ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 10, October 2018

Short Survey on Graph Database

Asmita Shrivastav

Student, Department of Information Technology, Vishwakarma Government Engineering College, Gujarat Technological University, Gujarat, India

ABSTRACT: From recent decades graph database is receiving immense popularity and used in many applications instead of typical tabular method of storing data. As database size keeps on increasing day by day, it gets difficult to manage huge amount of data and perform complex data analysis and manipulation using traditional methods. This survey paper covers implementations carried out in today's world along with studying different types of available applications.

KEYWORDS: Graph Database, Relational database, Graph Models, Tabular datasets.

I. INTRODUCTION

In a visual way, graph is easiest in data structures to represent and store data. Graph databases store data in form of nodes (vertices), edges and relationship. Data entities are stored in nodes and relationships between these entities are defined on edges. All the nodes are interconnected to each other. Each edge has start node, end node and direction to show where ownership between entities is going. Each node has its unique identifier, outgoing edges and incoming edges and a set of properties in key-value pairs. Relationships can also have properties, this can be used when metadata for graph algorithms is required, also applicable to constrain queries during run time. Because of such arrangements of data, traversal among million nodes and edges within fraction of time becomes efficient even though data keeps on booming. Graph databases are useful when associativity among data entities is crucial.
In the example shown below in figure 1, George is teacher who teaches student named Mike. Mike has a friend named Ryan. This data is generated by studying graph in figure 1. In the figure, teaches and friends are relationship shown between nodes.
All the nodes has properties such as place and name.

Example:

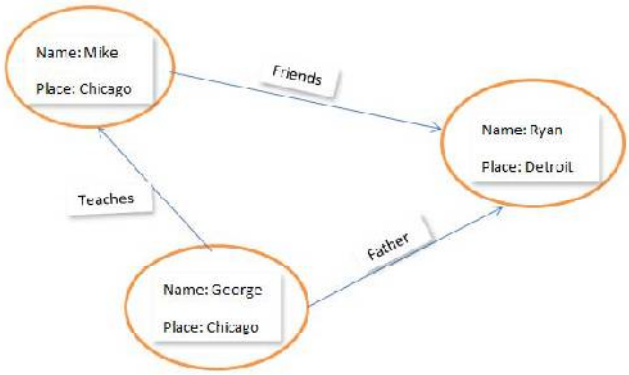

Figure 1

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 10, October 2018

II. STRENGTH OF GRAPH DATABASES

No Indexing and its Performance

Graph Databases do not rely on index for any of queries executed, therefore growing data is not a concern for graph databases, and they can easily adapt increased data. Performance of graph database while dealing with connected data is more as compared to relational database. As there is no indexing, join intensive query performance remains constant along with the increment in dataset whereas in relational database performance while dealing with increased data degrades. The reason behind constant performance in graph database is that queries are limited to specific part of graph, thus time required for execution of each query is exactly proportional to size of fixed portion of graph traversed to get output of query instead of size of whole graph.

Flexibility

Business requirements or consumer requirements may keep on changing while designing products, at this kind of time graph databases work efficiently. Graphs are flexible in many manners such as they can new nodes, new edges and any new kind of various relationships. Moreover, existing structure of graphs can be modified or new sub graphs can be added without affecting any queries and existing applications and functionalities. This flexible nature of graph database is positive implication for developer productivity as well as project risk. Due to this, data architects do not have to invest time in modelling domain whenever requirements are changed and maintain overhead. Thus, graph databases allow structure and schema to update with growing understanding of problem space.

Agility

Modern graph databases allow developers to execute smooth development and grateful system maintenance. Graph database empowers to evolve application in a controlled manner because of its schema free nature along with testable nature of graph database's application programming language and graph query language. Graph database development aligns well with today's agile and test-driven software development practices, allowing graph database–backed applications to evolve in step with changing business environments.

