**Graph store**

Graph store rất quan trọng trong các ứng dụng mà cần phải phân tích mối liên hệ giữa đối tượng hoặc truy cập vào tất cả các nút trong một đồ thị trong một cách đặc biệt (ví dụ như đồ thị traversal). Graph store được tối ưu để lưu trữ các nút và các cạnh trong đồ thị cho hiệu quả và cho phép truy vấn đến những đồ thị đó. Graph databases là hữu ích cho bất kỳ vấn đề kinh doanh mà có các mối quan hệ phức tạp giữa các đối tượng như: mạng xã hội, các phương pháp dựa trên các quy tắc, tạo mashups, và hệ thống đồ thị đó có thể nhanh chóng phân tích cấu trúc mạng phức tạp và tìm kiếm các mẫu trong các cấu trúc này.

 Đến cuối phần này, bạn sẽ có thể để xác định các tính năng chính của một đồ thị lưu trữ và hiểu làm thế nào mà các graph store được sử dụng để giải quyết các vấn đề kinh doanh cụ thể. Bạn sẽ trở nên quen thuộc với các thuật ngữ đồ thị như các nút, các cạnh, và thuộc tính của nó,và bạn sẽ biết về các tiêu chuẩn W3C công bố dữ liệu cho đồ thị. Bạn cũng sẽ thấy làm thế nào các graph store được thực hiện có hiệu quả tại các công ty để thực hiện liên kết phân tích, sử dụng với các phương pháp và suy luận, và tích hợp dữ liệu liên quan.

1. Tổng quan

Graph store là hệ thống các nút liên quan liên kết lại với nhau tạo thành 1 đồ thị. Mỗi đồ thị gồm có 3 trường dữ liệu: nút, cạnh và thuộc tính. Các nút thường được đại diện cho các đối tượng trong thế giới thực như con người, tổ chức, sđt, … thậm chí là các tê bào trong cơ thể sống. Cạnh là thứ kết nối giữa các đối tượng lại với nhau và được biểu diễn bằng các cạnh nối giữa các nút. Graph stores liên kết các nút lại với nhau bằng cách tạo ra các định danh cho các nút và sử dụng các định danh đó để nối chúng lại với nhau thành 1 mạng lưới.

Graph store thường được sử dụng khi bạn cần lưu các dữ liệu phức tạp mà liên quan với nhau, mà mỗi liên hệ của nó đều có các thuộc tính riêng. Graph store cho phép bạn thực hiện các câu query đơn giản để show ra các nút lân cận. Tốc độ tính toán nhanh là 1 đặc tính tự nhiên của mỗi nút, nhờ vậy mà nó có khả năng giữ lại các dữ liệu của đồ thị trong RAM, là do khi 1 đồ thị được nạp vào bộ nhớ, thì nó có thể lấy lại dữ liệu mà không yêu cầu input hay output.

Graph store thì khó có thể mở rộng ra trên nhiều máy chủ. Các nút trong đồ thị đều được liên kết chặt chẽ, do đó khi mở rộng ra trên nhiều máy chủ thì sẽ phá huỷ tính chặt chẽ của đồ thị mặc dù dữ liệu có thể được nhân rộng trên nhiều máy chủ để performance được nâng cao nhưng việc ghi các nút trên nhiều server thì thực hiện phức tạp.

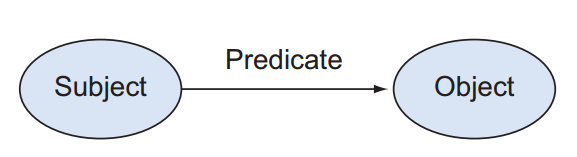
1. Liên kết dữ liệu bên ngoài với chuẩn RDF (Resource Description Format)

Bạn có thể tạo thêm các phương thức riêng để xác định xem các nút nào cùng tham chiếu đến 1 điểm trong đồ thị. Một số graph store sẽ cấp ID cho mỗi nút để load những nút đó vô RAM.

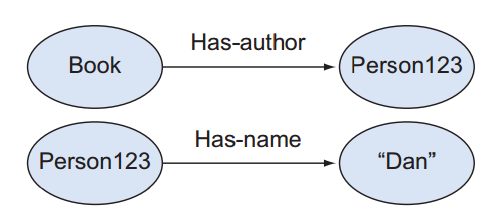
Sử dụng các URL để định danh cho các nút và lưu các thông tin về các liên kết giữa các page cũng như các nút lại với nhau sẽ được W3C tổng quát hoá lại vô 1 chuẩn gọi là RDF. RDF được tạo ra để bạn có thể tham gia vào các tập dữ liệu ngoài của các tổ chức khác nhau.

