**Hệ Thống Hỗn Hợp Của Cơ Sở Dữ Liệu Cho Lưu Trữ Dữ Liệu Lớn Và Quản Lý**

Blessing E. James and P.O.Asagba

Khoa Công Nghệ Thông Tin, University of Port Harcourt, Choba, Rivers State,  
Nigeria

**Mở đầu**

Hệ thống Cơ sở dữ liệu quan hệ đã trở thành một tiêu chuẩn lưu trữ trong hơn 40 năm.Nhưng hiện nay, các thành tựu trong lĩnh vực khoa học đã làm cho viêc lưu trữ dữ liệu gia tăng, vận tốc và sự đa dạng Ngoài những gì cơ sở dữ liệu quan hệ có thể xử lý . các nhà phát triển đang chuyển đần sang NoSQL, một loại cơ sở dữ liệu phi quan hệ. Công việc này được đề xuất một hệ thống cơ sở dữ liệu kết hợp

**1.Giới Thiệu**

Công nghiệp hóa nhanh chóng, khả năng có thể tri chả các thiết bị cao hơn, sự leo thang đột ngột trong việc sử dụng thiết bị di động và tiến bộ gần đây trong kiến ​​thức công nghệ chẳng hạn như kết nối của mọi đối tượng với Internet được gọi là Internet of Things (IoT), mesh or cloud computing, big user etc. đã dẫn đến một lĩnh vực dữ liệu khổng lồ không thể tưởng tượng được biết đến là dữ liệu lớn[18][19]. Nói chung là,dữ liệu lớn được coi là khối lượng khổng lồ (terabyte và pentabyte) dữ liệu bao gồm các kiểu dữ liệu (có cấu trúc, bán cấu trúc và không có cấu trúc) và tính khả dụng theo thời gian thực (velocity) cho nên dữ liệu đó trở thành bất khả thi để lưu trữ hoặc quản lý bằng các phương tiện như truyền thống các hệ thống cơ sở dữ liệu quan hệ. Nói một cách đơn giản, dữ liệu lớn là dữ liệu có khối lượng và bản chất (kiểu dữ liệu bán cấu trúc và phi cấu trúc) lớn hơn những gì hệ thống cơ sở dữ liệu thông thường có thể xử lý. Sự gia tăng khổng lồ về số lượng dữ liệu này đã mở ra những cơ hội tuyệt vời cho thành tựu khoa học, chiến lược kinh doanh cải tiến, phương pháp chăm sóc sức khỏe v.v. **[**14]

Hệ thống cơ sở dữ liệu quan hệ đã phục vụ như một hệ thống lưu trữ trong vài năm. Tuy nhiên,trong vòng bốn năm qua, đã có những giải pháp tuyệt vời trong thế giới máy tính làm giảm bớt sự phổ biến của cơ sở dữ liệu quan hệ và dẫn đến một lực đẩy gia tăng trong việc xem xét mô hình lưu trữ mới được gọi là NoSQL **[**17**]**. Điều này là do cơ sở dữ liệu quan hệ không bao giờ được thiết kế để lưu trữ hoặc quản lý khối lượng dữ liệu lớn như vậy, với tốc độ cao và đa dạng, dữ liệu phi cấu trúc hoặc tốc độ tăng trưởng nhanh [15]. Do đó, các doanh nghiệp đang chuyển sang một phương pháp lưu trữ mới gọi là NoSQL tìm giải pháp cho những thách thức vốn có trong dữ liệu lớn. Hầu hết NoSQL mở rộng theo chiều ngang với sự gia tăng về khối lượng dữ liệu và cũng đủ linh hoạt để giữ cấu trúc một phần và phân tán thu thập dữ liệu [11]. Với NoSQL, dữ liệu ở dạng thoại, video, email và tài liệu có thể được lưu trữ và quản lý đúng cách. Tuy nhiên, phương pháp này có hấp dẫn như thế nào thì nó không phải là không có mặt hạn chế. Các nhà đổi mới của NoSQL hoàn toàn hoặc có thể vô tình bỏ qua một số thành phần cơ sở dữ liệu mong muốn như giao dịch và bảo mật để đạt được những gì mà cơ sở dữ liệu quan hệ không thể cung cấp.

Trong công việc này, chúng tôi đã phát triển một hệ thống cơ sở dữ liệu lai sử dụng cơ sở dữ liệu MongoDB và MySQL đó là các biến thể phổ biến của cơ sở dữ liệu NoSQL và hệ thống cơ sở dữ liệu quan hệ tương ứng. Trước khi lưu trữ dữ liệu, dữ liệu được phân loại thành dữ liệu có cấu trúc và dữ liệu phi cấu trúc tùy thuộc vào bản chất của dữ liệu. Dữ liệu phi cấu trúc được lưu trữ và quản lý trong cơ sở dữ liệu MongoDB trong khi đó việc lưu trữ và quản lý dữ liệu có cấu trúc chặt chẽ được thực hiện bằng cơ sở dữ liệu MySQL. Hệ thống kết hợp của chúng tôi sao cho các cơ sở dữ liệu tạo nên hệ thống có thể hoạt động riêng biệt, chẳng hạn như hệ thống của chúng tôi có thể được sử dụng như một cơ sở dữ liệu MongoDB hoàn chỉnh và riêng biệt. Thay vì từ bỏ các chức năng của hệ thống cơ sở dữ liệu quan hệ cho cơ sở dữ liệu NoSQL, thì chúng tôi đã phát triển một phương pháp mang lại lợi ích của cả hai hệ thống trong một hệ thống cơ sở dữ liệu duy nhất.