III. USE CASES

PageRank Algorithm

The concept of graph database is used in this algorithm which is used by Google to rank websites in Google's search results. The number and quality of pages linked to page is counted and on that count importance of page is decided. Pages are linked in form of graph so it counts number of outgoing nodes connected to the node of a pages and assign that count as weightage of page and also consider properties of relationship while assigning weight to each node. PageRank is also usedby Twitter to give recommendations of other accounts to users that they may like to follow. This algorithm can also be used in predicting traffic flow in public spaces and streets.

Fraud Detection

Graph databases are uniquely positioned to spot the connections between large data sets and identify patterns, a useful trait when it comes to spotting complex, modern fraud techniques. It already counts a number of major banks using its graph services to aid fraud detection.

Master Data Management

Graph databases allow companies to bring together customer, product, supplier and logistics information to give a holistic view of master data. This allows enterprises to gain better insights into its data ownership, supply chain and also a real-time view of customers to deliver flexible marketing offers, pricing and support. One of Neo4j's earliest adopters was Cisco, which chose Neo4j when it decided to rebuild its master data management system.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 10, October 2018

Network management

Enterprise customers can use graph databases to map entire communications or IT networks and start to run complex scenario testing to prepare for outages better. Graphs are useful in analyzing networks and data centers and in saving time from conventional process of authentication because graphs have high interconnection of plans, customers and groups. Therefore many telecommunication companies are giving importance of graph database to model their network. Most significantly, failure cases are also covered and plans to recover the loss are just a node away which in turn saves ample amount of time in case of any unexpected failure or hazard.

IV. GRAPH DATABASES VS. RELATIONAL DATABASES

When relationship among data blocks are significant then graph database proves to be best fit to explore and store data. By contrast relational database are appropriate for All-like queries. For better understanding, consider the following example where data about companies, people who work for them and how long they have been working there are to store.

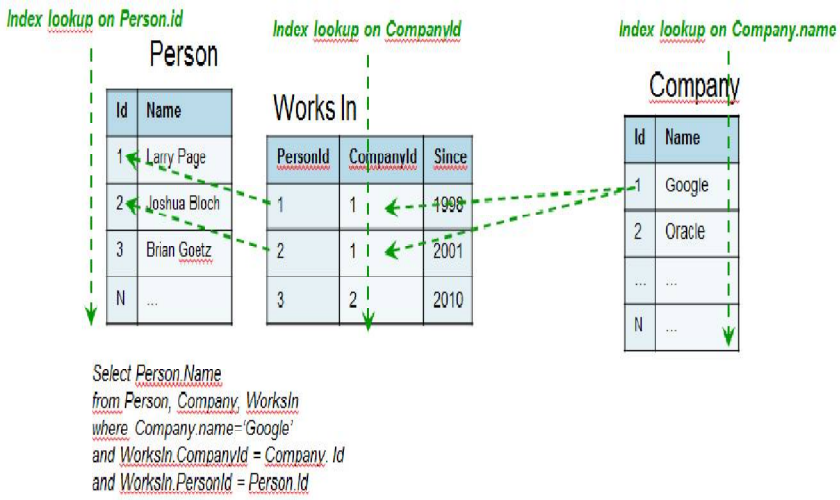

Figure 2

When query shown in figure is executed, it could be required probably 3 index lookups corresponding to foreign keys in model.Whereas only 1 index lookup is required for traversing relationships by using references of pointers used in directed graph.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 10, October 2018

