Query Processing and Index Structure on Cloud Peer-to-Peer OLAP

Nam Hun Park\textsuperscript{1} and Kil Hong Joo\textsuperscript{2}\textsuperscript{*}

\textsuperscript{1} Dept. of Computer Science, Anyang University, 102 Samsungli, Buleunmyun, Ganghwagun, Incheon, Korea, 417-833
nmhnpark@anyang.ac.kr

\textsuperscript{2} Dept. of Computer Education, Gyeongin National University of Education, San 6-8 Seoksudong Manangu Anyangsi, Gyeonggi, Korea, 430-040
khjoo@ginue.ac.kr

Abstract. The conventional researches on a distributed On-Line Analytical Processing (OLAP) system have been in hardship to be adapted to real business environment. In this paper, to provide not only flexibility and expandability of ROLAP but also the speediness of MOLAP, the cloud server architecture is proposed which shares clients’ cube cache by P2P and manages central index on cube data on P2P nodes. The requests on server are confined to non-aggregated areas and multiple client nodes exchange data simultaneously and asynchronously in the proposed Cloud P2P OLAP. With time series properties, the volume of requested data is minimized and the reuse of past cache data is focused. While Grid OLAP should keep self-distributed system, Cloud use services as much as used by public services. Therefore, Cloud P2P OLAP solves a lot of theoretical limits of conventional distributed OLAP.

Keywords: Cloud Computing, OLAP

1 Introduction

The conventional researches on a distributed On-Line Analytical Processing (OLAP) system have been in hardship to be adapted to real business environment. However, the recent spread of Cloud PaaS(Platform as a Services) provides new chances in the field of a distributed OLAP. OLAP query execution costs many minutes by its enormous data and OLAP query properties. On the other hand, MOLAP has fast responses. But it has a physical space limit to materialize all cells in possible combinations. Therefore, MOLAP is unsuited to analyze large data.

In this paper, each node acts as not only a P2P duplicator of aggregated data cubes but also a hybrid server to process queries on sub-cubes. Also, the requests on server are confined to non-aggregated areas and multiple client nodes exchange data simultaneously and asynchronously in the proposed Cloud P2P OLAP. In particular, data are retrieved from physically or logically adjacent nodes. With time series properties, the volume of requested data is minimized and the reuse of past cache data is focused.

\textsuperscript{*} Corresponding Author
2 Related Works

Most commercial OLAP systems provide MOLAP and ROLAP simultaneously and also provide hybrid OLAP which has MOLAP and ROLAP properties together. But, MOLAP requires too large storage space to materialize all combinations with millions of records, ROLAP is universally used by major companies. Recently, the extension of OLAP to a distributed, grid and P2P environment is a challenging issue. Also, to make ROLAP with quick responses like MOLAP and to solve astronomical space allocation problems of MOLAP, a new concept of storing result cells from ROLAP into clients or distributed nodes of grid systems receives attentions. Recently, Cloud Computing grafted to OLAP[2] is proposed. Because Cloud system includes OLAP data storage, [2] doesn’t have advantages in performance and cost efficiency.

However, except large portals like google, yahoo, naver, etc., with private cloud, the advantages from economies of scale is hard to be utilized. This is not a proper approach for OLAP on companies. Recent researches of Hive and Thusoo[3] propose data warehouse solutions to manage time series data of petabytes with hadoop. Also, infrastructure only for analysis on specified items of mass data has been proposed.

To adapt cloud systems to database systems, lots of problems have to be solved[4]. The database and mass storage service on cloud under virtual systems have a drawback of using large network costs when input-output cost passes over available zones.

3 Cloud OLAP Architecture

This paper proposes three index layers with Cloud P2P OLAP. The central cloud index server is composed by two layers and the client-server which works as a client and also a server is composed by one index layer. By using multiple layers together, difference performance improvement factors are considered together and the depth of layers on the hybrid P2P is expanded when further costs arise. In this paper, the multi-layer(N levels) hybrid P2P is proposed. N means thee level layers are able to be expanded by the size of indexes. When loads are on index servers, the first layer and the second layer can be expanded physically like a tree structure. The second layer is much larger than the first layer, one instance on the first layer is connected to M instances just like a tree. After the depth of the index is increased, the horizontal expansion to the maximum number of available star schemes is feasible.

In this paper, two-dimensional adjacency information is proposed to represent division similarity information together with one-dimensional physical information. The physical adjacency is important considering network transfer efficiency, and the logical adjacency is important to find existing data in a similar department and to reuse data afterward. For an index structure to show two-dimensional information, R-tree, KD-tree, etc. are used. However, to adapt time-series indexes together, a quadtree index is proposed in this paper. After determining the adjacency, nodes within the queried time-series are examined from the most adjacent nodes by expanded recursive calls. If the balance mechanism works, the time-series information at each depth are summarized and used to reconstruct the entire time-series tree. In this case, the quadtree is very advantageous. However, the balance is not guaranteed, so the inquiry
performance can be a drawback. In this paper, to overcome the unbalanced quadtree drawbacks, the preprocessing algorithm is additionally proposed to reconstruct the index. When OLAP updates the materialized view at nighttime periodically, Cloud P2P OLAP also reconstructs the index to maintain day-unit balance. When the adjacent index is considered, the entire bit of the time-series should be examined. Figure 1 shows this case. The time complexity in Figure 1 does not guarantee the good performance. To make up for the weak point, the summary of the entire time-series index are centrally managed by bit or sum indexes. The detailed index with each bit is managed like linked-lists.

4 Experiments

The proposed system is implemented on .NET Framework 3.5 with C#. The SOA service communication protocols in P2P and Cloud are developed with WCF and SOAP protocols. The 4 dimensional data set with 2,930,459 row is used to analyze the performance.

Figure 3 shows the performance comparison between the proposed method and the general OLAP system. When queried results are asked directly to a server, the response time is over 8sec. Query scenarios are executed 100 times. In the worst case, the response time is similar to the general OLAP. In the best case, every query has responses within 1sec. The Cloud P2P OLAP is proven to have much faster responses and much less costs.
5 Conclusions

This paper proposed Cloud P2P OLAP, which dynamically adds/removes instances to maximize expandability and usability. The architecture including cloud systems and P2P together are newly proposed and proved to have better performance than general OLAP systems by experiments.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science Technology(NRF-2011-0025300).

References