**Cơ sở dữ liệu SQL và NoSQL**

***Tóm tắt: NoSQL (Không chỉ SQL) là một cơ sở dữ liệu được sử dụng để lưu trữ một lượng lớn dữ liệu. Cơ sở dữ liệu NoSQL được phân phối, phi quan hệ, mã nguồn mở và có thể mở rộng theo chiều ngang (theo cách tuyến tính). NoSQL không tuân theo thuộc tính của ACID như chúng ta theo dõi trong SQL. Trong bài nghiên cứu này, chúng tôi đang khảo sát về NoSQL, nền tảng của nó, các nguyên tắc cơ bản như định lý ACID, BASE và CAP. Cũng trên cơ sở định lý CAP, nghiên cứu được thực hiện về các loại kho dữ liệu NoSQL khác nhau với các ví dụ, đặc điểm và ưu nhược điểm của NoSQL.***

***Từ khóa: ACID, BASE, SQL, NoSQL, CAP, CURD.***

**I. GIỚI THIỆU**

NoSQL là viết tắt của Không chỉ SQL. Nó được phát âm là **noseequel**. Đây là một trong những loại lưu trữ dữ liệu khác ngoài cơ sở dữ liệu (đã được sử dụng trước đó) được sử dụng để lưu trữ lượng lưu trữ dữ liệu khổng lồ như dữ liệu trong facebook (không ngừng tăng lên từng ngày). NoSQL là một hệ thống quản lý cơ sở dữ liệu phi quan hệ (đôi khi được gọi là bắt nguồn từ cơ sở dữ liệu quan hệ), cơ sở dữ liệu truy xuất thông tin nhanh và có tính di động. NoSQL về cơ bản bắt nguồn từ hệ thống cơ sở dữ liệu RDB. Cơ sở dữ liệu này thường tương tác với hệ điều hành UNIX. Cơ sở dữ liệu NoSQL là những cơ sở dữ liệu phi quan hệ, mã nguồn mở, phân tán trong tự nhiên cũng như nó có hiệu suất cao theo cách tuyến tính có thể mở rộng theo chiều ngang. Cơ sở dữ liệu phi quan hệ không tổ chức dữ liệu của nó trong các bảng liên quan (tức là dữ liệu được lưu trữ theo cách không chuẩn hóa). Cơ sở dữ liệu NoSQL là mã nguồn mở; Do đó, mọi người có thể xem xét mã của nó một cách tự do, cập nhật nó theo nhu cầu của mình và biên dịch nó. Phân tán có nghĩa là dữ liệu được lan truyền đến các máy khác nhau và được quản lý bởi các máy khác nhau nên ở đây nó sử dụng khái niệm sao chép dữ liệu. NoSQL có thể được biểu diễn một cách tượng trưng như trong hình 1:



Hình 1: Biểu diễn tượng trưng của NoSQL

Hình 1 cho biết việc truy vấn cơ sở dữ liệu mà không có bất kỳ tương tác hoặc giao diện nào của ngôn ngữ SQL. Đường nghiêng trong hình hiển thị việc sử dụng cơ sở dữ liệu mà không cần sử dụng SQL (Ngôn ngữ truy vấn có cấu trúc). Vì vậy, để truy cập các cơ sở dữ liệu này, chúng ta có thể sử dụng một số định dạng khác như XML để lưu trữ và truy xuất thông tin từ cơ sở dữ liệu.

Với sự ra đời của các trang mạng xã hội như Facebook và Twitter, nhu cầu về công nghệ mới có thể xử lý lượng dữ liệu khổng lồ đã dẫn đến sự xuất hiện của nhiều công nghệ mới khác nhau và một trong những công nghệ nổi bật là NoSQL khá hữu ích trong việc lưu kho dữ liệu. NoSQL (phi quan hệ) tương đối nhanh hơn cơ sở dữ liệu quan hệ. Trước đây, trong SQL, chúng tôi đã sử dụng ngôn ngữ Query để tìm nạp cũng như lưu trữ dữ liệu; đối với NoSQL, chúng tôi lưu trữ các thực thể dữ liệu lớn bằng cách sử dụng tài liệu ở định dạng XML (Ngôn ngữ đánh dấu có thể mở rộng). Ngôn ngữ XML về cơ bản được sử dụng để lưu trữ dữ liệu có cấu trúc ở dạng con người có thể đọc được. Điều này được thảo luận thêm trong các phần sau.

