượng của hệ thống quy mô gần như tuyến tính khi kích thước cơ sở dữ liệu và số nhóm sao chép tăng lên, chỉ tăng một chút thời gian phản hồi.

DEMO Báo Cáo Sản Phẩm

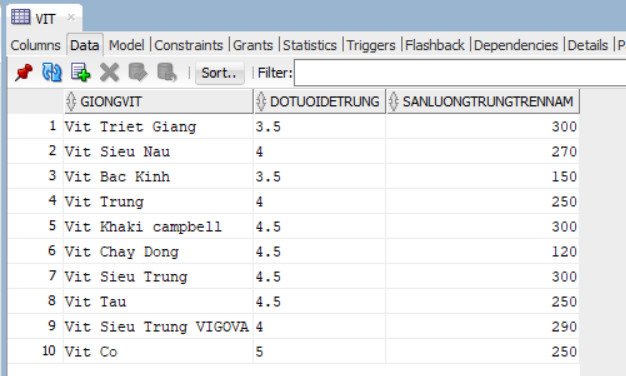
Dữ liệu của các Bảng.

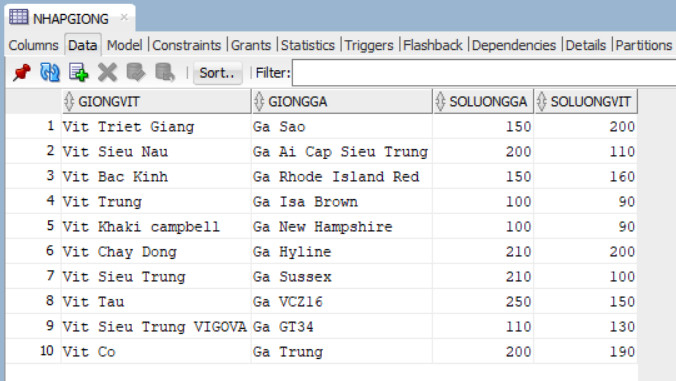


Bảng Hóa Đơn.