**2. PHƯƠNG PHÁP LƯU TRỮ CƠ SỞ DỮ LIỆU QUAN HỆ**

Mô hình chi phối cơ sở dữ liệu quan hệ được xây dựng dựa trên các nguyên tắc quan hệ trong toán học. Nhiều cơ sở dữ liệu phổ biến và miễn phí nằm trong lớp này. Có thể lưu trữ dữ liệu trong các bảng hàng và cột trong khi vẫn giữ và áp đặt mối quan hệ giữa dữ liệu bị ảnh hưởng là một trong những đặc điểm cơ bản của cơ sở dữ liệu quan hệ [1].Dữ liệu là một trong những đặc điểm cơ bản của cơ sở dữ liệu quan hệ [1]. Các quy tắc cơ bản của cơ sở dữ liệu quan hệ[1] là:

* Dữ liệu và thông tin phải được lưu trữ trong bảng hàng và cột.
* Để truy nhập nội dung của cột,tên của bảng,cột và khóa chính phải được chỉ định.
* Các trường hợp nhập thiếu, không phù hợp phải được xử lý một cách có hệ thống khác với các mục nhập dự kiến và không phụ thuộc vào loại dữ liệu được nhập.
* Phải có ít nhất một ngôn ngữ được hệ thống quản lý cơ sở dữ liệu hỗ trợ có thể được sử dụng riêng biệt hoặc trong các chương trình.
* Hệ thống quản lý cơ sở dữ liệu phải có khả năng cập nhật dạng xem
* Các thao tác cơ bản như chèn, cập nhật và xóa phải được hỗ trợ bởi hệ thống quản lý cơ sở dữ liệu.
* Các sửa đổi được thực hiện trên cấu trúc logic như thêm hoặc xóa các cột từ bảng không được ảnh hưởng đến chế độ xem của người dùng.
* Các thay đổi ở cấp độ vật lý như lưu trữ không được ảnh hưởng đến toàn bộ ứng dụng.
* Các hạn chế liên quan đến Tính toàn vẹn nên được cách ly khỏi ứng dụng.
* Trong môi trường phân tán, tác động của việc phân phối cơ sở dữ liệu phải được nhìn thấy bởi người dùng.

Trong mô hình quan hệ, dữ liệu được thể hiện trong bảng hoặc quan hệ.Một bảng được hiển thị như trong bảng 1 chủ yếu là một tập hợp các mục nhập dữ liệu liên quan và nó được tạo thành từ một số cột và hàng được gọi là trường và bản ghi tương ứng. Hầu hết tất cả các cơ sở dữ liệu được xây dựng trên các giao dịch đảm bảo mô hình quan hệ Atomicity Consistency Isolation and Durability.

Ảnh có chứa bàn

Mô tả được tạo tự động

Bảng 1: Biểu diễn dữ liệu dạng bảng trong cơ sở dữ liệu quan hệ

3. **Cách tiếp cận lưu trữ cơ sở dữ liệu trên NoSQL**

Cơ sở dữ liệu NoSQL là cơ sở dữ liệu đối tượng được thiết kế để giải quyết các vấn đề được tạo ra bởi mở rộng khối lượng và tính đa dạng của dữ liệu, đặc biệt là trong các ứng dữ liệu lớn. Cơ sở dữ liệu NoSQL được coi là rất cần thiết trong tình huống khối lượng dữ liệu vượt xa những gì có thể xử lý bởi cơ sở dữ liệu quan hệ và cả thành phần thông tin không được lưu trữ trong cơ sở dữ liệu quan hệ. Cơ sở dữ liệu NoSQL được xây dựng trên mô hình phân tán để đảm bảo Các thuộc tính có sẵn, Trạng thái mềm, Tính nhất quán cuối cùng (BASE) [21]. Hình 1 cho thấy một mô hình của cơ sở dữ liệu NoSQL dựa trên tài liệu. Trong mô hình này thay vì lưu trữ dữ liệu trong các hàng và cột trong một bảng, dữ liệu có thể lưu vào một số cột trong bảng có thể được lưu trữ trong các tài liệu sau đó được nhóm lại thành các bộ sưu tập. Có bốn danh mục cơ bản của cơ sở dữ liệu NoSQL [7].

* **Kho Khóa-Giá Trị**: Dữ liệu được lưu trữ bằng hai mục được kết nối nhưng riêng biệt - một điểm đặc biệtmã định danh được gọi là khóa và một giá trị tương ứng có thể là dữ liệu hoặc con trỏ đếnvị trí của dữ liệu. Nó rất thích hợp cho các hệ thống dựa trên khóa. E.g. Dynamo, Riak
* **Kho Họ Cột**: Dữ liệu được tổ chức theo hàng và cột như trong hệ quản trị cơ sỡ dữ liệu quan hệ. E.g. Cassandra
* **Kho Biểu Đồ Quan Hệ** : cho các quá trình có thể được biểu diễn dưới dạng mối quan hệ với các yếu tố kết nối với nhau như mạng xã hội.
* **Kho Tài Liệu**: Dữ liệu được lưu trữ trong tài liệu. Thích hợp để lưu trữ tài liệu trong nhiều hình thức khác nhau. E.g. MongoDB

3.1 **Sức mạnh của cơ sở dữ liệu NoSQL và các giới hạn của cơ sở dữ liệu quốc gia**

Cơ sở dữ liệu NoSQL được phát triển để loại bỏ những hạn chế hoặc nhược điểm gặp phải trongsử dụng cơ sở dữ liệu quan hệ đặc biệt là trong các môi trường lưu trữ dữ liệu lớn. Như vậy, hầu hết các nhược điểm trong hệ thống lưu trữ quan hệ tạo thành thế mạnh hoặc lợi thế của hệ thống cơ sở dữ liệu NoSQL. Điểm yếu của cơ sở dữ liệu quan hệ và thế mạnh của cơ sở dữ liệu NoSQL phụ thuộc chủ yếu vào các tính năng được thảo luận trong các phần sau.