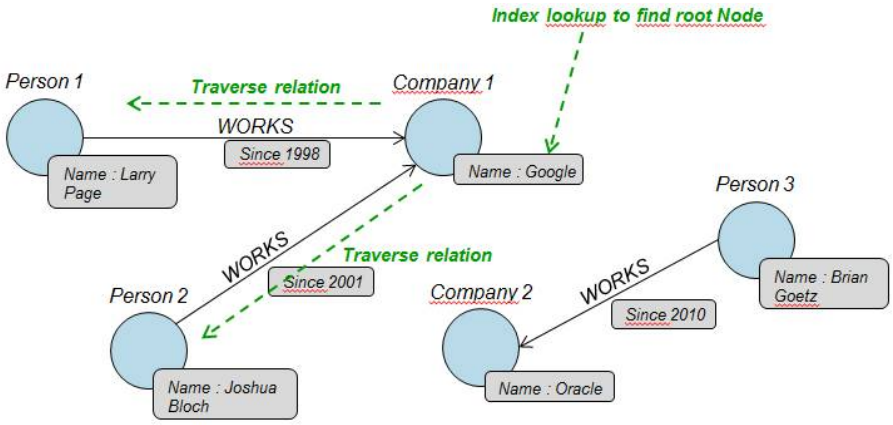

Figure 3

Comparing the performance of relational databases on graph analytics. Here a leading graph database system (Neo4j) and three relational databases: a row-oriented database (MySQL), a column-oriented database, and a main-memory database are compared. Two queries, PageRank and Shortest Paths, on each of these systems.

Considering two datasets from the Stanford large network dataset collection:

- A Facebook dataset having 4K nodes and 88K edges, and
- A Twitter dataset having 81K nodes and 1.8M edges.

Figures 4(a) and 4(b) show the result (below).

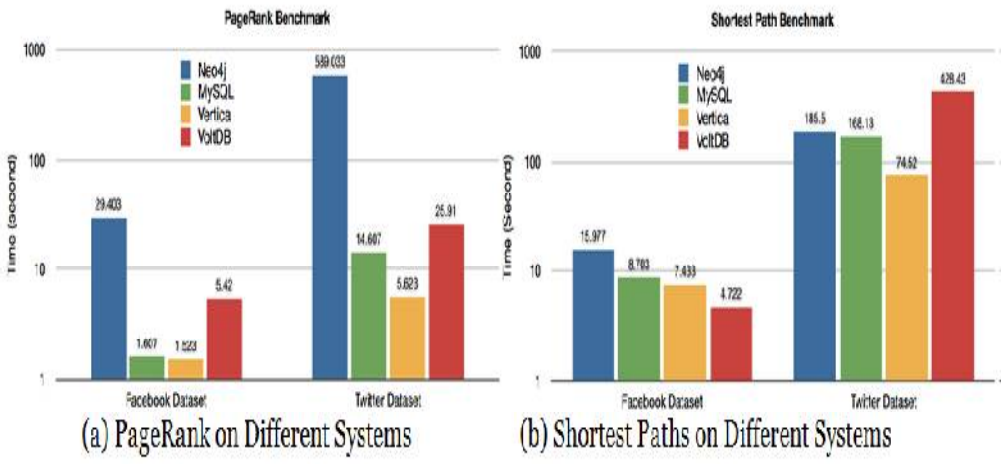

Figure 4

It can be seen that relational databases outperform graph database (Neo4j) on PageRank by up to two orders of magnitude. This is because PageRank involves full scanning and joining of the nodes and edges table, at which relational databases are good at doing. Finding Shortest Paths involves starting from a source node and successively

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 10, October 2018

exploring its outgoing edges, a very different access pattern from PageRank. In most cases, relational databases match or outperform graph cases.

V. CONCLUSION

This paper presents the brief overview about the graph database using few examples to elaborate its properties, its use cases and its comparison with relational database.

REFERENCES

1. Alleograph, https://franz.com/agraph/allegrograph/
2. Cloudgraph, http://www.cloudgraph.com/
3. Dex, http://www.sparsity-technologies.com/dex.
4. Filament, http://filament.sourceforge.net/
5. G-store, http://g-store.sourceforge.net/
6. Horton,http://research.microsoft.com/en-us/projects/ldg/
7. Hypergraphdb, http://www.hypergraphdb.org/.
8. Neo4j, http://www.neo4j.org/.
9. Orientdb. http://www.orientdb.org/.
10. Sones,http://dekorte.com/projects/opensource/vertexdb/
11. Trinity,http://research.microsoft.com/en-us/projects/trinity/.
12. Vertexdb,http://dekorte.com/projects/opensource/vertexdb/