**II. BỐI CẢNH**

Vào những năm 1960, cơ sở dữ liệu Đa giá trị đã được phát triển để thực hiện một số hoạt động cơ sở dữ liệu phân cấp bằng cách sử dụng lưu trữ cây B +. Theo Scott Jones M[umps] được phát triển vào năm 1966 để hoạt động dựa trên các loại cơ sở dữ liệu như vậy (phân cấp). Năm 1977,

***© 2012, IJARCSSE All Rights Reserved Page | 20***

***Vatika Sharma và cộng sự, Tạp chí quốc tế về nghiên cứu advenced trong khoa học máy tính và kỹ thuật phần mềm 2 (8), tháng 8- 2012, trang 20-27***

M[umps] đã được phê duyệt là ngôn ngữ tiêu chuẩn ANSI. Năm 1979, Ken Thompson đã phát triển DBM (Trình quản lý cơ sở dữ liệu) sử dụng các kỹ thuật băm để cung cấp cho người dùng khả năng truy xuất dữ liệu nhanh hơn. Trong những năm 1980, số lượng lớn những người kế nhiệm DBM đã được phát triển như sau:

∙ TDBM là một cơ sở dữ liệu kiểu DBM hỗ trợ các giao dịch nguyên tử (tuân theo các thuộc tính ACID A: Atomicity,

C: Consistency, I: Isolation, D: Durability). Những ưu điểm của TDBM bao gồm:

(1) Nếu ứng dụng đang sử dụng TDBM, trong trường hợp có hư hỏng hoặc sự cố, nội dung được lưu trữ trong cơ sở dữ liệu sẽ được an toàn.

(2) TDBM cũng hỗ trợ các giao dịch nguyên tử lồng nhau bằng cách sử dụng các hoạt động đa luồng.

SDBM là cơ sở dữ liệu loại DBM được sử dụng vì một số lý do cấp phép.

∙ GT. M là một sự kế thừa khác của DBM tập trung vào xử lý giao dịch có hiệu suất cao. Năm 2000, DBM này có nguồn mở [4].

Sự phát triển của cơ sở dữ liệu SQL bắt đầu vào cuối những năm 1990. Sau một vài năm, nó đã trở thành một đối thủ cạnh tranh nghiêm trọng với RDBMS (Hệ thống quản lý cơ sở dữ liệu quan hệ) [4]. Trong năm 2009 và 2010 đã có các hội nghị NoSQL được tổ chức như NoSQL live, NoSQL eu, NoSQL east, v.v. Tên NoSQL này lần đầu tiên được Carlo Strozzi sử dụng vào năm 1998 làm tên của tệp mà ông đang phát triển cho cơ sở dữ liệu của mình.

Bây giờ một ngày, NoSQL đang trở nên rất phổ biến do lưu trữ cao và cũng bởi vì các thuộc tính của nó tránh các tính năng cơ bản của SQL.

Cơ sở dữ liệu quan hệ được thiết kế để chạy trên một máy duy nhất, vì vậy để làm được điều này, chúng ta cần một máy lớn để mở rộng quy mô. Một giải pháp để khắc phục điều này là mua nhiều máy móc (và biến chúng thành một cụm) để lưu trữ dữ liệu. Nó sẽ là một quá trình rẻ hơn cũng như có thể mở rộng theo chiều ngang (cách tuyến tính). Ngay cả khi một máy bị hỏng, độ tin cậy tổng thể của cụm là khá cao. Do đó, sự thống trị của SQL đang giảm dần và NoSQL đang trở nên phổ biến.



Hình 2: Sự suy giảm sự thống trị của SQL

**III. TIỀN ĐỀ CỦA NoSQL**

**A. ACID miễn phí**

ACID là viết tắt của Atomicity, Consistency, Isolation và Durability. Khái niệm ACID về cơ bản đến từ môi trường SQL. Nhưng trong NoSQL, chúng tôi sẽ không sử dụng khái niệm ACID vì tính năng Nhất quán của SQL [6]. Trong bài viết này, chúng ta sẽ thấy khái niệm ACID tạo ra vấn đề như thế nào cho NoSQL.