3.1.1 **Khả Năng Mở Rộng**

Trong hệ thống lưu trữ quan hệ, việc mở rộng đạt được bằng cách thay thế bộ lưu trữ hoặc máy chủ hiện có với một máy chủ lớn hơn (đắt hơn) có nghĩa là tăng sức ngựa của phần cứng máy chủ hiện có. Điều này được gọi là mở rộng quy mô theo chiều dọc hoặc mở rộng quy mô. Rõ ràng là khi khối lượng của dữ liệu tăng lên, nó có thể đi đến giai đoạn mà máy chủ giá cả phải chăng nhất có thể không đáp ứng được yêu cầu lưu trữ như được thể hiện trong hình 2, điều này có thể làm giảm hiệu suất của hệ thống. Và hệ thống cũng gặp khó khăn với một điểm lỗi duy nhất. Cơ sở dữ liệu NoSQL được xây dựng trên kiến trúc phân tán để phân vùng (phân vùng) cơ sở dữ liệu trên một số máy chủ là khả thi. Như vậy, việc mở rộng đạt được thông qua việc bổ sung các máy chủ rẻ tiền được kết nối với cụm cơ sở dữ liệu được hiển thị trong hình 3. Điều này được gọi là mở rộng quy mô theo chiều ngang hoặc mở rộng quy mô. Mở rộng quy mô theo chiều ngang giúp tăng hiệu suất của hệ thống với chi phí tối thiểu bằng cách thúc đẩy dữ liệu nhanh chóng mở rộng và loại bỏ điểm lỗi duy nhất tồn tại trong cơ sở dữ liệu quan hệ.

A picture containing table

Description automatically generated

Hình 1: Biểu diễn dữ liệu cơ sở tài liệu (Source: [23])

Diagram

Description automatically generated

Hình 2: Chia tỷ lệ dọc trong cơ sở dữ liệu quan hệ (Source [23])

**3.1.2 Khả Năng Linh Hoạt**

Cơ sở dữ liệu quan hệ là lược đồ bất khả tri; dữ liệu không thể được lưu trữ nếu như không xác định được lược đồ của dữ liệu đó. Như vậy, trong môi trường dữ liệu lớn nơi cần lưu trữ dữ liệu phi cấu trúc, đó là điều bất khả thi để biết trước lược đồ hoặc cấu trúc của dữ liệu. Mặt khác, NoSQL có lược đồ động cho nên lược đồ không thể được xác định trước. Như vậy, cơ sở dữ liệu NoSQL có thể được để lưu trữ cả dữ liệu có cấu trúc và không có cấu trúc.

**Điểm yếu của cơ sở dữ liệu NoSQL**

Giải quyết một số vấn đề trong cơ sở dữ liệu quan hệ đã giới thiệu một số điểm yếu nhất định trong NoSQLcơ sở dữ liệu. Một số điểm yếu của cơ sở dữ liệu NoSQL là:

* **Giao dịch phức tạp**: MongoDB không hỗ trợ các giao dịch đa tài liệu. Với tính khả dụng của cơ sở dữ liệu NoSQL, hỗ trợ các giao dịch ACID qua các tài liệu thường đã được từ bỏ. Loại trừ giao dịch ACID là một sự đánh đổi được NoSQL sử dụng để cung cấp giải pháp cho các vấn đề liên quan đến khả năng mở rộng.
* **Tính ổn định**: Một số cơ sở dữ liệu NoSQL vẫn đang trong giai đoạn tiền sản xuất và do đó không đủ ổn định hoặc trưởng thành cho một số nhiệm vụ nhạy cảm.
* **Hỗ trợ toàn cầu**: Doanh nghiệp yêu cầu hỗ trợ toàn cầu và dịch vụ từ các nhà cung cấp cơ sở dữ liệukhi một thành phần cốt lõi của hệ thống bị lỗi. NoSQL thiếu các dịch vụ như vậy cho doanh nghiệpkhách hàng.

Graphical user interface

Description automatically generated with medium confidence

Hình 3: Chia tỷ lệ theo chiều ngang trong cơ sở dữ liệu NoSQL (Source [24])

**4. Dữ Liệu Lớn**

Dữ liệu lớn chủ yếu đề cập đến nhóm dữ liệu đã trở nên cực kỳ lớn (petabytevà terabyte) bao gồm các kiểu dữ liệu khác nhau (có cấu trúc, bán cấu trúc và không có cấu trúc) vàtính khả dụng theo thời gian thực (tốc độ) như vậy nó không hiệu quả khi được lưu trữ hoặc xử lý vớicác công cụ hoặc phương tiện truyền thống như hệ thống cơ sở dữ liệu thông thường. Những công cụ truyền thống này đã được sử dụng trong nhiều năm để lưu trữ, xử lý và phân tích lượng thông tin khổng lồ trong các công ty, ngành công nghiệp, v.v. Dữ liệu lớn không chỉ đề cập đến việc thu thập dữ liệu khổng lồ, mà khối lượng cực lớn của dữ liệu có cấu trúc và dữ liệu phi cấu trúc có tốc độ thay đổi rất cao,bắt nguồn từ các con đường của diver có thể bao gồm email, phương tiện truyền thông xã hội, cuộc gọi điện thoại, v.v. Dữ liệu lớn là được sử dụng để mô tả một tập hợp dữ liệu phức tạp có khối lượng tăng liên tục và nhanh chóng với tốc độ mà việc sử dụng các công cụ quản lý thông thường để lưu trữ và phân tích trở nên kém hiệu quả và không phù hợp. Sự phức tạp trong khối lượng khổng lồ phân tán dữ liệu quy cho thực tế là dữ liệu được thu thập từ các nguồn khác nhau cũng có thể có và có thể cần phải tích hợp chúng thành một đơn vị để phân tích.

