

BESTPEER++: A PEER-TO-PEER BASED DATA SHARING USING LARGE SCALE

ASIF VB

ME CSE, Maharaja Prithvi Engineering College, Avinashi
asifbvaya@gmail.com

Abstract— The corporate network is often used for sharing information among the participating companies and facilitating collaboration in a certain industry sector where companies share a common interest. It can effectively help the companies to reduce their operational costs and increase the revenues. However, the inter-company data sharing and processing poses unique challenges to such a data management system including scalability, performance, throughput, and security. In this paper, we present Best Peer++, a system which delivers elastic data sharing services for corporate network applications in the cloud based on Best Peer—a peer-to-peer (P2P) based data management platform. By integrating cloud computing, database, and P2P technologies into one system, Best Peer++ provides an economical, flexible and scalable platform for corporate network applications and delivers data sharing services to participants based on the widely accepted pay-as-you-go business model. We evaluate Best Peer++ on Amazon EC2 Cloud platform. The benchmarking results show that Best Peer++ outperforms HadoopDB, a recently proposed large-scale data processing system, in performance when both systems are employed to handle typical corporate network workloads. The benchmarking results also demonstrate that Best Peer++ achieves near linear scalability for throughput with respect to the number of peer nodes.

I. INTRODUCTION

The data mining process is to extract information from a data set and transform it into an understandable structure for further use. A side from the raw analysis step, it involves database and data management aspects, data pre-processing, model and inferences considerations, interestingness metrics complexity considerations, post-processing of discovered structures, visualization, and online updating. The term is a misnomer, because the goal is the extraction of patterns and knowledge from large amount of data, not the extraction of data itself. The actual data mining task is the automatic or semi-automatic analysis of large quantities of data to extract previously unknown interesting patterns such as groups of data records, unusual records and dependencies. This usually involves using database techniques such as spatial indices. These patterns can then be seen as a kind of summary of the input data, and may be used in further analysis or, for example, in machine learning and predictive analytics.

From a technical perspective, the key for the success of a corporate network is choosing the right data sharing platform, a system which enables the shared data (stored and maintained by different companies) network-wide visible and supports efficient analytical queries over those data. Traditionally, data sharing is achieved by building a centralized data warehouse, which periodically extracts data from the internal production systems (e.g., ERP) of each company for subsequent querying. Unfortunately, such a warehousing solution has some deficiencies in real deployment. A system which enables the shared data network-wide visible and supports efficient analytical queries over those data. which periodically extracts data from the internal production systems of each company for subsequent querying. such a warehousing solution has some deficiencies in real deployment. The Best peer++ unique challenges posed by sharing and processing data in an environment and it delivers elastic data sharing services. Best Peer++ employs a hybrid design for achieving high performance query processing.

A. Dataset Collection

Most commonly a data set corresponds to the contents of a single database table, or a single statistical data matrix, where every column of the table represents a particular variable, and each row corresponds to a given member of the data set in question. The data set lists values for each of the variables, such as height and weight of an object, for each member of the data set. Each value is known as a datum. The data set may comprise data for one or more members, corresponding to the number of rows. The TPC-H benchmark data set consist of eight tables. Here, use the original TPC-H schema as the shared global schema. The TPC-H data sets, implement the benchmark queries by ourselves since the TPC-H queries are simple and time-consuming queries. In the online data flow, user queries are submitted to the normal peer and then processed by the query processor. The query processor performs user queries using a fetch and process strategy.

Schema mapping is a component that defines the mapping between the local schema of each production system and the global shared schema employed by the corporate network. Currently, Best Peer++ only supports relational schema mapping, namely both local schema and the global schema are relational. The mapping consists of metadata mappings (i.e., mapping local

table definitions to global table definitions) and value mappings (i. e., mapping local terms to global terms). Besides schema level mapping, Best Peer++ can also support instance level mapping [19], which complements the mapping process when there is not sufficient schema information. In general, the schema mapping process requires human to be involved and is rather time consuming. However, it only needs to perform once. Furthermore, Best Peer++ adopts templates to facilitate the mapping process. Specifically, for each popular production system (i.e., SAP or PeopleSoft), we provide a mapping template which defines the transformation of local schemas of those systems to a global schema.

Data Loader is a component that extracts data from production systems to normal peer instances according to the result of schema mapping. While the process of extracting and transforming data is straightforward, the main challenge comes from maintaining consistency between raw data stored in the production systems and extracted data stored in the normal peer instance (and subsequently data indices created from these extracted data) while the raw data being updated inside the production systems.