Ta có thể load cơ sở dữ liệu từ ngoài vào graph store và có thể thực hiện các câu truy vấn trên database mới – là database sau khi các tập dữ liệu bên ngoài sáp nhập vô. Đặc điểm nhận dạng là khi có nhiều nút cùng tham chiếu đến 1 đối tượng được thể hiện qua đồ thị có hướng bởi RDF, hướng từ nút source đến nút destination.

Các thuật ngữ như source, link, destination có thể thay đổi tuỳ thuộc vào tình huống sử dụng nhưng thường thì các subject, predicate, object thì được sử dụng.



Những thuật ngữ này xuất phát từ hệ thống logic và ngôn ngữ. Những thuật ngữ này để mô tả các nút được xác định trong chuẩn RDF của W3C. Trong RDF, mối quan hệ node-arc-node được gọi là triple và được kết hợp với thực tế. Hình phía dưới có 2 xác định. Thứ nhất là Book, has-author, Person123. Thứ hai là Person123, has-name, “Dan”



Khi lưu trong graph store, có 2 câu lệnh độc lập có thể được lưu trong các hệ thống khác nhau. Person123 thì đều có trong cả 2 xác định trên, do đó mà ứng dụng của mình có thể nhận ra rằng Author của Book có tên là “Dan”.



Trên thực tế thì hai nút ở 2 nhóm khác nhau có thể tham chiếu đến cùng 1 đối tượng vật lý. Ví dụ, nút Person123 ở 2 đồ thị trên là như nhau. Do đó, ta có thể ghép chúng lại thành 1 đồ thị như phía dưới. Quá trình này rất hữu ích trong các mô hình phức tạp cũng như trong các lĩnh vực đòi hỏi sự suy luận.

Có 1 phương pháp có thể xác định HTML bất cứ ở đâu trên thế giới bằng cách thống nhất cấu trúc tài nguyên sử dụng, do đó ta có thể sử dụng cấu trúc này ở bất kì khi nào có thể. Khác với URL, URI không trỏ đến bất kì trang web nào. Do đó, bạn phải tìm ra cách làm cho chúng nhất quán trên toàn bộ trang web và phù hợp khi so sánh 2 nút với nhau.

Trong thực tế, mỗi triple thường liên kết thông tin với các triple khác. Các thuộc tính thường được sử dụng để liên kết các triple lại với nhau được gọi là siêu dữ liệu liên kết vì chúng mô tả thông tin về sự liên kết đó. Việc lưu trữ siêu dữ liệu liên kết thì không mất quá nhiều không gian đĩa cứng nhưng nó lại giúp ta dễ dàng kiểm soát và quản lý dữ liệu hơn.

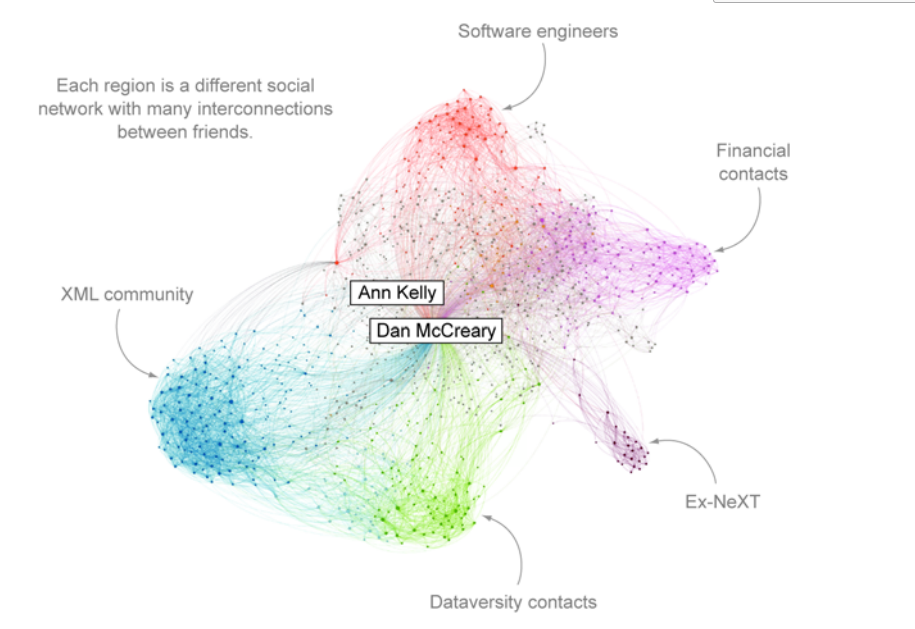
1. Use cases for Graph stores

Các tình huống mà graph store có thể được sử dụng hiệu quả để giải quyết 1 vấn đề cụ thể:

* Link analysis được sử dụng khi ta muốn thực hiện tìm kiếm các mô hình và các mối quan hệ trong các tình huống sử dụng như mạng xã hội, điện thoại, email.
* Rules and inference được sử dụng khi muốn truy vấn đến các cấu trúc phức tạp
* Integrating linked data được sử dụng với số lượng lớn các siêu dữ liệu mở để thực hiện các công việc trong thời gian thực và xây dựng các mashup mà không cần phải lưu dữ liệu.

1. LINK ANALYSIS

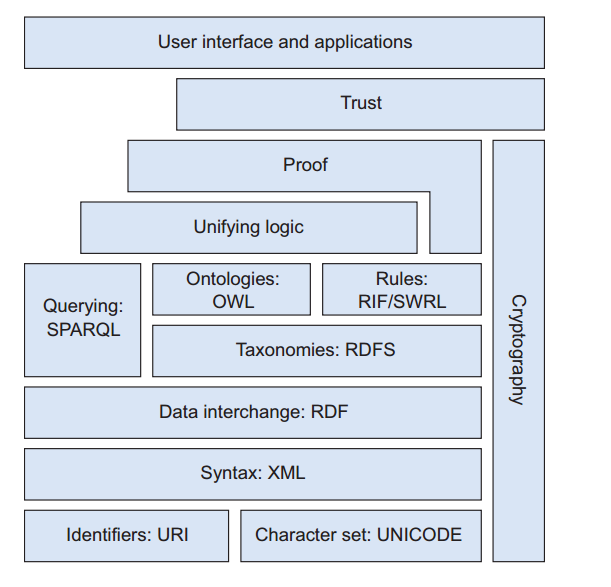
* Cách tốt nhất để giải quyết 1 vấn đề kinh doanh đó là thể hiện vấn đề này lên graph store. Hình dưới là ví dụ về đồ thị về mạng xã hội. VD: Khi mình thêm 1 bạn mới vào list frend của mình, mình cũng muốn biết xem người bạn mình dự định sẽ thêm vào có ai là bạn chung với mình không? Và để có được thông tin này, trước tiên mình sẽ cần cái list friend của mình . Và mỗi người trong list friend của mình thì đều có 1 list friend riêng của họ. Và nếu như trong list friend của họ có tên của bạn thì đó là những người bạn chung.



* Graph store có thể thực thi nhiều hoạt động trong quá trình phân tích nhanh hơn bằng cách sử dụng kỹ thuật loại bỏ các nút không cần từ bộ nhớ. Mặc dù vậy, nhưng cũng cần phải đủ RAM để có thể lưu trữ tất cả liên kết trong quá trình phân tích.
* Graph store thường được sử dụng cho những thứ như mạng xã hội vì nó thích hợp để xác định mô hình khác biệt của các liên kết giữa các nút.
* Graph store cũng rất hữu ích trong việc liên kết hợp dữ liệu và tìm kiếm các mô hình trong các tài liệu lớn. Khai thác các thực thể là quá trình xác định các thực thể quan trọng nhất trong 1 tài liệu. Các đối tượng thường là 1 danh từ trong tài liệu như con người, ngày tháng, nơi chốn, sản phẩm. Khi các thực thể chính đã được xác định, chúng được sử dụng để thực hiện các chức năng tìm kiếm nâng cao.
* Quá trình khai thác thực thể (là 1 loại của “Xử lý ngôn ngữ tự nhiên” (NLP)) có thể kết hợp với các công cụ khác để trích xuất dữ kiện đơn giản. Ví dụ cho câu sau “John Adams was born on October 19, 1735” có thể được chia thành các xác định sau:
  + Hồ sơ về 1 người là “John Adams” và là 1 chủ đề
  + Một mối liên hệ liên kết subject với object.
  + Đối tượng về ngày có gái trị là “October 19, 1735”
* Mặc dù xác định đơn giản có thể dễ dàng tìm thấy bằng cách xử lý NLP, nhưng quá trình hiểu được từng câu hoàn toàn có thể trở nên phức tạp và phụ thuộc vào ngữ cảnh sử dụng, tình huống sử dụng. Các xác định này tốt nhất nên được thể hiện trong cấu trúc đồ thị