Như trong môi trường phân tán, dữ liệu được lan truyền đến các máy khác nhau, mỗi máy lưu trữ dữ liệu của nó và cần duy trì tính nhất quán. Ví dụ: nếu có sự thay đổi trong một tupple của bảng thì cần phải thay đổi trong mỗi và mọi máy mà dữ liệu cụ thể đó nằm trên đó. Nếu thông tin liên quan đến sự cập nhật lan truyền ngay lập tức, thì tính nhất quán sẽ được đưa ra; Nếu không, thì sự không nhất quán được thực hiện.

**B. CƠ SỞ**

BASE là viết tắt của Về cơ bản, Có sẵn, Trạng thái mềm và Tính nhất quán cuối cùng. BASE ngược lại với ACID [1]. Cơ sở dữ liệu NoSQL được phân chia giữa các con đường từ ACID đến BASE. Sau khi thống nhất giao dịch, trạng thái mà chúng ta sẽ nhận được là trạng thái mềm không phải là trạng thái rắn. Trọng tâm chính dẫn đầu đằng sau BASE là tính khả dụng vĩnh viễn [6].

Ví dụ, suy nghĩ về cơ sở dữ liệu trong các ngân hàng, nếu hai người đang truy cập vào cùng một tài khoản ở các thành phố khác nhau thì việc cập nhật dữ liệu là cần thiết không chỉ kịp thời mà còn cần một số cơ sở dữ liệu thời gian thực. Những cập nhật đó cần được thực hiện thường xuyên trên tất cả các máy. Một số ví dụ khác là đặt chỗ đường sắt trực tuyến, buôn bán sách trực tuyến, v.v.

**C. Giới hạn**

CAP là viết tắt của Tính nhất quán, Tính khả dụng và Dung sai phân vùng. CAP về cơ bản là một định lý tuân theo ba nguyên tắc [1][2]:

***© 2012, IJARCSSE Tất cả các quyền Trang | 21***

***Vatika Sharma và cộng sự, Tạp chí quốc tế về nghiên cứu advenced trong khoa học máy tính và kỹ thuật phần mềm 2 (8), tháng 8- 2012, trang 20-27***

(1) Dữ liệu có sẵn trên tất cả các máy phải giống nhau về mọi mặt và cập nhật được thực hiện trên tất cả các máy thường xuyên, tức là. dữ liệu nhất quán.

(2) Dữ liệu phải có sẵn vĩnh viễn và phải có thể truy cập được mỗi lần, tức là tính khả dụng. (3) Trong quá trình hỏng hóc máy hoặc bất kỳ lỗi nào trong cơ sở dữ liệu máy sẽ hoạt động tốt mà không dừng công việc của chúng, tức là dung sai phân vùng.

**IV. VAI TRÒ CỦA KIẾN TRÚC DỮ LIỆU TRONG NOSQL**

A. **Thành phần:** Có bốn thành phần trong khối xây dựng của nó [6].

[1.] **Ngôn ngữ mô hình hóa:** Nó mô tả cấu trúc của cơ sở dữ liệu và cũng xác định lược đồ mà nó dựa trên. Dữ liệu được lưu trữ dưới dạng hàng và cột sử dụng định dạng XML. Và mỗi dữ liệu (giá trị) tương ứng với nó được gán một khóa có bản chất duy nhất. Để truy cập dữ liệu nhanh hơn, mô hình được xây dựng trong một môi trường phù hợp.

[2.] **Cấu trúc cơ sở dữ liệu**: Mỗi và mọi cơ sở dữ liệu trong khi xây dựng sử dụng cấu trúc dữ liệu riêng và lưu trữ dữ liệu bằng thiết bị lưu trữ vĩnh viễn.

[3.] **Ngôn ngữ truy vấn cơ**  sở dữ liệu: Tất cả các hoạt động được thực hiện trên cơ sở dữ liệu đang tạo, cập nhật, đọc và xóa (CURD).

[4.] **Giao dịch:** Trong bất kỳ giao dịch nào trong dữ liệu, có thể có bất kỳ loại lỗi hoặc lỗi nào; sau đó, máy sẽ không ngừng hoạt động.