4.1 **Tính Chất Của Dữ Liệu Lớn**

Mặc dù từ lớn biểu thị kích thước của một thứ, nhưng trong dữ liệu lớn, lớn không giới hạn ở khối lượng dữ liệumà còn kết hợp các thuộc tính khác như Vận tốc và Sự đa dạng. Ba thuộc tính nàymô tả các thuộc tính chính của dữ liệu lớn được gọi là 3 V của dữ liệu lớn [22].

* **Khối lượng**: Đây đôi khi được coi là thuộc tính cuối cùng của dữ liệu lớn. Nó mô tả một lượng dữ liệu rất lớn và ngày càng tăng, từ terabyte (1012 byte) đến yotta-bytelà hàng nghìn tỷ gigabyte.
* **Tốc độ**: Tốc độ đề cập đến thời gian thực sẵn có của dữ liệu để xử lý. Dữ liệu lớn cũng là đặc trưng bởi sự xuất hiện tức thời của dữ liệu khổng lồ để xử lý. Nó đòi hỏi tỷ lệ tại đó dữ liệu được lưu chuyển trong hệ thống, ví dụ: vận tốc mà dữ liệu được lấy ngoài các hoạt động bên trong và bên ngoài và các nguồn như tương tác với máy móc,con người, phương tiện truyền thông xã hội, v.v.
* **Đa dạng**: Điều này thể hiện tính đa dạng của dữ liệu trong một tập dữ liệu. Dữ liệu lớn được tạo thành từ dữ liệu thu được từ nhiều nguồn khác nhau như email, máy móc, mạng xã hội, kinh doanh giao dịch, thiết bị di động, v.v. Dữ liệu từ các nguồn khác nhau giả định các dạng khác nhau như dưới dạng trang tính trải rộng, ảnh, video, v.v. Sự đa dạng như một thuộc tính của dữ liệu lớn mô tả các diver các dạng dữ liệu có nguồn gốc từ nhiều nguồn khác nhau.
* **Tính xác thực**: Đề cập đến tính trung thực của dữ liệu. Nó đề cập đến mức độ liên quan của dữ liệu(đã xử lý hoặc phân tích) cho nhiệm vụ đang làm. Nó cho thấy sự cần thiết phải tránh tích tụdữ liệu bẩn.
* **Tính biến động**: Thỏa thuận với tuổi thọ hợp lý của dữ liệu được lưu trữ, sử dụng trong thế giới dữ liệu thời gian thực . Nó điều tra tính hợp lệ của dữ liệu được lưu trữ đối với phân tích hiện tại.
* **Tính hợp lệ**: Các quyết định có giá trị như dữ liệu được sử dụng trong phân tích

**4.2 Những Thử Thách Của Dữ Liệu Lớn**

Những thử thách của dữ liệu lớn cũng có thể được gọi là các bước liên quan đến quá trình xử lý dữ liệu lớn này.khối lượng dữ liệu cực lớn của các loại diver và tốc độ sử dụng cao. Để tận dụng nhiều lợi ích của dữ liệu lớn, dữ liệu phải trải qua các quy trình sau [5], mặt khác từ, những thách thức sau đây phải được vượt qua;

* **Nhập**: Quy trình thu thập và nhập dữ liệu để sử dụng hoặc lưu trữ.Dữ liệu có thể được nhập sau khi được cung cấp bởi nguồn hoặc được nhóm thành các lô vànhập trong khoảng thời gian quy định. Quá trình này thường bắt đầu với việc xếp hạng các nguồn dữ liệu,xác thực từng tệp trước khi truyền dữ liệu đến đích chính xác.
* **Lưu trữ**: Lưu trữ rất phức tạp; nó bao gồm tìm kiếm và truy xuất dữ liệu và cũng có thể bao gồm các vấn đề bảo mật và quyền riêng tư phức tạp. Khung lưu trữ cho dữ liệu lớn có thể chứa khối lượng lớn dữ liệu có cấu trúc, bán cấu trúc và phi cấu trúc một cách hiệu quả.
* **Phân tích**: Dữ liệu lớn gần như vô dụng nếu không có các công cụ và quy trình phân tích hiệu quả thông qua đó thông tin hữu ích được trích xuất từ ​​những thứ dường như là dữ liệu rác.Phân tích trong lĩnh vực dữ liệu lớn cũng bao gồm các tính toán trên dữ liệu lớn. Nó bao gồm các khung và công cụ như Map Reduce và Hadoop được sử dụng để rút ra ý nghĩa từ dữ liệu lớn. Kết quả phân tích dữ liệu có thể được sử dụng để cải thiện các dịch vụ được cung cấp cho khách hàng, chiến lược tiếp thị và ra quyết định chung.
* **Trực quan hóa**: Điều này bao gồm việc trình bày dữ liệu để dễ dàng xác định các mẫu hoặc nắm bắ tcủa những khái niệm mới. Các công cụ và kỹ thuật trực quan hóa dữ liệu lớn cho phép biểu diễn dữ liệu trong dạng biểu đồ (đồ thị), biểu đồ, bản đồ và thậm chí cả video giúp dễ dàng tìm kiếm thông tin liên lạc.

**5. Công viẹc được lặp lại**

[13][2] đẵ thực hiện một số báo cáo về phân loại, thuộc tính và so sánh về co sở dữ liệu NoSQL .

