빅데이터환경에서의 칼럼-패밀리 저장소 활용방안

박승범[*] · 이상원[**] · 안현섭[***] · 정인환[****]

[*]한국정보화진흥원 경영기획부

[**]원광대학교 정보전자상거래학부(융복합창의연구소)

[***]브라운슈바이크공과대학 경영정보학과

[****]한성대학교 컴퓨터공학과

Application Plan of Column-Family Stores 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

요 약

키 값들을 대응하기 위해서. 데이터들은 칼럼-패밀리 저장소에 보존되는데, 빅데이터 환경에서 주로 사용되는 도구들은 Cassandra, HBase, Hypertable, and Amazon Simple DB 등이 있다. 본 논문에서는 Cassandra를 참조하여, 칼럼-패밀리 저장소와 그 구조를 정의하고, 그 특징들에 대해 살펴보기로 하겠다. 일관성, 트랜잭션, 가용성, 확장성 등에 대해 알아보고, CQL이라는 Cassandra 질의어를 이용한 기본 질의와 고급 질의를 살펴보겠다. 아울러, 본 칼럼-패밀리 저장소를 적용해야 할 분야와 적용하지 말아야 할 분야에 대해 언급하고자 한다.

ABSTRACT

Data so as to meet key values are preserved at Column-Family Stores such as Cassandra, HBase, Hypertable, and Amazon Simple DB in the Big Data environment. In this paper, with referring to Cassandra, we define column-family data stores and its structure. And then, we check out their characteristics such as consistency, transaction, availability, retrieval function (basic queries and advance queries) with CQL (Cassandra Query Language), and expandability. Also, we appropriate or inappropriate subjects for application of column-family stores.

키워드

Big Data, Column-Family Stores, Application Plan, NoSQL

I. Introduction

Data are preserved so as to meet key at column-family stores such as Cassandra, HBase, Hypertable, and Amazon Simple DB in the Big Data environment. With referring to Apache Cassandra, we research on column-family data stores and its structure. Apache Cassandra is an open source distributed database management system. It is designed to handle large amounts of data across many commodity servers, providing high availability with no single point of failure. With asynchronous master-less replication allowing low latency operations for all clients, Cassandra offers robust support for clusters spanning multiple datacenters. In the field of NoSQL systems, Cassandra achieves the highest throughput for the maximum number of nodes in the Big Data environments. In this paper, after checking out their

characteristics such as consistency, transaction, availability, retrieval function (basic queries and advance queries) with CQL (Cassandra Query Language), and expandability, we appropriate or inappropriate subjects for application of column-family stores.

II. Column-Family Stores for Big Data

The method to store and process data by column instead of row has its origin in analytics and business intelligence. Column-stores operating in a shared-nothing massively parallel processing architecture can be used to build high-performance applications. Sybase IQ, Cassandra, HBase, Hypertable, Amazon Simple DB and Vertica are notable products in this field. The Column-Family Stores collect and manage data that meet with key values (Figure 1). The values are classified into several column-families and each column-family becomes data map. Column-Family Databases store data in column-families by use of rows. Each row has many columns related to its key of the row. So, column-family means a collection of related data. Cassandra as a Column-Family Stores guarantees consistency, transaction, availability, various queries, and extensibility of column-family data. The function of queries is performed by use of CQL with basic query services and indexing techniques.

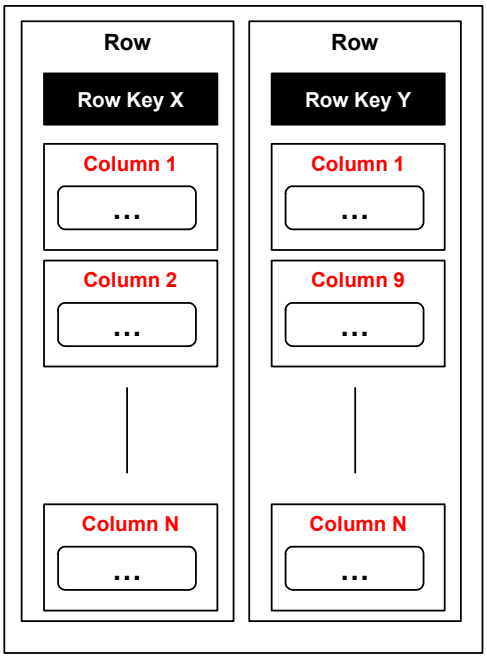

Figure 1. Data Model of Cassandra

III. Application Plan of Column-Family Stores for Big Data

The appropriate places of using Column-Family Stores are event logging (Figure 2), contents management systems, blogging-platform service, countering service, and expiring column service. But, Column-Family Stores cost a tremendous amount in query modification. So, it is not wonderful to use Column-Family Stores for prototyping or technical reviewing.

IV. Conclusions

Consequently, since column family contains columns of related data, it is a NoSQL object. It is a tuple that consists of a key-value pair, where the key is mapped to a value that is a set of columns. In managing Big Data, using column families could make it possible to store blog entries with tags, categories, links, and trackbacks in different columns.

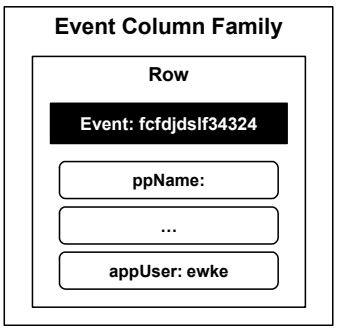

Figure 2. An Example of Event Column Family

References

[1] A B M Moniruzzaman and Syed Akhter Hossain, "NoSQL Database: New Era of Databases for Big data Analytics - Classification and Characteristics and Comparison," International Journal of Database Theory and Application, Vol. 6, No. 4, 2013.

[2] Burton H. Bloom, "Space/Time Trade-offs in Hash Coding with Allowable Errors," Communications of the ACM, Vol. 13, No. 1, pp. 422-426, 1970.

[3] Christof Strauch, NoSQL Databases, Stuttgart Media University Press, 2012.

[4] Edgar F. Codd, "A Relational Model of Data for Large Shared Data Banks," Communications of the ACM, Vol. 13, No. 6, pp. 377-387, 1970.

[5] John Ousterhout, Parag Agrawal, David Erickson, Christos Kozyrakis , Jacob Leverich, David Mazières, Subhasish Mitra, Aravind Narayanan, Guru Parulkar, Mendel Rosenblum, Stephen M. Rumble, Eric Stratmann and Ryan Stutsman, "The Case for RAMClouds: Scalable High-Performance Storage Entirely in DRAM," SIGOPS Operating Systems Review, Vol. 43, No. 1, pp. 92-105, 2010.

[6] Joshua Bloch, Effective Java - Programming Language Guide, Addison-Wesley Longman, 2001.

[7] Marcos K. Aguilera, Wojciech Golab and Mehul A. Shah "A Practical Scalable Distributed B-Tree," Proceedings of the VLDB Endowment, Vol. 2008, No. 1, pp. 598-609, 2008.

[8] Patrick O'Neil, Edward Cheng, Dieter Gawlick and Elizabeth O'Neil, "The Logstructured Merge-Tree," Acta Informatica, Vol. 33, No. 4, pp.

351-385, 1996.

[9] Pramod Sadalage and Martin Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Addison-Wesley, 2012.

[10] Stavros Harizopoulos, Daniel J. Abadi, Samuel Madden and Michael Stonebraker, "OLTP through the Looking Glass, and What We Found Fhere," ACM SIGMOD International Conference on Management of Data, Vol. 2008, No.1, pp. 981-992, 2008.