**B. Các loại kho dữ liệu NoSQL**

Trên cơ sở định lý CAP, cơ sở dữ liệu NoSQL được chia thành số lượng cơ sở dữ liệu. Có bốn loại kho dữ liệu khác nhau mới trong NoSQL [4].

**1. Cơ sở dữ liệu giá trị khóa:** Bản thân tên cơ sở dữ liệu giá trị khóa nói rằng nó là sự kết hợp của hai thứ là khóa và một giá trị. Đây là một trong những hệ thống cơ sở dữ liệu cấu hình thấp (truyền thống). Cơ sở dữ liệu Key Value (KV) là mẹ của tất cả các cơ sở dữ liệu của NoSQL.

∙ Khóa là một mã định danh duy nhất cho một mục nhập dữ liệu cụ thể. Khóa không nên được lặp lại nếu một khóa được sử dụng mà nó không trùng lặp về bản chất.

∙ Giá trị là một loại dữ liệu được trỏ bởi một khóa.

Cơ sở dữ liệu giá trị chính dường như là bảng băm hoặc bảng tra cứu. Trong loại cơ sở dữ liệu này, chỉ có một cách để truy vấn đó là với sự trợ giúp của khóa (duy nhất) và tất cả các khóa có thể đặt tên trong bất kỳ đối tượng dữ liệu nào và được sắp xếp theo thứ tự bảng chữ cái [3]. Để có tính khả dụng cao hơn của kho dữ liệu, các đối tượng dữ liệu được sao chép.

Điều này được minh họa trong hình 4 dưới đây. Ví dụ, hãy lấy một ví dụ về cơ sở dữ liệu ngân hàng như trong hình 3.

|  |  |
| --- | --- |
| **CƠ SỞ DỮ LIỆU NGÂN HÀNG** | |
| Chìa khoá | Giá trị |
| 1 | Mã số hội viên:1  Ngày gia nhập: 15-7-1985  Chức danh: Thu ngân |
| 2 | Mã số hội viên:2  Ngày gia nhập: 19-March-1982  Chức danh: Quản lý |
| 3 | Mã số hội viên:3  Ngày gia nhập: 4-4-4-1988  Chức danh: Nhân viên lễ tân |

Hình 3: Cơ sở dữ liệu Key Value (KV)

Trong hình đã cho có hai cột đại diện cho khóa và một giá trị. Ở đây khóa là duy nhất và đại diện cho các giá trị hoặc thuộc tính của chúng tương ứng với nó và dữ liệu được biểu diễn dưới dạng vòng và việc phân vùng dữ liệu được thực hiện trên cơ sở bảng chữ cái của chúng (theo thứ tự sắp xếp) và dữ liệu cũng được sao chép dưới dạng vòng. Điều này như được trình bày trong phần tiếp theo.

**1.1 Đặc điểm của cơ sở dữ liệu giá trị chính**

∙ Số lượng khóa có thể có một tập hợp các thuộc tính động trong cơ sở dữ liệu giá trị khóa trong quá trình lưu trữ dữ liệu.

∙ Dữ liệu được lưu trữ trong cơ sở dữ liệu được lưu trữ theo thứ tự bảng chữ cái.

***© 2012, IJARCSSE Tất cả các quyền Trang | 22***

***Vatika Sharma và cộng sự, Tạp chí quốc tế về nghiên cứu advenced trong khoa học máy tính và kỹ thuật phần mềm 2 (8), tháng 8- 2012, trang 20-27***

∙ Tất cả các hoạt động có thể được thực hiện trên dữ liệu, tức là CRUD ( Tạo, Đọc và Cập nhật và Xóa). ∙ Tất cả các mối quan hệ với dữ liệu được lưu trữ trong mã ứng dụng (không lan truyền rõ ràng).

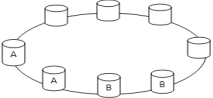
**1.2 Sử dụng cơ sở dữ liệu Key Value (KV)**

∙ Đây là một trong những mô hình dữ liệu đơn giản trong số tất cả (chúng ta sẽ thảo luận sau) vì nó chỉ sử dụng khóa và giá trị. (như mô tả trong hình 3)

∙ Nó xử lý tải dữ liệu khổng lồ.

∙ Nó mở rộng quy mô đến khối lượng lớn dữ liệu.

