

Research of Data Storage and Querying Methods Based on Ring Distributed Hash

Ye Chen*

Fujian Jiangxia University, Library, Fujian, Fuzhou, 350108, China

Abstract: In this paper, the main contributions of this work include three aspects. First, the deployment on different data-centers of Impala which is a database based on Ring Distributed Hash. This thesis deploys Impala system on different data-centers across WAN or across regions. Second, the research of data storage and search method based on circular distributed hash. This thesis adopts distributed hash table and Distributed Hash ring to distributed file system. Every data and storage node will be calculated the hash value, by which we map the data to its storage node. According to the saved metadata, we take use of binary search method to find the location of data. Third, the research of data migration based on circular distributed hash. If a new node is added in Ring Distributed Hash, its "neighbor" will transport some data to it. Similarly, when a node breaks down, its data will be made backup to its "neighbor".

Keywords: Data storage, data querying, distributed, hash, ring.

1. INTRODUCTION

The "4V" character of big data: volume, variety, value and velocity, has made the original relational database cluster unable to hold huge structured data. Therefore, the database based on distributed file system has been a research hotspot. It takes the Hardtop Distributed File System (Distributed Hash) to store data and adapts Massively Parallel Processing architecture as scheduling engine. The Distributed Hash is always deployed on some nodes which have independent infrastructure and connect to each other via Internet [1].

Among them, one is responsible for metadata storage and others for file data. All data communication information is transformed by network. Currently, the relational database based on Distributed Hash has those following insufficiencies: First, it cannot be employed on different datacenters, which has no query function. [2] Second, it distributes data belongs to one table to less storage nodes without any optimization, that deeply reduces the concurrency of table traversal in effect. Third, to load balancing, the data must be migrated when the amount of nodes changed. All nodes are involved in migration process, thus the long migration has bad influence on real-time query to database based on Distributed Hash.

Supported by 242 issue "Research and verification and cloud storage key technologies of relational data", this thesis focuses on exploratory research of data storage, search and migration method from the aspect of Distributed Hash storage. And we achieve to employ and query the relational database based on Distributed Hash on different datacenters.

Recent years, cloud computing has become one of the latest research area in both academe and IT industry because of its high-performance and scalability for storing and computing on large-scaled data. [3] Nowadays, Hardtop technologies have become the de-facto standard of Big Data processing. Several researchers have started to combine cloud computing and semantic web technologies to explore high-performance ontology querying and reasoning solutions in the distributed computing context. However, this novel research area is still in the initial stage, lots of key problems need to be solved.

With the growing size of data on the Internet, it's becoming increasingly difficult for people to access information accurately and quickly. Through adding semantic support to the Web, Semantic Web enables the machine to understand the meaning of the data and help people access information quickly. RDF (Resource Description Framework, Resource Description Framework) is the standard data model for exchanging data in Semantic Web, which describes semantic data in triples like <subject, predicate, object>. [4] With the continuous improvement and widespread use of the Semantic Web Technology, RDF data size increases sharply. The emergence of massive RDF data brings great challenges to RDF management system. It become a research hotspot in the field of semantic web to research on how to construct scalability RDF storage and query system.

Map Reduce is an emerging parallel programming technique for processing massive data in recent years. Hardtop is a famous open source implementation tool of Map Reduce. Using Hardtop platform to process massive data has been widespread concerned.

To overcome the drawbacks, this thesis researches on the approaches of distributed querying and reasoning for large-scaled ontology data by utilizing cloud computing technologies. This thesis can establish the theoretical research basis

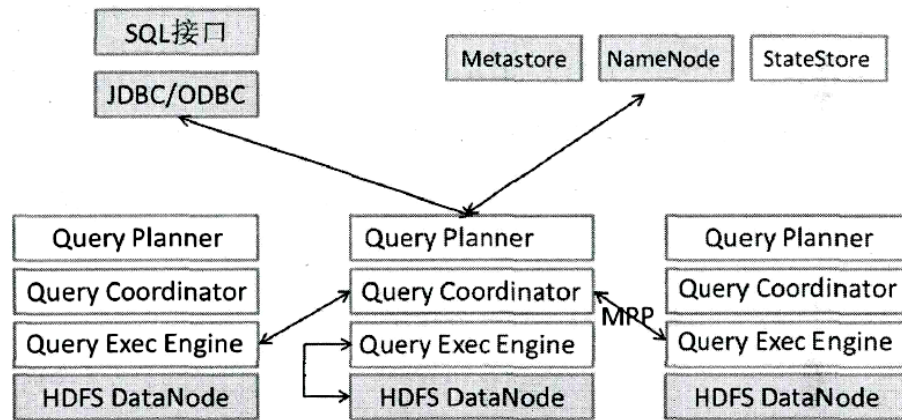


Fig. (1). Impala system architecture.

