International Journal of Economics & Management Sciences

Lina and Chen, Int J Econ Manag Sci 2015, 4:10
http://dx.doi.org/10.4172/2162-6359.1000300

Opinion Open Access

Big Data Hadoop for Cloud-Based Enterprise Collaboration Systems

Hsiao-Kang Lin and Chun-I Chen*

Department of Industrial Management, I-Shou University, Kaohsiung, Taiwan ROC

Abstract

With the growing popularity of cloud computing, enterprises are turning their collaboration platforms towards Software as a Service (SaaS) applications over the Internet. Companies hosting cloud-computing services face the challenges of multi-dimensional information and knowledge from a variety of distributed sources. Semantic web technologies provide the solution to semantic interoperability of heterogeneous data. In addition, the NoSQL Hadoop database (HBase) is being adopted dramatically on the cloud for semantic data integration. This paper investigates a Hadoop Database (HBase) approach for semantic web information representation across distributed and collaborative enterprises systems.

Keywords: Cloud-based enterprise collaboration system; Semantics web; Hadoop Database (HBase)

Introduction

In recent years, modem enterprises systems have undergone profound changes driven by the development of synergistic operating networks and cloud service-oriented models. With the growing popularity of cloud computing, enterprises are turning their collaborative platforms towards Software as a Service (SaaS) via the Internet. According to the (Deloitte) report [1], called this as digital collaboration that delivering innovation, productivity and happiness. To work with digital collaboration in the virtual enterprises is suffused with multi-dimensional information and knowledge from a variety of distributed sources. Semantic web technologies (e.g., RDF/RDFs/OWL) are crucial for supporting semantics-enabled applications and provide the solution to semantic interoperability on the web data.

The knowledge-based collaboration project, in particularly SYNERGY, provides the necessary technological infrastructure for supporting enterprise collaboration that allows the sharing of knowledge within Virtual Organizations (VOs) to the mutual benefit of the VO partners (Popplewell, Stojanovic et al.) [2]. Collaboration Moderators (CM) is a major component of the SYNERGY project. A CM is a specialist application for supporting individual collaboration partners by raising awareness of issues affecting items of interest identified by the members of VO. The prototype CM was implemented as a web-based software infrastructure. It is an integration platform that utilizes web services standards to support Service-Oriented Architecture (SOA) within VOs. The information and knowledge is stored using semantic web technology. A common RDF schema - Virtual Organization Knowledge Base (VOKB) is modeled on pre-agreed schema and stored onto relational databases within the collaborative VOs for knowledge sharing (Dai, Popplewell et al., Harding and Swarnkar) [3,4]. The participation partners in the VO require transforming/mapping their origin database schema into the VOKB to facilitate semantic data integration and interoperability.

However, Vora [5] pointed that it is becoming increasingly difficult to share and exchange digital data efficiently and economically via the predefined schema of conventional SQL databases. To respond this problem, HBase is being widely used for storing, processing and analyzing distributed semantic web data management [6,7]. HBase is a column-oriented database that does not require predefined schema and has been adopted dramatically on the cloud for semantics data integration. This paper proposes a novel approach for a semantics web representation of the information stored in a set of the Hadoop Database (HBase).

Design of VOKB Model Based on HBase

In this paper, we used the RDF and the HBase table to establish a data integration platform for the Virtual Organization Knowledge Base (VOKB), defined by the SYNERGY project. The VOKB basic classes and their relations for VOKB ontology are presented in Figure 1. A VO is a short-term association with a specific goal of acquiring and exploiting a business opportunity. However, despite the short-term nature of the VO, to operate successfully partners in a VO must share knowledge and information to a significant degree. In addition, there is a body of knowledge, which is relevant to the VO as a whole, regarding structural, and operation aspects of the VO. Again mutual understanding of this knowledge amongst partners is essential to the success of the VO. The VO Knowledge Ontology therefore describes the terminology in the VO knowledge. For example, A **Virtual Organisation(VO)** has VO Partners each of which is an **Enterprise**s collaborating with other enterprise who are in turn **VO Partner**s in the same VO. A VO has **VO Activity** and a common goal to carryout **VO Project** in order to deliver **VO Product**. The details of the VOKB ontology structures can be found in SYNERGY project deliverable reports [3].