∙ Sao chép dữ liệu được thực hiện bằng cách sử dụng cơ sở dữ liệu dưới dạng vòng. Dữ liệu được sao chép được lưu trữ dưới dạng vòng cũng như theo thứ tự bảng chữ cái. Điều này được thể hiện trong hình 4 dưới đây:

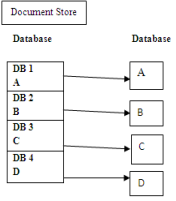


Hình 4: Phân vùng vòng và sao chép dữ liệu

**2. Cơ sở dữ liệu lưu trữ tài liệu**

Cơ sở dữ liệu Kho tài liệu là những cơ sở dữ liệu NoSQL sử dụng bản ghi làm tài liệu. Loại cơ sở dữ liệu này lưu trữ các tài liệu phi cấu trúc (văn bản) hoặc bán cấu trúc (XML) thường có tính chất phân cấp. Ở đây mỗi tài liệu bao gồm một tập hợp các khóa và giá trị gần giống như trong cơ sở dữ liệu Key Value. Mỗi cơ sở dữ liệu nằm trong tài liệu lưu trữ các điểm đến các trường của nó bằng cách sử dụng con trỏ khi nó sử dụng kỹ thuật băm. Kho tài liệu Cơ sở dữ liệu không có lược đồ và không cố định về bản chất.

Cấu trúc của Cơ sở dữ liệu lưu trữ tài liệu như được minh họa trong hình 5 được đưa ra dưới đây. Hình vẽ mô tả rằng nó bao gồm số lượng cơ sở dữ liệu trong kho lưu trữ tài liệu như cơ sở dữ liệu 1,2,3,4 và đang có id A, B, C, D nằm trong đó đang trỏ đến cơ sở dữ liệu của nó đang có một số mối quan hệ với nó. Cơ sở dữ liệu chỉ ra giá trị của nó bằng cách sử dụng một số khóa duy nhất nằm trong cơ sở dữ liệu của nó. Điều này bao gồm một mảng cơ sở dữ liệu (ở dạng vùng lưu trữ). Điều này sẽ rõ ràng hơn sau khi lấy một ví dụ được thảo luận dưới đây.



Hình 5: Cơ cấu kho tài liệu

***© 2012, IJARCSSE Tất cả các quyền Trang | 23***

***Vatika Sharma và cộng sự, Tạp chí quốc tế về nghiên cứu advenced trong khoa học máy tính và kỹ thuật phần mềm 2 (8), tháng 8- 2012, trang 20-27***

**2.1 Đặc điểm của kho tài liệu Cơ sở dữ liệu**

∙ Tài liệu được giải quyết trong cơ sở dữ liệu bằng cách sử dụng khóa (duy nhất) đại diện cho tài liệu đó.

∙ Có một số loại để tổ chức dữ liệu là bộ sưu tập, thẻ, siêu dữ liệu không hiển thị và hệ thống phân cấp thư mục.

∙ Trong đó, chúng ta có thể sử dụng tra cứu khóa-giá trị để truy xuất tài liệu.

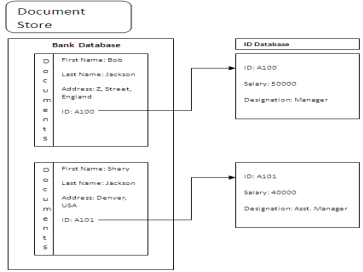
**2.2 Ví dụ về cơ sở dữ liệu Kho tài liệu**

Hình 6 minh họa một ví dụ về Cơ sở dữ liệu lưu trữ tài liệu. Trong hình này, chúng tôi đang lấy một ví dụ về cơ sở dữ liệu ngân hàng sử dụng ID làm khóa duy nhất. Và nó cũng bao gồm cơ sở dữ liệu ID chứa tất cả thông tin liên quan đến người đang có ID cụ thể đó. ID trong cơ sở dữ liệu ngân hàng trỏ đến cơ sở dữ liệu ID bằng cách sử dụng con trỏ. Cơ sở dữ liệu ngân hàng bao gồm hai tài liệu:

(1) Một người có tên: Bob; Họ: Jackson; Địa chỉ: Z Street, England; Mã số hội viên: A100.

(2) Một người có tên: sherry; Họ: Hilton; Địa chỉ: Denver, Hoa Kỳ; Mã số hội viên: A101.