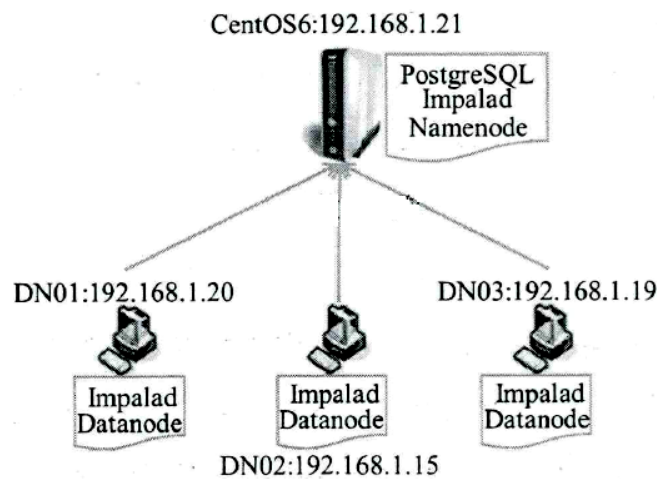


Fig. (2). Data center impala system.

for implementing large-scaled semantic web ontology data management cloud services in the future.

2. QUERYING LANGUAGE

With the rapid development of the Internet and the Internet of things, the amount of data on the net is growing explosively. That puts forward a new challenge of data sharing and management; there are more complex semantic relations need to be managed and used in big data conditions. The data storage, retrieval of the Large-scale RDF data and data processing method to supporting the data mining, have important theoretical and practical sibilance for computer, manufacturing, railway and other industries. under the requirement of railway sensor data applications, this paper puts forward a kind of Base-based RDF data storage and query method using structured index (Fig. 1).

Firstly, considering the requirements of big data storage, the paper puts forward a kind of structure-based RDF data index method. By the analysis of the connection relation among the nodes of data graph, this method utilizes the index structure to implement data partition guaranteeing aggregation storage of data with same structure. It can reduce

join operation during query processing, and accelerate query speed [5].

Secondly, this paper utilizes Base to manage RDF data according of structure-oriented data partition, and uses "predicate: subject-object" triple pattern as the storage structure in Base. Additionally, it proposes the row-key encode method to solve the multi-value problem of RDF data which have same subject. As a result, it effectively reduces the range of the data query, and improves the efficiency of query.

Thirdly, it brings forward the query method of RDF data utilizing structure index and SPARQL rewrite. According to the relationship of shared elements in different query statement, and the weights of executive a query, correlation degree is calculated to support SPARQL-rewrite. After SPARQL rewrite, the realization of data query depends on two layer query from the structure index to physical query, and enhances the query efficiency.

Finally, according to the railway sensor scenarios, this paper makes experiment verification for the efficiency of the system. Compared with the classical RDF data storage and retrieval system-Sesame, our system gains a better query efficiency (Fig. 2).

The Semistructured Data Model

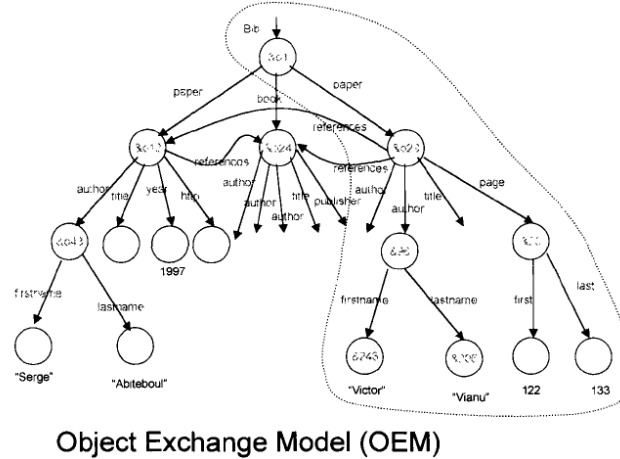


Fig. (3). OEM Data model.

According to the background knowledge of the XML spatial metadata and the most popular relational database SQL Server, this paper uses the remote sensing image data to solve the storing problem of XML spatial metadata in China. At present, when dealing with large amounts of data, the file system has inherent defect. What's more, object-oriented database systems and native XML databases are immature. Therefore, the coming period, the proportion of storing and querying spatial data management based on relational database XML spatial metadata is still large.