[13][2] đă khám phá ra được điểm mạnh lẫn điểm yếu của các kiểu cơ sở dữ liệu NoSQL. [3] đã điều tra độ co dãn của cơ sở dữ liệu NOSQL dựa trên khả năng mở rộng của việc đọc và cập nhật hoạt động của cơ sở dữ liệu. Việc đó đã được chỉ ra rằng tốc độ thực hiện thao tác đọc trong Hbase là cao trong khi thao tác chèn nhanh trong Cassandra và Riak chậm ở cả việc đọc và ghi Hoạt động. [17] đã thực hiện so sánh chi tiết giữa MongoDB và Microsoft SQL Cơ sở dữ liệu. Microsoft SQL là một hệ thống cơ sở dữ liệu quan hệ, như vậy dung lượng lưu trữ của hệ thống chỉ có thể được tăng lên bằng cách thay đổi một máy chủ có dung lượng lớn hơn và điều này sẽ phát sinh thêm chi phí. Mặt khác, cơ sở dữ liệu NoSQL là hệ thống phi quan hệ có khả năng dễ dàng thay đổi quy mô theo chiều ngang để chứa nhiều dữ liệu hơn. Hệ thống được triển khai bằng Java, ngôn ngữ lập trình sử dụng Môi trường phát triển tích hợp Eclipse. Việc này đã được theo dõi và thấy là mặc dù MongDB và Microsoft SQL thực hiện các thao tác ghi nhanh hơn các thao tác đọc, thao tác ghi và đọc trong MongoDB gần như nhanh gấp mười lần việc đọc và viết trong cơ sở dữ liệu Microsoft SQL. Mặc dù MongoDB có khả năng đọc/ghi cao hơn, nhưng vẫn tồn tại các tình huống là tốc độ không phải là yêu cầu cuối cùng hoặc duy nhất cho cơ sở dữ liệu. MongoDB không thích hợp cho các tác vụ truyền dẫn lớn. Hiệu suất của MongoDB, PostgreSQL và Cassandra được đánh giá bằng cách [4] đã cho thấy rằng Cassandra thích hợp hơn cho các hệ thống cảm biến phân tán lớn. [6]Khi thực hiện công việc trên các mối quan hệ của cơ sở dữ liêu quan hệ thì việc lưu trữ và quản lý dữ liệu phụ thuộc nhiều hơn vào kỹ thuật lập trình của kỹ sư và lập trình viên.

**6. Hệ thống đề xuất**

Mục đích của hệ thống đề xuất là thiết kế một Hệ thống Cơ sở dữ liệu Kết hợp để lưu trữ và quản lý dữ liệu lớn. Hệ thống kết hợp được tạo thành từ cơ sở dữ liệu MySQL và MongoDB là các máy chủ cơ sở dữ liệu quan hệ và NoSQL (không quan hệ) phổ biến nhất. Dữ liệu được nhóm thành danh mục dữ liệu có cấu trúc và phi cấu trúc, dữ liệu có cấu trúc được chuyển vào cơ sở dữ liệu MongoDB, trong khi việc lựa chọn cơ sở dữ liệu cho dữ liệu phi cấu trúc phụ thuộc vào chế độ mà ứng dụng chạy trong đó; đây có thể là MongoDB cho chế độ kết hợp và MySQL cho chế độ SQL. Các cơ sở dữ liệu được tích hợp trong hệ thống đề xuất cũng có thể hoạt động riêng biệt. [8] Đã trình bày tổng quan về cơ sở dữ liệu NoSQL hiện có sử dụng mô hình dữ liệu, mô hình truy vấn, mô hình nhân rộng và mô hình nhất quán.

**6.1 Thiết kế hệ thống đề xuất**

Thiết kế hệ thống cho thấy các thành phần tạo nên một hệ thống. Hệ thống đề xuất bao gồm các thành phần cơ bản sau đây:

* Cơ sở dữ liệu MySQL(SQL Component): chứa các công cụ để thực hiện các thao tác lưu trữ và xử lý lưu trữ dữ liệu trong cơ sở dữ liệu MySQL. Các công cụ lưu trữ được tạo thành từ một tệp nhật ký hoạt động và các nhóm tệp dữ liệu này có thể được chia nhỏ theo thứ bậc thành bảng tệp dữ liệu, chỉ mục, phạm vi và trang là đơn vị lưu trữ nhỏ nhất trong cơ sở dữ liệu quan hệ.
* cơ sở dữ liệu MongoDB(MongoDB Component): MongoDB sử dụng bản sao để đảm bảo dự phòng và nhất quán. Luồng dữ liệu từ các đích đến khác nhau và có các định dạng khác nhau được chia nhỏ và phân tán đồng đều cho một tập hợp các thiết bị đầu cuối có thể mở rộng không tĩnh được gọi là phân đoạn. Dữ liệu mô tả dữ liệu khác trong cụm được lưu trong cấu hình máy chủ. Mỗi máy chủ này chứa bản sao của tất cả siêu dữ liệu nhằm mục đích dự phòng. Khi client thực hiện yêu cầu, nó tạo thành một trong các quy trình định tuyến được sử dụng để kiểm tra các cấu hình máy chủ để biết vị trí của yêu cầu.

Các thành phần này sẽ được mô tả chi tiết hơn và mô hình thiết kế kiến trúc của hệ thống đề xuất được mô phỏng qua hình sau cho thấy cách các thành phần này kết nối với nhau.