Bây giờ chúng ta có thể coi ID của cả hai tài liệu là một khóa được sử dụng cho thư từ. Khóa phải là duy nhất về bản chất và không vô hiệu. Chúng tôi không thể tạo khóa trùng lặp. Chúng tôi đã chọn ID làm khóa vì tên, địa chỉ không thể là duy nhất và ID của một người luôn là duy nhất. Vì vậy, đó là lý do tại sao chúng tôi đã chọn ID làm khóa. Bây giờ chúng ta đang có một cơ sở dữ liệu khác là cơ sở dữ liệu ID. Cơ sở dữ liệu ID bao gồm tiền lương và chỉ định của một người. Cơ sở dữ liệu ID bao gồm các cơ sở dữ liệu của ID A100, A101. Các tài liệu hiện có trong cơ sở dữ liệu ngân hàng bao gồm ID và trỏ đến dữ liệu ID cụ thể của nó. Đây là cách dữ liệu được tổ chức trong cơ sở dữ liệu lưu trữ tài liệu. Dữ liệu được tổ chức dưới dạng bộ sưu tập, thẻ, siêu dữ liệu, v.v. Trong đó, chúng ta có thể truy xuất dữ liệu bằng cách sử dụng cặp khóa và giá trị. Điều này được minh họa rõ hơn trong ví dụ mà chúng ta có khóa A100 và các thuộc tính tương ứng của nó bao gồm mức lương của nó với chỉ định.



Hình 6: Cơ sở dữ liệu kho lưu trữ tài liệu

**3. Cơ sở dữ liệu cột**

Cơ sở dữ liệu cột còn được gọi là cơ sở dữ liệu họ cột vì chúng là cơ sở dữ liệu hướng cột.

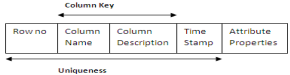
Có hai loại cơ sở dữ liệu hướng cột có chi tiết như được đưa ra dưới đây:

***© 2012, IJARCSSE Tất cả các quyền Trang | 24***

***Vatika Sharma và cộng sự, Tạp chí quốc tế về nghiên cứu advenced trong khoa học máy tính và kỹ thuật phần mềm 2 (8), tháng 8- 2012, trang 20-27***

**(1) Lưu trữ dữ liệu cột rộng:**

Nó là một loại cơ sở dữ liệu NoSQL. Kho dữ liệu Wide Column là những cơ sở dữ liệu được sử dụng để xử lý web, truyền dữ liệu và tài liệu. Cấu trúc của kho dữ liệu Cột rộng như được mô tả trong hình 7 bên dưới:



Hình 7: Cơ cấu kho dữ liệu cột rộng

Ý nghĩa của từng trường được cung cấp trong cấu trúc của cơ sở dữ liệu lưu trữ dữ liệu cột rộng như được mô tả bên dưới trong Hình 7.

|  |  |
| --- | --- |
| **THUỘC TÍNH** | **Ý NGHĨA** |
| **Hàng không** | Nó là một chìa khóa độc đáo trong tự nhiên. Nó có thể là một chuỗi hoặc một số. |
| **Tên Cột** | Dữ liệu được lưu trữ trên cơ sở họ cột. |
| **Mô tả cột** | Nó mô tả mục dữ liệu được lưu trữ. |
| **Dấu thời gian** | Nó cho biết thời gian hoàn chỉnh của trường hợp cụ thể. |
| **Giá trị dữ liệu** | Giá trị hoặc thuộc tính liên quan đến khóa tương ứng đó. |

Hình 8: Ý nghĩa của các trường trong cấu trúc cơ sở dữ liệu cột rộng

**(2)** Cơ**sở dữ liệu định hướng C olumn:**

Để hiểu cơ sở dữ liệu hướng cột, hãy lấy một ví dụ về cơ sở dữ liệu ngân hàng được đưa ra trong hình 9 có các trường thuộc tính là EmpID, Mức lương và chỉ định và các giá trị tương ứng với nó như được mô tả trong cơ sở dữ liệu.

|  |  |  |
| --- | --- | --- |
| **EmpID** | **Lương** | **Chỉ định** |
| 100 | 10,000 | Clerk |
| 200 | 20,000 | Trợ lý Giám đốc |
| 300 | 30,000 | Giám đốc |
| 400 | 40,000 | Đầu khu vực |

Hình 9: Ví dụ về cơ sở dữ liệu ngân hàng

Đại diện của cơ sở dữ liệu định hướng Hàng và cơ sở dữ liệu hướng cột:

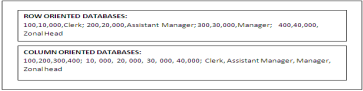
∙ Cơ sở dữ liệu hướng hàng là những cơ sở dữ liệu trong đó tất cả các hàng được đặt cùng nhau từng cái một.