Because of the XML(Extensible Markup Language)'s capability of representing various kind of information, it will be the standard of exchanging information over the next generation Internet. XML Data Model is hopeful to be a coordinate with Relational and Object-Oriented Data Model. [6] One research focus in the database community is how we can integrate databases and XML to explore the theory and technology of XML-Database. In this thesis, on the basis of analyzing and understanding W3c's specifications of XML and XQuery (XML Query), we mainly investigate on aspect of XML databases that is storing and querying XML. First, a processing model of XQuery query engine is proposed from the research of XML query language. Second, the XML data models are studied in order to support XML-query. Finally, we investigate and implement the storing and querying of XML based on relational databases. In the research of XML query language, we consider the requirement of two categories (for data-centric and for document-centric) of XML applications and point out that the basic operations in XML-query are pattern matching, selection and filtering, and result reconstruction. The path expression and the FLWR expression in XQuery are typical structures that are comprised of these operations. According to W3C, the execution of XML-query involves four internal phases, i.e. Parsing, Normalization, Static Type Checking, and Dynamic Evaluation [7]. It also involves three input/output phases namely Schema Import, XML Loading, and Serialization. In study of XML data model, we investigate the semi-structure characteristics of

XML: self-describe, graph-based model and irregular. Then the comparison and analysis of normal tree-structure model and node labeled direct graph are presented. We draw the conclusion that the XQuery data model is a tree-like model with various types of nodes. The nodes are constructed by constructor and accessed by accessory. Both of them are defined in XQuery data model. It is our viewpoint that XML database research will receive benefits from the well-established relational database theory and technology especially in XML databases' early stage. For this reason, in our study, we use relational databases storing and querying XML. The key problem of storing XML in relational databases is how to design a set of table cleverly and rationally for storing XML. To solve this problem, the first method is that we use XML's DTD (or Schema) to generate table definitions for storing XML-Schema-based method. An alternative way we use is that we store both the data and the structure (especially path) of XML into several predefined tables-Path-based method. We support XQuery by using a method basing on the data model; we call it DM-method. Using this method, the physical store of XML is hid from XQuery query engine and the support for XQuery is achieved over relational databases. In particular, we make two contributions. First, we give processing query engine according to methods that can store XML W3C XQuery language. Second, we model of XML implement two in relational databases, and propose a method that can hide relational storing manner from XQuery engine (Fig. 3).

3. ANALYSIS BASED ON DISTRIBUTED HASH METHODS

With the dramatic development of WWW (World Wide Web), web data witness an explosive boom up in both quantity and scale, which makes web data forming the hugest database in the world. Moreover, other data associated with web data, such as search engine records, (Fig. 4) and click records on web service, is also growing rapidly. Compared

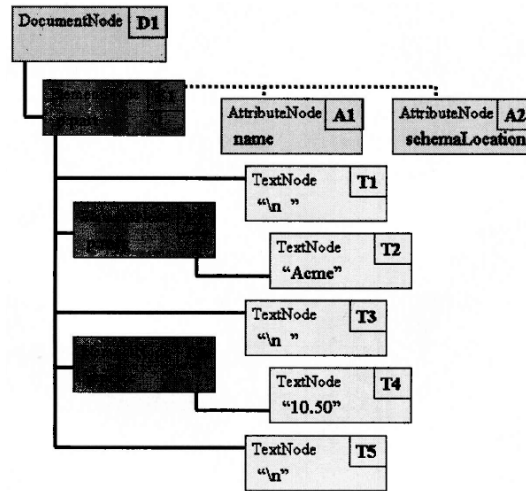


Fig. (4). DM.

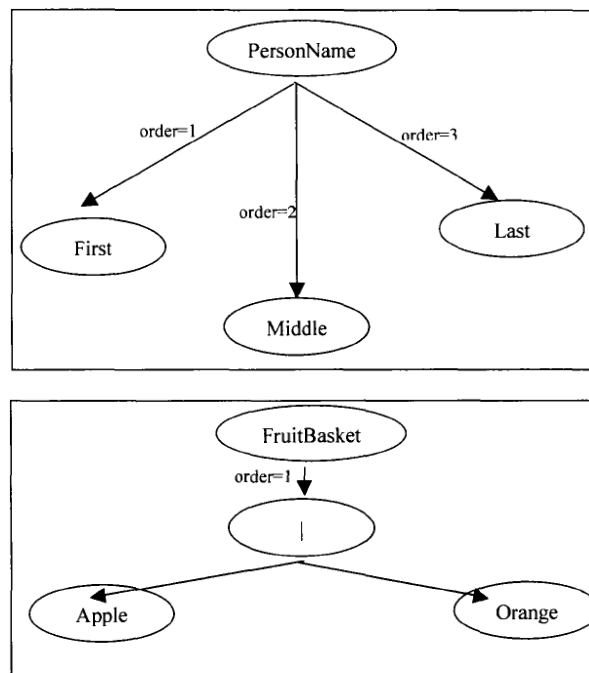


Fig. (5). Layered.