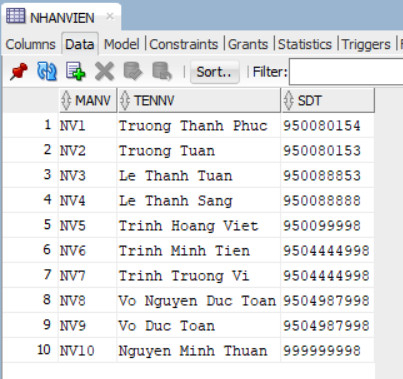


Bảng GÀ

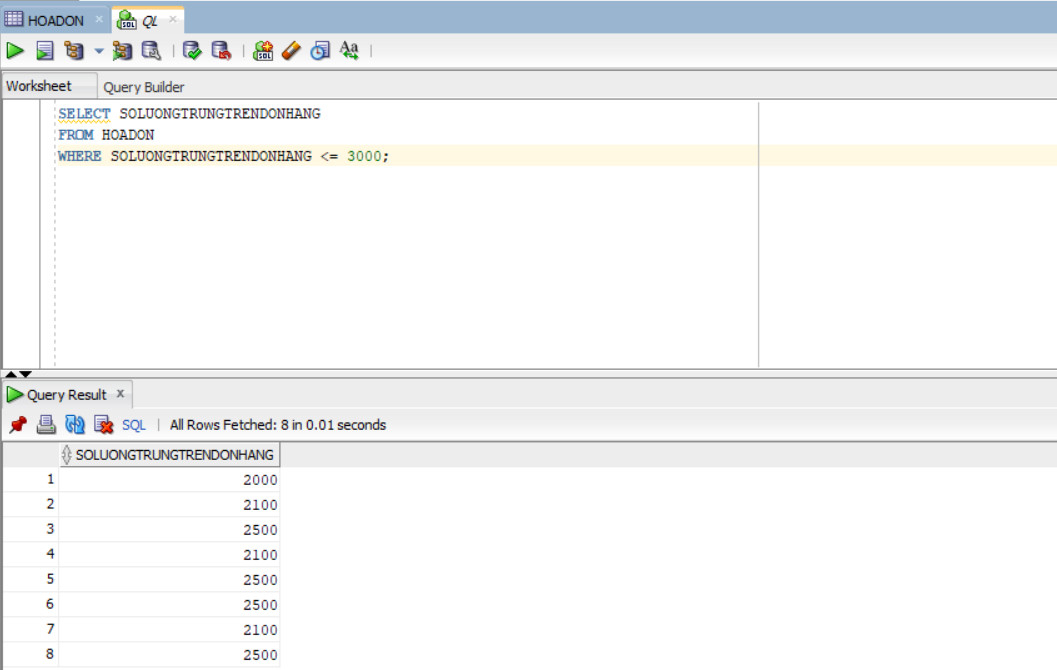
Bảng VỊT



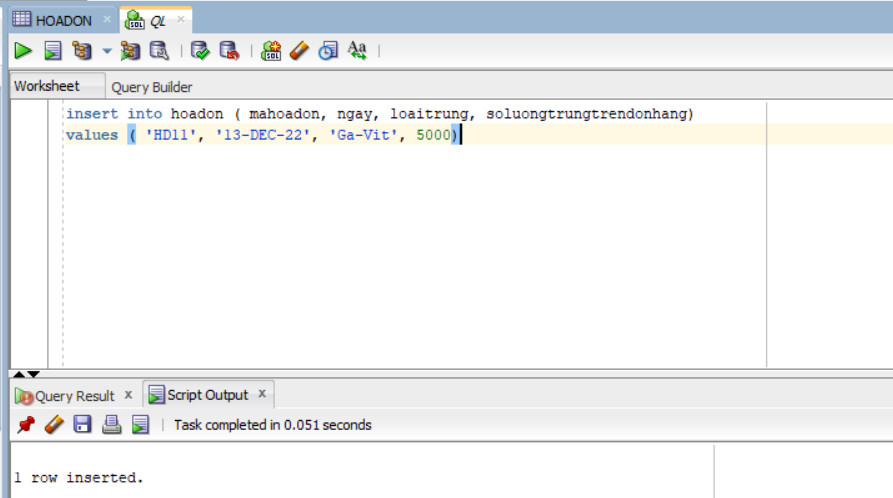
Bảng Nhập Giống.



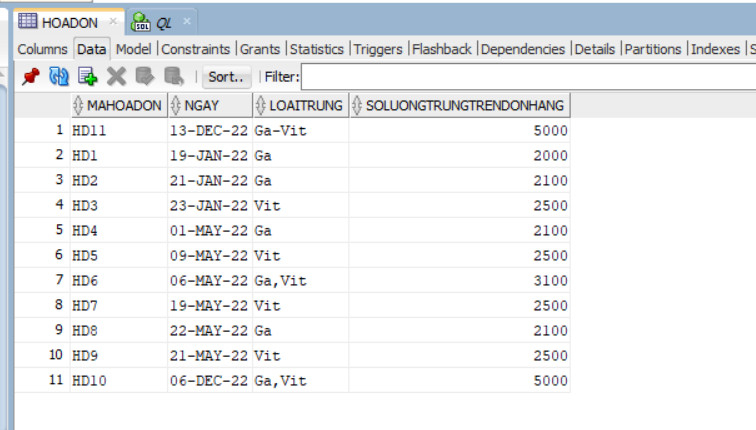
Bảng Nhân Viên



Toán tử so sánh



Thêm dữ liệu



CHƯƠNG 10: Kết luận

Cơ sở dữ liệu NoSQL của Oracle mang đến hiệu suất và lưu trữ chất lượng cao cho doanh nghiệp với tính sẵn sàng cao, rộng rãi môi trường NoSQL phân tán. Nó đã được chứng minh về mặt thương mại, hệ thống lưu trữ được tối ưu hóa ghi mang lại hiệu quả vượt trội hiệu suất cũng như độ bền và độ tin cậy, và thiết kế “Không có điểm hỏng hóc duy nhất” của nó đảm bảo rằng hệ thống tiếp tục chạy và dữ liệu đó vẫn có sẵn sau một loạt các khả năng lỗi.

LỜI CẢM ƠN

Nhóm em xin gửi lời cảm ơn đến Thầy Phan Trọng Huynh đã giúp đỡ nhóm em trong việc tổng hợp và khai thác thông tin một cách hiệu quả nhất. Cảm ơn thầy Phan Trọng Huynh đã tận tình giảng dạy và chỉ bảo trong suốt thời gian diễn ra môn học và giúp nhóm em hoàn thành tốt đề tài dự án Oracle NoSQL Database này.

Tuy đã có nhiều cố gắng nhưng do kiến thức và thời gian nghiên cứu có hạn nên bài tiểu luận của em không tránh khỏi những sai sót. Do đó, một lần nữa em rất mong nhận được sự góp ý của các thầy để em có điều kiện hoàn thiện hơn kiến thức của mình.

Sinh Viên Thực Hiện

Nhóm 7

**Tài liệu tham khảo**

Web

[1] [www.oracle.com](http://www.oracle.com)

[2] [www.sql.edu.vn](http://www.sql.edu.vn)

[3] [www.vietnix.vn](http://www.vietnix.vn)

Links:

<https://sites.google.com/site/congdongcvietnamvn/oracle-nosql>

<https://vietnix.vn/nosql-la-gi/>