Application Plan of Graph Databases in the Big Data Environment

Sungbum Park* · Sangwon Lee** · Hyunsup Ahn*** · In-Hwan Jung****

*Department of Management Planning, National Information Society Agency
**D. of Information & Electronic Commerce (I. of Convergence & Creativity), Wonkwang University
***Department of Wirtschaftsinformatik, Technische Universität Braunschweig
****Department of Computer Engineering, Hansung University

E-mail: parksb@nia.or.kr, sangwonlee@wku.ac.kr, hs.ahn@tu-bs.de, ihjung@hansung.ac.kr

ABSTRACT

Even though Relational Databases have been widely used in many enterprises, the relations among entities are not managed effectively and efficiently. In order to analyze Big Data, it is absolutely needed to express various relations among entities in a graphical form. In this paper, we define Graph Databases and its structure. And then, we check out their characteristics such as transaction, consistency, availability, retrieval function, and expandability. Also, we appropriate or inappropriate subjects for application of Graph Databases.

키워드
Big Data, Graph Databases, Application Plan, NoSQL

I. Introduction

Relational Databases have been widely used in many enterprises. But, the relations among entities are not managed effectively and efficiently. Many enterprises absolutely need to express various relations among entities in a graphical form when analyzing Big Data. In this paper, we define graph databases and its structure. And then, we check out their characteristics such as transaction, consistency, availability, retrieval function, and expandability. The queries are explained by some examples of query languages such Gremlin and Cyper. Also, we appropriate or inappropriate subjects for application of column-family stores.

II. Graph Databases for Big Data

By definition, a graph database is a kind of storage system that provides index-free adjacency. Graph Databases are based on graph theory (Figure 1). Graph Databases employ nodes, properties, and edges. Nodes are very similar in nature to the objects.
Object-oriented programmers will be familiar with objects. So, a graph database is a database that uses graph structures with nodes, edges, and properties to represent and store data. Every element contains a direct pointer to its adjacent element. In addition, no index lookups are necessary. Rather than Relational Databases, Graph Databases are faster for associative data sets and map more directly to the structure of object-oriented applications. As they do not typically require expensive join operations, they can scale more naturally to large data sets. In the meantime, Relational Databases are more suitable to manage ad hoc and changing data with evolving schema. They depend less on a rigid schema. Like Column-Family Stores and Document Databases, Graph Databases guarantee consistency, transaction, availability, various queries, and extensibility of column-family data. Examples of Figure 2 and 3 are relations with attributes and node partition each.

The appropriate places of using Graph Databases are linked data, routing, dispatch, location-based service and recommendation service. But, Graph Databases do not support an optimal solution for entity update, especially in modifying entities or attributes.

III. Application Plan of Graph Databases for Big Data

Consequently, Graph Databases are a powerful tool for graph-like queries, for example computing the shortest path between two nodes in the graph. Other graph-like queries can be performed over a graph database in a natural way such as graph’s diameter computations or community detection.

References

[1] A B M Moniruzzaman and Syed Akhter Hossain, “NoSQL Database: New Era of Databases for Big data Analytics – Classification and Characteristics and