**6.2 Thuật toán**

Trong khi, hệ thống dữ liệu quan hệ của chúng tôi sử dụng cả hai thuật toán B-tree và tỉ lệ thuật toán được thêm vào. Trong khi cơ sở dữ liệu quan hệ (SQL) chạy chủ yếu trên tỉ lệ thuật toán đã được thêm vào. Tỷ lệ thêm vào của thuật toán được sử dụng công cụ lưu trữ của sql để viết dữ liệu vào tệp của cơ sở dữ liệu dựa trên bộ nhớ trống trong từng tệp dữ liệu một thay vì trực tiếp viết vào mỗi tệp khác nhau cho tới khi đầy sau đó tới cái thứ hai một cách tuần tự. Như vậy, công cụ lưu trữ SQL server sẽ thường xuyên ghi hơn vào các tệp có nhiều dung lượng trống hơn. MongoDB mặt khác được xây dựng dựa trên thuật toán B-tree, tuy nhiên , quy trình từng bước để xử lý dữ liệu trong hệ thống kết hợp của chúng tôi được biểu diễn qua những bước sau:

Bước 1: Tải dữ liệu (Load data)

Bước 2: định nghĩa class của dữ liệu( define class of data)

Bước 3: khởi tạo giao diện hệ thống kết hợp của cơ sở dữ liệu( Initialize DB hybrid Interface )

Bước 4: kiểm thử dữ liệu:

Nếu :

* + - dữ liệu có cấu trúc xác định thì được lưu trong cơ sở dữ liệu SQL
    - dữ liệu không có cấu trúc xác định thì được lưu trong cơ sở dữ liệu MongoDB

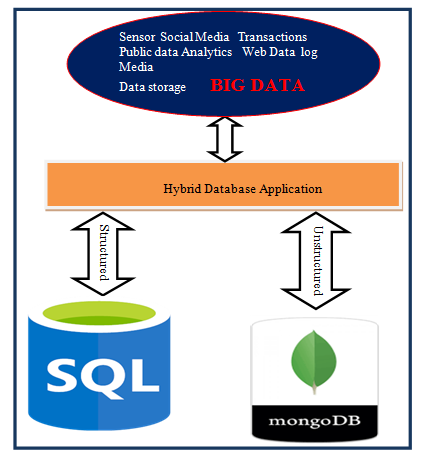
**7. Cài đặt cục bộ và kết quả thực hiện**

Hệ thống được đề nghị hoạt động và phán triển trên môi trường C#.của ứng dụng lập trình và phát triển visual studio. MongoDB và MySQL đều được sử dụng để lưu trữ dữ liệu. một số lớp(thành phần, classes) đã được sử dụng trong chương trình theo thứ tự là con người, biểu tượng, thông tin, bảng điều khiển , con người.

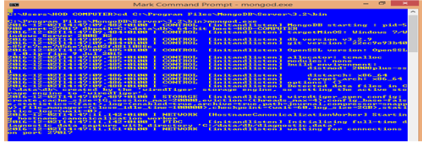
Để ứng dụng của hoạt động bình thường, mongoDB server trước tiên sẽ được khởi động từ cửa sổ lệnh bằng cách điều hướng đến thư mục cài đặt và thực hiện lệnh Mongod.exe như trong Hình 4. Theo mặc định, máy chủ MongoDB sẽ khởi động tại cổng 27017. Để bắt đầu thực hiện trên máy khách bằng cửa sổ lệnh, lệnh Mongo.exe được thực thi trên một cửa sổ nhắc lệnh riêng biệt.

Để thực thi phần mềm ứng dụng lai của chúng ta, dữ liệu đầu vào được tạo thành từ dữ liệu có cấu trúc(structured) và phi cấu trúc(unstructured) Dữ liệu được lưu trong ổ đĩa cục bộ của hệ thống. Ta có thể truy cập ứng dụng của mình bằng cách nhấp đúp vào biểu tượng ứng dụng. Tổng quan về ứng được hiển thị trong hình.

Khi dữ liệu lớn được tải trong hệ thống lai tạo cơ sở dữ liệu được sử dụng để lưu trữ được xác định bởi chế độ mà ứng dụng chạy trong đó. Dữ liệu phi cấu trúc được chuyển đến cơ sở dữ liệu MongoDB (trừ khi ứng dụng ở chế độ MongoDB, trong trường hợp đó MongoDB lưu trữ cả hai dữ liệu có cấu trúc và phi cấu trúc) trong khi cơ sở dữ liệu MySQL được sử dụng để lưu trữ và quản lý có cấu trúc dữ liệu.



Hình 3. Thiết kế kiến ​​trúc của hệ thống được đề xuất



Hình 4: khởi động MongoDB

Text

Description automatically generated

Hình 5: kết nối màn hình làm việc của hệ thống của MongoDB tới hệ thống cơ sở dữ liệu hệ thống mặc định

Graphical user interface, text, application

Description automatically generated

Hình 6: Tổng quan về giao diện và các chế độ hoạt động của DB\_Hybird có thể thực hiện

Graphical user interface, application, table, Excel

Description automatically generated

Hình 7: cho thấy cách tổ chức dữ liệu ở chế độ SQL

Graphical user interface, table

Description automatically generated

Hình 8: cho thấy cách tổ chức dữ liệu trong chế độ lai tạp

**8. THẢO LUẬN VỀ KẾT QUẢ**

Chúng tôi đã tải dữ liệu lớn trong hệ thống cơ sở dữ liệu kết hợp của mình trong chế độ SQL, chế độ MongoDB và cả chế độ kết hợp. Cơ sở dữ liệu được sử dụng để lưu trữ và quản lý dữ liệu khác nhau tùy thuộc vào chế độ mà hệ thống chạy. Đầu ra của hệ thống cơ sở dữ liệu kết hợp được triển khai để lưu trữ và quản lý dữ liệu lớn được thảo luận ở đây;

