XML OLAP Cube in the Cloud Towards the DWaaS

Rihane DKAICH, MISC Laboratory, Faculty of Sciences, Kenitra, Morocco
Ikram EL AZAMI, MISC Laboratory, Faculty of Sciences, Kenitra, Morocco
Abdelaziz MOULOUDI, MISC Laboratory, Faculty of Sciences, Kenitra, Morocco

ABSTRACT

Datawarehouses can be extremely large and resource demanding, which is not always affordable in a local environment. Hence, in order to deal with the big amounts of data held in the datawarehouses, Cloud warehousing seems to be the solution. On the other hand, many enterprises use datawarehouses for data analysis and use XML to deal with semi-structured data but also to take advantage of the web environment. Therefore, the idea of combining the two solutions in a parallel environment seems necessary. Thus, the goal of the study presented in this paper is to use XML to store and exchange data and to connect it to the distributed processing of multidimensional data. This article deals with the problematic of storing documents in distributed environments such as the cloud nodes, and discusses the possibility of combining the storage of data and the decisional analysis based on OLAP cubes in cloud environments, using the MapReduce model for query processing.

KEYWORDS
Cloud Computing, Datawarehouse, MapReduce, OLAP Cube, XML

INTRODUCTION

Datawarehouses store big amounts of historized data that the decision makers use to run queries, establish reports and for analysis and datamining. They can be extremely large and resource demanding, while queries and analysis should not exceed a certain response time. Hence, datawarehouses face challenges of size, speed and distributed operations processing (Furtado, 2009), which lead the authors of this paper to the cloud warehousing allowing them to run massive volumes of data coming from different sources and to guarantee as fast as possible responses to the end users. On the other hand, XML is a standard established for semi structured data that also offers many advantages in the web environment (Sen, Ghosh, Paul, & Chaki, 2012). Data may be provided by heterogeneous data sources (Tseng & Chen, 2005), and for data integration they can be converted into XML data. Moreover, XML and databases are more complementary than competitive. In fact, by revealing a non-tabular data structure, XML enables this structure to be used by databases technologies. Conversely, the databases techniques can enhance the integrity and semantic integration of XML resource sets (Seligman & Rosenthal, 2001). This paper deals with the problematic of storing documents in distributed environments such as the cloud nodes, and discusses the possibility of combining the storage of data and the decisional analysis based on OLAP cubes in cloud environments, using the MapReduce model for query processing.
RELATED WORK

Extensible Markup Language (XML) is a simple, very flexible text format derived from SGML (ISO 8879). Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere (Quin, 2016). Thus, many studies have focused on XML documents distribution to store XML data in distributed environments.

Distributed XML documents, which may be called distributed trees, are documents which have been partitioned and sent to various nodes and are linked together to form a complete XML document (Abiteboul, Gottob, & Manna, 2008). The process can be accomplished through embedded function calls to the separate documents over a network from within a centralized node in the distributed system (Abiteboul et al., 2008). Previous researches have focused on distributing XML documents based on data size, others are based on data structure. The most efficient distributed systems consider both data size and structure. The system discribed in (Seyed-Abbassi & Gordon, 2015; Aljawarneh, 2011) distributes XML documents in a cloud services system using a kernel document and many distributed cloud nodes. Figure 1 discribes the system which distributes the XML document through an algorithm of least load (Seyed-Abbassi & Gordon, 2015) based on the number of cloud servers available and the size of the original XML document and preserving the tree structure of the XML document. The algorithm splits each of the subtrees into parts based on which distribution node has the most space still available (Seyed-Abbassi & Gordon, 2015) and once the load is determined, it stores each subtree in the corresponding cloud. The kernel document indicates which cloud service holds which partitionned document. For the backup the system stores the same partitions with the same loads into a second cloud service, this redundancy allows to find the lost data in case a node is unavailable due to electrical issues, being compromised, hacking, or any other problem (Seyed-Abbassi & Gordon, 2015; Aljawarneh et al., 2015).

Other studies have focused on enhancing queries processing on large data volumes. The system designed in (Kurita, Hatano, Miyazaki, & Uemura, 2007) splits XML data and distributes it evenly...
Efficiency Measurement in: Branch Bank Service with Data Envelopment Analysis
www.igi-global.com/article/efficiency-measurement-branch-bank-service/65744?camid=4v1a

Cloud Computing Deployment and Selection Criteria for Organizations
www.igi-global.com/article/cloud-computing-deployment-and-selection-criteria-for-organizations/105506?camid=4v1a