∙ Cơ sở dữ liệu định hướng cột những cơ sở dữ liệu trong đó tất cả các giá trị chứa các cột được đặt cùng nhau.

Với sự trợ giúp của cơ sở dữ liệu được đưa ra ở trên, chúng tôi sẽ đại diện cho cơ sở dữ liệu hướng hàng và cột, như thể hiện trong hình 10 bên dưới:

***© 2012, IJARCSSE Tất cả các quyền Trang | 25***

***Vatika Sharma và cộng sự, Tạp chí quốc tế về nghiên cứu advenced trong khoa học máy tính và kỹ thuật phần mềm 2 (8), tháng 8- 2012, trang 20-27***

******

Hình 10: Biểu diễn cơ sở dữ liệu định hướng hàng và cột

**3.1 Đặc điểm của cơ sở dữ liệu cột**

(1) Cơ sở dữ liệu cột nhanh hơn cơ sở dữ liệu dựa trên hàng trong khi truy vấn.

(2) Trong cơ sở dữ liệu cột, việc gán đơn vị lưu trữ được thực hiện cho từng cột.

(3) Trong DBMS cột chỉ đọc các cột bắt buộc, vì vậy việc đọc nhanh hơn trong trường hợp này.

**4. Cơ sở dữ liệu đồ thị**

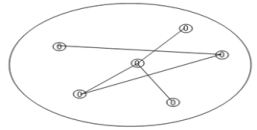
Cơ sở dữ liệu đồ thị dựa trên lý thuyết đồ thị. Nói chung, chúng ta thấy rằng đồ thị thường bao gồm \node, properties và edge. Cơ sở dữ liệu Đồ thị NoSQL bao gồm:

(1) Các nút đại diện cho các thực thể

(2) Thuộc tính đại diện cho các thuộc tính

(3) Các cạnh đại diện cho các mối quan hệ [6].

Cấu trúc của cơ sở dữ liệu đồ thị như hình dưới đây:



Hình 11: Cấu trúc của cơ sở dữ liệu đồ thị

**4.1 Ví dụ về cơ sở dữ liệu Đồ thị**

Ví dụ về Cấu trúc của cơ sở dữ liệu Đồ thị được thể hiện trong Hình 12. Nó giải thích cho chúng ta cách biểu diễn các nút, mối quan hệ và thuộc tính. Ở đây có hai nút (được biểu thị bằng vòng tròn) A và B đại diện cho NGÂN HÀNG và TÀI KHOẢN. Cả hai đều đang có một mối quan hệ được thể hiện bằng một đường dây mà ngân hàng có số lượng tài khoản. Tiếp theo là các thuộc tính/thuộc tính/giá trị đại diện cho các loại tài khoản trong ví dụ được đưa ra trong hình dưới đây. Tài sản là Tiết kiệm, Hiện tại và

Rd.

**4.2 Đặc điểm của cơ sở dữ liệu Đồ thị**

∙ Các đường truyền đồ thị được thực hiện với tốc độ không đổi độc lập với tổng kích thước của biểu đồ. Không có hoạt động tập hợp nào liên quan làm giảm hiệu suất như được thấy với các hoạt động tham gia trong RDBMS.

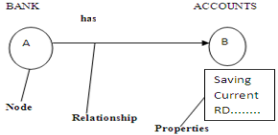
∙ Cơ sở dữ liệu đồ thị đang có hiệu suất cao trong bối cảnh truyền sâu của chúng.

∙ Chúng được sử dụng để tính toán đường đi ngắn nhất.

∙ Chúng có thể mở rộng. Nhưng độ phức tạp của nó tăng lên.