Trong chế độ SQL, dữ liệu được lưu trữ và quản lý trong cơ sở dữ liệu MySQL. Hệ thống loại bỏ dữ liệu trong biểu mẫu phi cấu trúc vì nó không thể được lưu trữ trong cơ sở dữ liệu SQL. Ảnh chụp màn hình của hoạt động tải dữ liệu được thực hiện bởi hệ thống kết hợp của chúng tôi ở chế độ SQL được đưa ra trong hình 7, dạng xem nội dung cơ sở dữ liệu được hiển thị trong hình 8 cho thấy rằng ứng dụng kết hợp của chúng tôi loại bỏ dữ liệu phi cấu trúc ở chế độ SQL. Trong chế độ MongoDB, việc lưu trữ và quản lý cả dữ liệu có cấu trúc và phi cấu trúc được thực hiện bằng cơ sở dữ liệu MongoDB. Cái này lưu trữ cả dữ liệu có cấu trúc và phi cấu trúc trong MongoDB.Và cả chế độ kết hợp, dữ liệu ở dạng có cấu trúc được lưu trữ và quản lý bằng cơ sở dữ liệu SQL trong khi MongoDB được sử dụng để lưu trữ và quản lý dữ liệu phi cấu trúc, điều này được thể hiện trong hình.9

Tạp chí Quốc tế về Khoa học Máy tính, Kỹ thuật và Ứng dụng (IJCSEA) Tập 7, Số 3/4, Tháng Tám 2017

Từ đầu ra trong hình 7, hình 8 và hình 9 , có thể thấy rằng hệ thống lai của chúng tôi tích hợp các chức năng hoặc môi trường của MySQL là một cơ sở dữ liệu quan hệ phổ biến và MongoDB là một cơ sở dữ liệu phi quan hệ trong một hệ thống cơ sở dữ liệu cho phép các cơ sở dữ liệu trong hệ thống hoạt động riêng biệt và cũng trong tích hợp.

Hệ thống được đề xuất hỗ trợ các hoạt động quản lý như Cập nhật và Xóa có thể được sử dụng để quản lý dữ liệu trong hệ thống. Hình 10 cho thấy hoạt động cập nhật trong hệ thống cơ sở dữ liệu kết hợp của chúng tôi.

Ảnh có chứa bàn

Mô tả được tạo tự động

Hình 9. Cập nhật hoạt động

Graphical user interface, application

Description automatically generated

Hình 10. Tải hoạt động dữ liệu ở chế độ SQL.

Tạp chí Quốc tế về Khoa học Máy tính, Kỹ thuật và Ứng dụng (IJCSEA) Tập 7, Số 3/4, Tháng Tám 2017

**9. KẾT LUẬN VÀ KHUYẾN NGHỊ**

Tóm lại, chúng tôi đã đề xuất một phương pháp kết hợp cơ sở dữ liệu SQL thuộc nhóm hệ thống cơ sở dữ liệu quan hệ và MongoDB là cơ sở dữ liệu NoSQL để lưu trữ và quản lý dữ liệu lớn. Với kết quả thu được, có thể hiểu được rằng hệ thống của chúng tôi có thể được sử dụng để lưu trữ và quản lý lớn loại bỏ các điểm yếu trong cả hai cơ sở dữ liệu.

10. **Sự Đóng Góp Cho Tri Thức**

Những công việc sau đây trình bày những đóng góp cho tri thức:

1. Phát triển hệ thông cơ sở dữ liệu cho hệ thống lưu trữ dữ liệu lớn và quản lý.
2. Cách tiếp cận này cải thiện việc sử dụng cơ sở dữ liệu MongoDB để lưu trữ dữ liệu lớn.
3. Nghiên cứu thiết lập khả năng để có tính linh hoạt và khả năng mở rộng của cơ sở dữ liệu NoSQL và cả tính ổn định và các thành phần giao dịch của cơ sở dữ liệu quan hệ trong một hệ quản trị cơ sở dữ liệu.

**Tài liệu tham khảo**

[1] Codd, E. F. (1970). “A relational model of data for large shared data banks”, Communications of theACM Vol. 13, No. 6, pp 377–387.

[2] Hecht, R., Jablonski, S. (2011) “ NoSQL evaluation: A use case oriented survey” Cloud and ServiceComputing (CSC) International Conference. pp336 – 341

[3] Konstantinou, I., Angelou, E., Boumpouka, C., Tsoumakos, D. and Koziris, N. (2011) “On theelasticity of NoSQL databases over cloud management platforms” In Proceedings of the 20th ACMinternational conference on Information and knowledge management (CIKM '11), New York, NY,USA

[4] Van der Veen, J.S., van der Waaij, B.,Meijer, R.J.(2012) “Sensor Data Storage Performance: SQL orNoSQL, Physical or Virtual” IEEE 5th International Conference on Cloud Computing (CLOUD) pp.431 – 438.

[5] Singh, N., Garg,N.& Mittal,V.(2012). “Big data- Insight, Motivation and Challenges” Internal Journalof Scientific and Engineering Research, Vol. 4

[6] Schram, A., Anderson, K. M. (2012) “MySQL to NoSQL: data modelling challenges in supportingscalability” Proceedings of the 3rd annual conference on Systems, programming, and applications:software for humanity (SPLASH '12). ACM, New York, NY, USA, pp. 191-202.

[7] Indrawan-Santiago, M. (2012). “Database Research: Are We at a Crossroad? Reflection on NoSQL.”Proceedings of the 15th International Conference on Network-Based Information Systems, pp 45-51.