1. GRAPHS, RULES, AND INFERENCE

* Việc quy định các thuật ngữ thì phụ thuộc vào bối cảnh, tình huống, địa điểm sử dụng. Ở đây, các thuật ngữ này được sử dụng để xác định các quy tắc trừu tượng liên quan đến sự hiểu biết các đối tượng trong hệ thống và làm thế nào các thuộc tính của đối tượng cho phép ta hiểu được đối tượng sâu hơn, và sử dụng tốt các tập dữ liệu lớn.
* RDF được thiết kế để trở thành 1 chuẩn trong nhiều loại vấn đề trong cấu trúc đồ thị. Mục đích chính của việc sử dụng RDF là để lưu trữ các logic và luật. Và khi đã thiết lập các luật này, ta có thể suy luận hoặc sử dụng luật để khám phá về 1 điều khác trong hệ thống. Việc sử dụng RDF và suy luận không giới hạn đến mạng xã hội và đánh giá sản phẩm. RDF là 1 cấu trúc với mục đích chung mà có thể được sử dụng để lưu trữ nhiều hình thức logic
* Trong phần phân tích về các liên kết thì ta đã hiểu cách văn bản được mã hoá với các thực thể như con người, ngày tháng để giúp ta tìm thấy các hoạt động liên quan. Bây giờ, ta có thể làm những thứ to tác hơn để có được các thông tin bổ sung từ các hoạt động để giải quyết 1 vấn đề kinh doanh.
* W3C là 1 framework của chuẩn sử dụng RDF để giải quyết các vấn đề kinh doanh. Framework này thường được gọi là Semantic Web Stack. Hình phía dưới là 1 loại Semantic Web Stack với các chuẩn cơ bản như URI, XML, RDF ở dưới cùng của ngăn xếp. Ở giữa thì gồm các chuẩn truy vấn (SPARQL), chuẩn quy tắc (RIF/ SWRL). Ở trên cùng của ngăn xếp là giao diện người dùng và ứng dụng trên lớp trừu tượng logic, proof, trust. Các lớp trừu tượng này vẫn chưa được chuẩn hoá. Đây cũng là nơi tập trung nhiều nghiên cứu và phát triển Semantic Web. Còn những phần đã được chuẩn hoá thì có thể an toàn trong việc trao đổi dữ liệu trên internet.



1. USING GRAPHS TO PROCESS PUBLIC DATASETS

* Graph store thì rất hữu ích trong việc phân tích các dữ liệu khi các dữ liệu đó không hoàn toàn do ta tạo ra. Điều gì xảy ra nếu ta cần phân tích 3 dataset khác nhau được tạo ra bởi ba tổ chức khác nhau? Thậm chí, các tổ chức này còn có không biết đối phương có tồn tại hay không? Vậy làm thế nào ta có thể tự động sáp nhập các dataset của họ lại với nhau để ta có thể lấy được thông tin ta cần? Làm thế nào để tạo mashup và recombinations của dữ liệu này một cách hiệu quả? Một câu trả lời duy nhất là sử dụng một bộ công cụ được gọi là Linked Open Data (LOD). Đây là 1 kỹ thuật cho phép sáp nhập các dataset khác nhau để tạo ra các ứng dụng mới và dữ liệu mới.
* Chiến lược LOD thì rất quan trọng cho bất cứ ai làm nghiên cứu hoặc phân tích các dataset có sẵn. Các nghiên cứu này có các chủ đề như khách hàng mục tiêu, phân tích xu hướng, phân tích tâm lý hoặc tạo ra các dịch vụ thông tin mới. Hợp dữ liệu vào các loại hình mới nhằm cung cấp các cơ hội cho các doanh nghiệp mới.
* LOD tạo ra các tập dữ liệu mới bằng cách kết hợp các thông tin từ nhiều dataset có sẵn cho phù hợp với cấu trúc LOD như RDF, URI. Hình dáng của các trang web LOD phổ biến được gọi là sơ đồ cloud LOD được thể hiện như hình phía dưới. Tại trung tâm của sơ đồ cloud LOD, ta sẽ thấy các trang web có chứa số lượng lớn các dataset có cùng mục đích chung.