to traditional data, web data is semi-structure data and has the character of high increase rate, variety data type. Therefore, it is unlikely to deal with traditional data and web data in the same way. Nowadays, there is a large demand of web data analyses technology in all fields, which has been attracting increasingly attention in relevant area of database research.

Therefore, we introduce in CWI, a new query and analyze tool for massive data. In realistic application, we need to store and query large scale of data, implement keyword searching and query on data structure. As parts of CWI, TLGM and TLGM-QL meet these demands. We emphasize on implementing TLGM data model in the distributed environment, and we design and implement four basis operators supporting TLGM-QL. During the designing phase, we find that the disproportional spacing real world data will induce to degeneration of the store and query algorithms, which

increase the time cost. In order to solve this problem, we bring forward a series of algorithms to keep the difference between data nodes' storage and calculation load in a bearable scope. On this base, we bring a new reconstructing algorithm for sub graph to support query on graph structure. We also propose several balancing methods to ensure the efficient of algorithm above.

We design and run experiments on virtual and real world data to prove the system's efficiency.

Web data's character is analyzed, and TLGM model is introduced to illustrate the difference between web data and traditional data on storage, querying and indexing. Firstly, we try using relational database to store graph data, designing several queries and making experiment on it. By checking the experiment results, we show the limitation of centralized storage (Fig. 5).

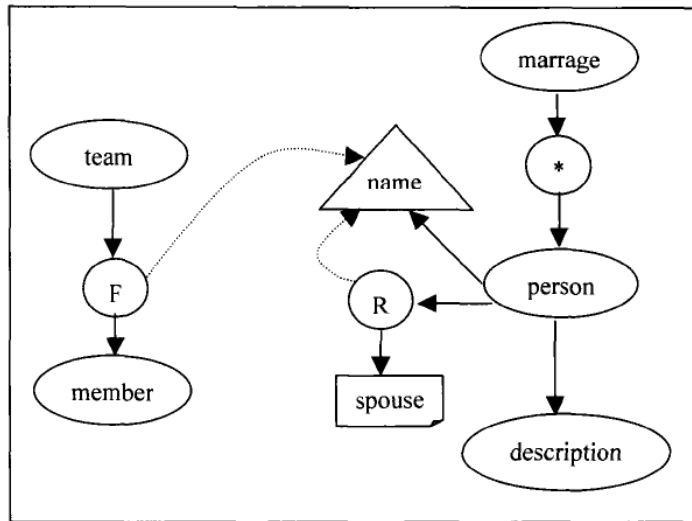


Fig. (6). Data storage.

We analyze the TLGM model and illuminate its implementation under distributed environment. Furthermore, we summarize the query language supported by this model, and propose four basis operators. We use some examples to prove these operators have good flexibility, and then we provide the pseudo code of them.

3.1. Key Codes

```

BinSearch(L, B;d)
{max=m
min=1
while(max>=min)
{ If(HaSh(D [(max+n,in}2])>= Bid)
{If(Hash(D[({X+min)-, ])>=Bid)
=max=(max+min)/2-1
else
D[(~+ min)/2]}
else
{if(Hash(D[(n)/2+I])C Bid)
min=(max+min)/2+1
else
D[(max+min)/2+I]

```

3.2. Data Storage and Query

With the rapid developments of the information technology and the emerging network applications, the electronic data increase sharply. As a result, it makes the resource requirement for storing and backspin these mass data grow

exponentially, and the scales of data centers are developing towards the PB level, or even the EB level. The related research shows that 60% of the data, on average, in the various application systems are duplicate, and the volumes of duplicate data keep on growing as time goes on. For example, there exist a lot of duplicate data in the office automation system, the archival storage system, the group mail and attachments, the Web pages, and the software release. In addition, the periodic backups accelerate the increase of duplicate data even more. The repetitive storing and transferring of these duplicate data consume a great deal of storage space and network bandwidth which increases the cost of data management significantly. Consequently, it has become a hot research topic in the area of backup and storage that improving the resource utilization and reducing the cost by the reduplication technology.