Building the "Enterprise Table" for Supporting the Web Content

Dimiduk and Khurana [8] described that a relational database as storing a piece of data in a table in a 2D coordinate system based first on row and second on column. By that analogy, HBase stores a piece of data in a table based on a 4D coordinate system. The coordinates used by HBase, in order, are table, rowkey, column family: column qualifier, version (time-stamp) ---> value.

Rowkeys are the single most important aspect of an HBase table design and determine how the application will interact with the HBase tables. Data within a column family is addressed via its column qualifier, or column. Column qualifiers need not be specified in advance. In other words, Column qualifiers are dynamic and can be defined at write time.

***Corresponding author:** Chun-I Chen, Department of Industrial Management, I-Shou University, Kaohsiung, Taiwan, China, Tel: 886-7-6577711; E-mail: EddyChen@isu.edu.tw

Received November 23, 2015; **Accepted** November 24, 2015; **Published** November 28, 2015

Citation: Lin HK, Chen CI (2015) Big Data Hadoop for Cloud-Based Enterprise Collaboration Systems. Int J Econ Manag Sci 4: 300. doi:10.4172/21626359.1000300

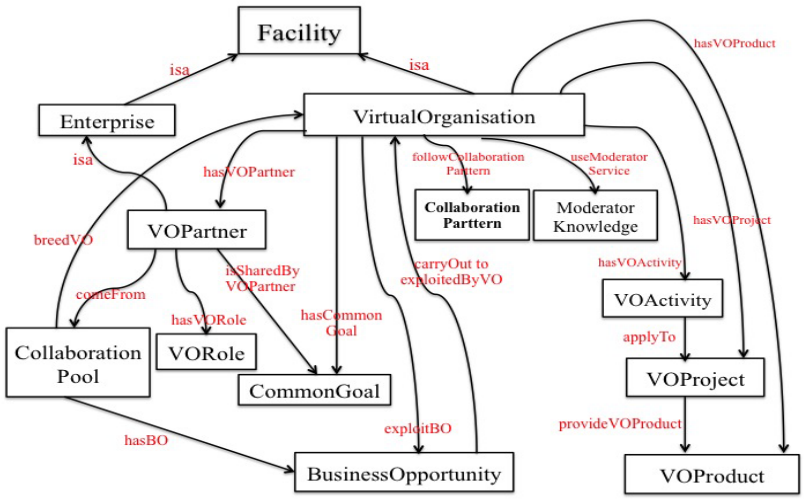

Figure 1: The VOKB ontology [3,10].

rowkeys URI of the Enterprise)	<CF (info)> : <CQs>				
	<info>:<name>	<info>:<contact info>	<info>:<contact email>	<info>:<core business>	...
com.rubicontechnology.www	Rubicon Technology	http://www.rubicontechnology.com/contact-us	jsooo58885@ gamil.com	Sapphire Substrates	...
com.auo.www	AUO	http://www.auo.com/?sn=97&lang=zh-TW	yuyulee588@ auo.com	Screen	...
com.dkaztec.www	DK Aztec	http://www.dkaztec.com/Eng/Etc/contactus	cloudofsky712@ dkaztec.com	Sapphire ingot	...
com.teraxtal.www	ACTC	http://www.teraxtal.com /connect_page.html	iechnssun@teraxtal.com	Sapphire Substrates	...
com.mediatek.www	MediaTek	http://www.mediatek.com/zhTW/about/contact/	v2859809@ mediatek.com	CPU chips	...
com.monocrystal.www	Monocrystal	http://www.monocrystal.com/en/contacts	yunyispecial89@gamil.com	Sapphire ingot	...
net.namiki.www	Namiki Precision Jewel	http://www.namiki.net/product/contact/index.html	twotwo112422@namiki.com	Sapphire ingot	...
.....	.....	.....	.....	.....	...
.....	.....	.....	.....	.....	...
tw.largan.com.www	Largan Precision Co.	http://www.largan.com.tw/html/contact.asp	zebragrape1126@gmail.com	Digital Camera	...
tw.com.usio.www	USIO	http://www.usio.com.tw/tw/contact.php	e852741x@ usio.com	Sapphire ingot	...