***© 2012, IJARCSSE Tất cả các quyền Trang | 26***

***Vatika Sharma và cộng sự, Tạp chí quốc tế về nghiên cứu advenced trong khoa học máy tính và kỹ thuật phần mềm 2 (8), tháng 8- 2012, trang 20-27***



Hình 12: Ví dụ về cấu trúc của cơ sở dữ liệu Đồ thị

**V. ĐẶC ĐIỂM CỦA NoSQL**

∙ NoSQL không sử dụng mô hình dữ liệu quan hệ do đó không sử dụng ngôn ngữ SQL.

∙ NoSQL lưu trữ khối lượng lớn dữ liệu.

∙ Trong môi trường phân tán (lan truyền dữ liệu đến các máy khác nhau), chúng tôi sử dụng NoSQL mà không có bất kỳ sự mâu thuẫn nào.

∙ Nếu có bất kỳ lỗi hoặc lỗi nào tồn tại trong bất kỳ máy nào, thì trong đó sẽ không có bất kỳ công việc nào bị ngừng lại.

∙ NoSQL là cơ sở dữ liệu mã nguồn mở, tức là mã nguồn của nó có sẵn cho tất cả mọi người và có thể sử dụng miễn phí mà không có bất kỳ chi phí nào.

∙ NoSQL cho phép dữ liệu lưu trữ trong bất kỳ bản ghi nào mà nó không có bất kỳ lược đồ cố định nào.

∙ NoSQL không sử dụng khái niệm về tính chất ACID.

∙ NoSQL có thể mở rộng theo chiều ngang dẫn đến hiệu suất cao theo cách tuyến tính.

∙ Nó có cấu trúc linh hoạt hơn.

**VI. KẾT LUẬN VÀ CÔNG VIỆC TƯƠNG LAI**

Mục đích chính của bài báo này là cung cấp một cái nhìn tổng quan về cơ sở dữ liệu NoSQL, về cách nó đã từ chối sự thống trị của SQL, với nền tảng và đặc điểm của nó. Nó cũng mô tả các nguyên tắc cơ bản của nó tạo thành cơ sở của cơ sở dữ liệu NoSQL như định lý ACID, BASE và CAP. Thuộc tính ACID không được sử dụng trong cơ sở dữ liệu cơ sở dữ liệu NoSQL vì tính nhất quán của dữ liệu, vì vậy chúng ta sẽ biết SQL tụt hậu như thế nào so với tính nhất quán của dữ liệu. Sau đó, trên cơ sở định lý CAP, chúng tôi đã mô tả các loại cơ sở dữ liệu NoSQL khác nhau là cơ sở dữ liệu Key-Value, Cơ sở dữ liệu lưu trữ tài liệu, cơ sở dữ liệu dựa trên cột và cơ sở dữ liệu Đồ thị với sự trợ giúp của một ví dụ. Ngoài tất cả những điều này, chúng tôi cũng đã mô tả các đặc điểm, độ phức tạp và hiệu suất của chúng. Nghiên cứu sâu hơn đang diễn ra trong các công nghệ mới đang phát sinh cho hoặc sau NoSQL đó là sự bền bỉ của đa giác, v.v.

**THAM KHẢO**

[1.] SilvanWeber, “ NoSQLDatabases ” http://www.christof-strauch.de /nosqldbs.pdf

[2.] Martin Fowler and Pramod Sadalage Rendered, “NoSQLdbs- “, February8,2012,11:26, http://martinfowler.com/articles/nosql-intro.pdf

[3.] An Oracle White Paper, “Oracle NoSQL Database”, September2011, http://www.oracle.com/technetwork/database/nosqldb/learnmore/nosql-database-498041.pdf

[4.] Luis Ferreira Universidade do Minho, “Bridging the gap between SQL and NoSQL”, httpsikhote.files.wordpress.com201105artigo-mi-star1.pdf

[5.] Andrew J. Brust, Blue Badge Insights, Inc., “NoSQL and the Windows Azure platform”, April 25, 2011 [6.] DAMA - Philadelphia / Delaware Valley, the “Role of Data Architecture in NOSQL”, Wednesday January 11th, 2012, http://www.damaphila.org/HaugheyNOSQL.pdf

[[

[[

***© 2012, IJARCSSE All Rights Reserved Page | 27*** Xem số liệu thống kê xuất bản