For a mass backup system based on inline reduplication, it is needed not only to improve the data compression performance, but also to ensure the security feature and the high throughput performance. Therefore, the research works of this dissertation mainly focus on the following points: improving the compression performance of the duplicate data detecting method, improving the data recovery performance of the linear delta chain, improving the data security of the reduplication system and improving the throughput of the reduplication system.

This method mitigates the contradiction between improving the compression performance and reducing the metadata cost by using different chunking strategies on the data changing region and non-changing region respectively (Fig. 6). So, the bottleneck of further improving the compression performance is broken through. This method can achieve a satisfying compression ratio with a relatively larger expected chunk size, and its time cost is much lower than the current tasteful detecting method.

The needed version file can be obtained without calculating the middle version files. So, the data recovery

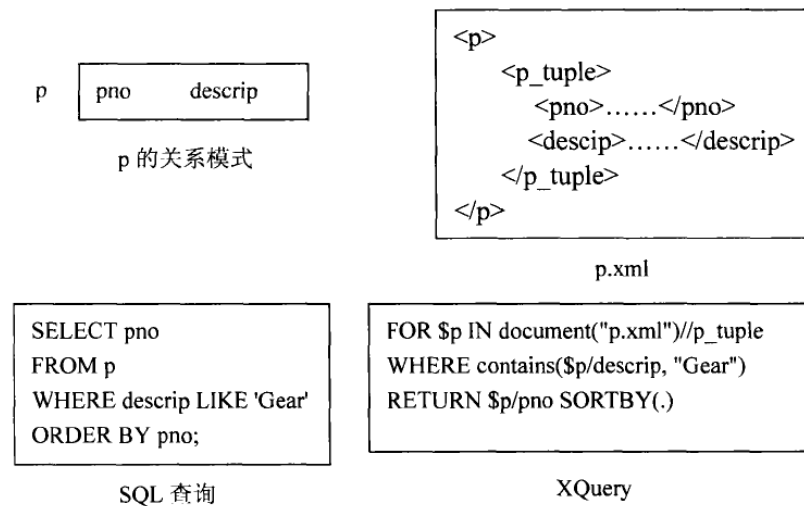


Fig. (7). SQL->XQuery.

performance of the linear delta chain is improved notably and its optimal compression performance is retained at the same time. The compression performance of the delta backup system based on this algorithm is much higher than the system based on the jumping version chain and the time cost of data recovery is much lower than the traditional data recovery method. This method uses the chunk as the basic unit of encryption and the symmetric keys used to encrypt the chunks are generated by a convergent method. Thus, the influences caused by the disagreement of key choosing and the avalanche effect of encryption algorithm are eliminated. This method solves the problem that the traditional encryption method is not compatible with the deduplication technology, so it can ensure the data confidentiality and the compression performance of the system simultaneously.

Two throughput-improving methods, which do not depend on the data locality of backup loads, are proposed. Firstly, a throughput-improving method that is applicable to the mixed backup loads is designed. Then, on this basis, a throughput-improving method that is suitable for the distributed application environment is proposed. These two methods solve the locality dependent problem of the current throughput-improving methods, and can process the non-traditional backup loads effectively. Both of them can achieve a near-optimal compression performance and a satisfying throughput performance.

4. SIMULATION

A novel algorithm of sub graph reconstruction is proposed, which is used to support query on graph structure. We implement this algorithm in the Map Reduce framework, which makes it having good scalability. In addition, we use cache strategy to improve the efficiency. We also make some improvement on balancing methods for real word data often cause imbalance load between different data nodes. Extensive experiments are performed to verify the efficiency of our algorithm (Fig. 7).

We believe our work is a good example of web data storage and querying with practice since we not only provide some key solutions for storing web data as graph, but also implement a novel framework to index and query massive web data. Our work has great importance in web data storing area.

Because XML data has its own characters that are different from the traditional data form, the mature technology of traditional database can not work efficiently. Therefore, it is necessary to research new processing method for XML data according to its special characters. And as one of the most important problems of XML data processing, how to store and query XML data efficiently has become a hot topic of the research on XML data management recently. To resolve the problem of storage of massive document without fixed scheme improve the performance efficiency of holistic path join algorithm, this paper focuses on the two important technique facets in storing and querying join algorithm on XML data, which based on fixed labeling scheme.