Table 1: Logical Enterprise HBase table.

This is what the NoSQL supporters mean that HBase is a schema-less database, making HBase flexible and highly scalable. Each cell in a table can contain multiple versions of the same data – these versions are indexed by timestamp. If not specifies, data with the latest timestamp will be read by default. Thus in the rest sections of this paper, we'll not consider timestamp explicitly. More descriptions of these six logical entities can be found in Table 1.

In this paper, we create the web index of the global smartphone value chain and use HBase to store all web table content. Table 1 shows a logical data model of HBase table called "Enterprise" with one column family - "info" and four column qualifiers for storing web information about the enterprise/organisation, such as the name, contact information, contact e-mail and core business…etc. The rowkeys of the "Enterprise" table are derived from the URIs of the organisation's home page. For example, the URI of Rubicon Technology (http://www.rubicontechnology.com.) is stored as the rowkey value in HBase table. According to Dimiduk and Khurana [8], rowkeys effect on read or write performance out of HBase. They suggested keeping the rowkey as short as is reasonable that would be lot easier to "scan". Also found in the HBase chapter of (White 2012)'s book, it is recommended to hold the URL key as: reversed domain, such as "com.rubicontechnology.www".

Building the "VO Project Table" in NoSQL Column-Family-Oriented Datastore

Another common used for the HBase logical data model showed in Table 2: logical VOProject table. Examples of two VOs projects, PhoneS and WatchS are presented. Both projects plan to improve the mobile device screen protection on their new models by using sapphire screens, which are naturally strong, extremely scratch resistant, can withstand flexing and transmit light very well. However, the main drawback to the use of sapphire for large-screen smartphones is that they would be difficult to produce and fragile.

The rowkeys of the VO Project table are derived from the URIs of the project home page. For example, the URI of "PhoneS" is converted to http://www.PhoneS.com/#project and store in reversed format as, com.PhoneS/#project. The table schema has two Column Families/CF:

Rowkeys URIs of the VO	CF (vo info):		CF (vo partners):	
	CQs	Value	CQs	Value
com.PhoneS/#project	title	PhoneS	1	infineon
	number	216089	2	sunnyoptica
	project period	01/02/15 to 31/05/16	3	tpk
			4	foxconn
			5	dynapack
			6	teraxtal
			7	isu
	…	…	…	…
com.WatchS/#project	title	WatchS	1	mediatekm
	number	608627	2	largan
	project period	01/07/15 to 30/03/16	3	usio
	…	…	…	…

Table 2: Logical VOProject HBase table.

Key			Values
rowkey	column key (CF: CQs)	time-stamp	
com.PhoneS/#project	"vo info" : "title"	1329088321289	"PhoneS"
com.PhoneS/#project	"vo info" : "number"	1329088323354	"216089"
com.PhoneS/#project	"vo info" : "project period"	1329088326598	"01/02/15 to 31/05/16"
……	……..	……..	……

Table 3: Physical HDFS for the "vo info" column family in the VOProject table.

Key				Values
rowkey	column key (CF: CQs)		time-stamp	
com.PhoneS/#project	"vo partners" :	"1"	3329386321267	infineon
com.PhoneS/#project	"vo partners" :	"2"	3329386323309	sunnyoptical
com.PhoneS/#project	"vo partners" :	"3"	3329386326588	tpk
……	……	…	……..	……..
com.PhoneS/#project	"vo partners" :	"6"	……..	**teraxtal**
com.PhoneS/#project	……..	"7"	……..	**isu**
……		……	……..	……

Table 4: Physical HDFS for the "vo partners" column family in the VOProject table.