[8] Tauro Clarence T.,Patil,Baswanth R., Prashant K.V.(2013). “A comparative analysis of differentNoSQL databases on data model, query model, and replication model” Internal Conference onEmerging Research in Communication and Application ERCICA. Bangalor India

[9] Nayak, A. ,Poriya ,A.,Dikshay Poojary (2013)” Types of NoSQL databases and its comparison withrelational databases”. International Journal of Applied Information Systems Vol.5, No. 4 pp 16-19

[10] Grolinger, K.,Higashinow,T. Wari,Capretz,M.AM (2013)”Data Management in cloud environments:NoSQL and NewSQL data stores” Vo l2. No. 22

[11] Porkony, J. (2013). “NoSQL databases: A step to database scalability in web environment.”,International Journal of Web Information Systems • Vol. 9, No.1 pp 69-82.

[12] Wu, L., Yuan,L., Huai, Y. (2013). “Survey of large scale data management system for big dataapplications” Journal of Computer Science and Technology. Vol. 30 pp 163-183.

[13] Moniruzzaman, A .B., Hossain, S.A.(2013). “NoSQL database: New era of databases for big dataanalytics-classification, characteristics and comparison” International Journal of Computer Scienceand Technology. Vol. 6 No. 4

[14] Nawsher Khan, Ibrar Yaqoob, Ibrahim Abaker Targio Hashem, Zakira Inayat,Waleed KamaleldinMahmoud Ali,Muhammad Alam,Muhammad Shiraz,Abdullah Gani (2014). “Big Data: Survey,Technologies, Opportunities, and Challenges” The Scientific World Journal Vol. 2014 , No. 712826

[15] Abramova, V., Bernardino,J., Furtado,P.(2014). “Experimental evaluation of: NoSQL databases”International Journal of Database Management System .Vol.6, No. 3

[16] Abramova, V., Bernardino,J., Furtado,P.(2014). “Which NoSQL database? A performance overview”Open Journal of Databases (OJDB). Vol. 1, Issue 2

[17] Wu, L., Yuan, L., You, J. (2014). “BASIC: An alternative to BASE for large scale data managementsystems”, https://webdocs.cs.ualberta.ca/~yuan/papers/ieee\_bigdata.pdf

[18] Manoj,V.(2014) “Comparative study of NoSQL document, column store databases and evaluation ofCassandra” International Journal of Database Management Systems ( IJDMS) Vol.6, No.4

[19] Rakesh, K., Shilpi, C., Somya, B(2015).”Effective way to handling big data problems using NoSQLdatabase (MongoDB)” Journal of Advanced Database Management & Systems. Vol. 2, pp 42–48.

[20] Wu,C.M.,Huang,Y.F.,Lee,J.(2015). “Comparisons between MongoDB and MS-SQL databases on theTWC website” American Journal of Software Engineering and Applications. Vol. 4 No.2 pp 35-41.

[21] GC, Deepak (20016) "A Critical Comparison of NOSQL Databases in the Context of Acid and Base"Culminating Projects inInformation Assurance. Paper 8.

[22] Mukherjee, S.,Shaw R.(2016) “Big data concepts, applications, challenges and future scope”International Journal of Advanced Research in Computer and Communication Engineering Vol. 5,Issue 2

[23]

http://refreshmymind.com/wpcontent/uploads/2016/02/nosql\_documents\_stores\_mongodbcouchdb.png

[24] http://refreshmymind.com/wpcontent/uploads/2016/02/nosql\_documents\_stores\_mongodbcouchdb.png

[25] Why NoSQL, http://img.blog.csdn.net/20151001143133915

**Tác giả**

Blessing E.James là Trợ lý sau đại học tại Khoa Khoa học Máy tính, Đại học Bang Akwa Ibom, Cô có bằng Cử nhân Công nghệ về Toán học và Khoa học Máy tính, năm 2010 của Đại học Công nghệ Liên bang Owerri, Bang Imo, Nigeria và bằng Thạc sĩ Bằng khoa học (2007) của Đại học Port Harcourt, Nigeria. Mối quan tâm của cô là về Dữ liệu lớn(bigdata), Hệ thống quản lý cơ sở dữ liệu(Database Management Systems), Quản lý tri thức cơ sở dữ liệu(Database Knowledge Management), Khai thác dữ liệu(data minning) và Học máy(machine learning).

A person wearing a hat and tie

Description automatically generated with low confidenceHoàng tử Oghenekaro ASAGBA đã có bằng Cử nhân Khoa học Máy tính tại Đại học Nigeria, Nsukka, vào năm 1991 với Hạng hai (Hons) Cấp trên. Ông có bằng Thạc sĩ Khoa học Máy tính tại Đại học Benin vào tháng 4, 1998, và bằng Tiến sĩ Khoa học Máy tính tại Đại học Port Harcourt vào tháng 3 năm 2009. Asagba là một nhà nghiên cứu nhiệt tình với hơn 50 bài báo nghiên cứu được xuất bản trên các tạp chí uy tín, cả trong nước và quốc tế. Asagba sở hữu hơn 20 năm kinh nghiệm nghiên cứu / giảng dạy ở cấp độ Đại học. Ông là Giáo sư / Học giả thỉnh giảng cho một số Các trường đại học ở Nigeria. Lĩnh vực nghiên cứu của ông bao gồm: Máy tính và Bảo mật Thông tin(Computing and Information Security), Phân tích Mạng(Network Analysis), Kỹ thuật Phần mềm(Software Engineering), Hệ thống Quản lý Cơ sở Dữ liệu(Database Management Systems), Mô hình hóa và Lập trình(Modelling and Programming). Ông là thành viên của Hiệp hội Máy tính Nigeria (NCS) và Hội đồng Đăng ký Chuyên nghiệp Máy tính của Nigeria

(CPN)