Firstly, according to the question that most of the present labeling schemes can not efficiently support the dynamic updating and the storage retrieval, a new dynamic labeling scheme, called Preorder, is proposed in this paper. The scheme uses a 3-tuples to represent the label of nodes; it can efficiently support the updates of XML document, the number of nodes need to re-label is very small when the updates occurred in document. At same time, the three parts of label combine with each other; it can preserve the structure of document and also can give the correct retrieval of original document.

5. DIATRIBUTED STORAGE

First, we present the architecture of the distributed data storage and management system. The inter-object relationship based semantic model and the concepts of object and inter-object relationship is presented. Then, the query and

retrieval language and query processing is described. After that the system and sub-system architecture is briefly introduced.

Based on peer-to-peer and object-based storage technology, we designed a container-based distributed object storage. In the container-based storage model, a container which is the unit of data placement and replication, manages a set of storage objects and is responsible for dirty tasks such as block allocations. The container-based storage model reduces metadata size, simplifies system design, and hence improves storage system scalability. The system maintains runtime metadata by self-organizing peer-to-peer technology, and handles server failure and addition transparently. It achieves reliability by primary/slave container replication with dynamic primary election, achieves consistency by state-based object access and replica healing.

Distance index is an essential data structure of query processing in data management systems. However, the creation and query performance of existing index methods is far from perfect. We propose two indexes for directed graph: DIX-C with constant query time, DIX-2HC with smaller index size, and also describe their undirected version UDIX-AP and UDIX-2HC. The related-query processing algorithm is then presented based on these distance indexes and interval encoding. The experiment results show that our indexes outperform previous methods, and that related-query processing algorithm is very efficient.

Existing access correlation discovery methods often rely on support to prune search space, and hence can't detect many valuable correlations with low support. Furthermore, they are not scalable enough to work in distributed environment, and none of them can detect inter-server correlations which is prevalent in distributed storage system. Therefore, we propose access correlation mining algorithms, namely HCM, VCM and PFC-Miner, which employ correlation confidence as the primary interest measure. HCM is more efficient, VCM can run incrementally, and both of them work only on stand-alone server. In contrast, PFC-Miner is a distributed approximate mining algorithm which is very scalable and able to discover inter-server correlations. Experimental results demonstrate the performance of proposed algorithms, and also show that the mined access correlation can be utilized to improve cache hit ratio.

In storage system, files usually have large number of replicas and even more highly similar replicas. While existing methods consider only identical replicas, we propose a keyword extraction method PAKE which detects files with high content similarity and subsequently extracts keywords from

their names. The experiment results show that PAKE can improve retrieval significantly compared to previous methods.

CONCLUSION

The main innovative contributions of this thesis are as follows. First, Delong Impala system to datacenters across regions, which improves the impala's business scope and offers ort for big data application across regions; Second, proposing data storage and search method based on circular distributed hash, which distributes the data evenly, improves the concurrency of table traversal and reduces the query time; Third, proposing data migration method based on circular distributed hash, which saves process time, and guarantees the consistency and effectiveness for database usage.

Finally, this thesis has given the emulation experiments for the data search and migration method and has proved the effectiveness of the proposed methods, in comparison with the original distributed Hash ring strategy

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

- [1] E. B. Randal, H. K. Randy, and D. L. Edward, "Big-data computing: creating revolutionary breakthroughs in commerce, science, and society," *Computing Community Consortium*, Version 8, 2008.
- [2] A. Divyakant, B. Philip, B. Elisa, et al. "Challenges and opportunities with big data," *A Community White Paper Developed by Leading Researchers Across the United States*, 2012.
- [3] K. Avita, W. Mohammad, and R. H. Goudar, "Big data: issues, challenges, tools and good practices," In: *6th International Conference on Contemporary Computing*, 2013, pp. 404-409.
- [4] S. Chen, "A high performance, custom data warehouse on top of MapReduce," *Proceeding of the VLDB Endowment*, vol. 3, 2010, pp. 1459-1468.
- [5] J. Zhao, "Publishing Chinese medicine knowledge as Linked Data on the Web," *Chinese Medicine*, vol. 5, pp. 1-12, 2010.
- [6] R. Maria, F. Joachim, and H. Joachim. Linked Environment Data. http://www.w3.org/egov/wild/Linked_Environment>Data, 2010
- [7] J. Goodwin, C. Dolbear, and G. Hart, "Geographical linked data: The administrative geography of Great Britain on the semantic web," *Transactions in GIS*, vol. 12, pp. 19-30, 2008.