"vo info" for storing information about the project, such as the title, number, and project period; and "vo partners" for storing the number of participations with individual partner' name as value, showed in Table 3. Moreover, each column family gets its own set of file physically stored on disk.

One of the strengths of HBase over a relational database is that the designers don't have to specify the columns (Column Qualifiers : CQ). If current project require certain new number of partners, we can just insert them without modifying the columns/CQ schema. As we can see from Table 2, columns/CQ are not pre-declared; they are essentially just added additional label for new column/CQ with value. For example, a new partner joins the PhoneS project. Simply add a new column/CQ called "6" with value "teraxtal", "7" with value "isu"…etc. Also notice that we haven't needed to change the table definition so far. That's HBase's NoSQL schema_less data model at work.

According to Vora [5], the HBase is a distributed column-oriented database built on top of Hadoop Distributed File System (HDFS). All column family members are physically stored together on the HDFSs. That is, each column family gets its own set of HDFSs on disk. White [9] described that it would be more accurate if it were described as a column-family-oriented store. VOProject data stored on disk in an HDFS looks something like Tables 3 and 4, physical HDFS for the "vo info" column family and the "vo partners" column family in the VOProject table [10,11]. Records in HBase are stored in the files as key-value pairs [12]. That is, HBase maintains maps of Keys to Values, as the following:

rowkey, column key (Column Family CF : Column Qualifiers CQs), timestamp --> value.

Conclusion

The Hadoop Ecosystem running in the cloud is generating new opportunities and presenting new challenges businesses across every industry sector. The challenges of data integration - incorporating data from the web and other unstructured data from multiple sources, is one of the most urgent issues facing the successful implementation of the manufacturing collaboration system. Based on our evaluation, HBase promote a new way of thinking about the data-processing pipeline. Although HBase still needs performance improvement, it shows real promise of becoming a mainstream solution.

References

1. Deloitte (2013) Digital Collaboration: Delivering innovation, productivity and happines.
2. Popplewell KN, Stojanovic A, Abecker D, Apostolou G, Mentzas, et al. (2008) Supporting Adaptive Enterprise Collaboration through Semantic Knowledge Services, Enterprise Interoperability III. Springer, London: 381-393.
3. Dai XK, Popplewell, Wulan M (2009) Collaboration Knowledge Services Framework. Synergy: Seventh Framework Programme ICT.
4. Harding JA, Swarnkar R (2013) Implementing collaboration moderator service to support various phases of virtual organisations. International Journal of Production Research 51: 7372-7387.
5. Vora MN (2011) Hadoop-HBase for large-scale data. 2011 International Conference on Computer Science and Network Technology (ICCSNT) 601-605.
6. Franke CS, Morin A, Chebotko J, Abraham, Brazier P (2013) Efficient Processing of Semantic Web Queries in HBase and MySQL Cluster. IEEE Xplore digital library 15: 36-43.
7. Yang CT, Liu JC, Hung WH, Lu HW, Wang W, et al. (2013) Implementation of Data Transform Method into NoSQL Database for Healthcare Data. Parallel and Distributed Computing, Applications and Technologies, 2013 International Conference: 198-205.
8. Dimiduk N, Khurana A (2012) HBase in Action. Manning Publications Co, Newyork.
9. White T (2012) Hadoop: The Definitive Guide (3rd eds.) O'Reilly Media.
10. Hepp M (2008) GoodRelations: An Ontology for Describing Products and Services Offers on the Web. EKAW: 329-346.
11. https://research.linagora.com/display/synergy/Synergy+knowledge+oriented+collaboration
12. Teswanich W, Chittayasothorn S (2007) A Transformation from RDF Documents and Schemas to Relational Databases. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, IEEE Xplore digital library: 38-41.

Citation: Lin HK, Chen CI (2015) Big Data Hadoop for Cloud-Based Enterprise Collaboration Systems. Int J Econ Manag Sci 4: 300. doi:10.4172/21626359.1000300