In the Best Peer++, the data are stored in the local MySQL database hosted by each normal peer. Thus, to process a query, we need to locate which normal peers host the tables involved in the query. For example, to process a simple query like select R.a from R where R.b=x, we need to know the location of the peers store tuples belonging to the global table R. We adopt the peer-to-peer technology to solve the data locating problem and only send queries to normal peers which host related data. In particular, we employ BATON ,a balanced binary tree overlay protocol to organize all normal peers.

B. Preprocessing Method

Data preparation and filtering steps can take considerable amount of processing time. Includes cleaning, normalization, transformation, feature extraction and selection, etc. Analyzing data that has not been carefully screened for such problems can produce misleading results. Thus, the representation and quality of data is first and foremost before running an analysis. If there is much irrelevant and redundant information present or noisy and unreliable data, then knowledge discovery during the training phase is more difficult. Data preprocessing describes any type of processing procedures that are performed on raw data to prepare it for another processing procedure and ultimately, analysis. Preprocessing linearizes the relationships among the variables in your Dataset and removes extraneous sources of variation that are of no interest to the analysis..

C. Query Processing

The cost of both P2P and map reduce engines are predicted based an histograms and runtime parameters of the cost model. The query planner compares the cost between the two methods and executes the one with lower cost. When the query is submitted the query planner retrieves related histogram and index information from the bootstrap node, analyze the query and construct a processing graph of a query. Comparing between two cost models, observe the table size and query complexity are the key factors that affect the query planner decision. Best Peer++ employs two query processing approaches: basic processing and adaptive processing. The basic query processing strategy is similar to the one adopted in the distributed databases domain. Overall, the query submitted to a normal peer P is evaluated in two steps: fetching and processing. In the fetching step, the query is decomposed into a set of sub queries which are then sent to the remote normal peers that host the data involved in the query (the list of these normal peers is determined by searching the indices stored in BATON, cf. Section 4.3).The sub query is then processed by each remote normal peer and the intermediate results are shuffled to the query submitting peer P.In the processing step, the normal peer P first collects all the required data from the other participating normal peers

D. Auto Fail-Over and Auto Scaling

If some peers are malfunctioned or crashed the bootstrap peer will be automatic fail over event for each failed normal peer. The bootstrap peer triggers an auto-scaling event to either upgrade the normal peer to a larger instance or allocate more storage spaces. its data can be safely recovered in cases of crashes within a recovery time constraint guaranteed by the service level agreements (SLAs) offered by the cloud services. In a data sharing platform like Best- Peer++, enforcing system's consistency guarantee is a crucial but difficult task. An important issue is the consistency of the whole system when there are node failures, more specifically how queries can be executed in these situations. Each normal peer intends to join an existing corporate network must first connect to the bootstrap peer. If the join request is permitted by the service provider, the bootstrap peer will put the newly joined peer into the peer list of the corporate network. At the same time, the joined peer will receive the corporate network information including the current participants, global schema, role definitions, and an issued certificate. When a normal peer needs to leave the network, it also notifies the bootstrap peer first. The bootstrap peer will move the departure peer to the black list and mark the certificate of the departing peer invalid. The bootstrap peer will the reclaim all resources allocated to the departing peer and finally remove the departing peer from the peer list.

The bootstrap periodically collects performance metrics of each normal peer (line 2). If some peers are malfunctioned or crashed, the bootstrap peer will trigger an automatic fail-over event for each failed normal peer (line 6-10). The automatic fail-over is performed by first launching a new instance from cloud. Then, the bootstrap peer asks the newly launched instance to perform database recovery from the latest database backup stored in Amazon EBS. Finally, the failed peer is put into the blacklist. Similarly, if any normal peer is overloaded (e.g., CPU is over-utilized or free storage space is low), the bootstrap peer triggers an auto-scaling event to either upgrade the normal peer to a larger instance or allocate more storage spaces.

ALGORITHM

Input: Query Q

Output: Query configuration on a specific query engine

```
TableSet S ← TableParser(Q);
Cost Cmin ← MAX_VALUE;
QueryPlan Target ← null;
QueryPlanSet QS ← ∅;
Foreach Table T ∈ S do
{
  GraphSet GS = GraphGen(T);
}
Foreach Graph G ∈ GS do
{
  QueryPlan P1 = P2PlanGen(G);
  QueryPlan P2 = MapredPlanGen(G);
  QS = QS ∪ {P1};
  QS = QS ∪ {P2};
}
Foreach QueryPlan P ∈ QS do
  If CostEst(P) < Cmin then
  Cmin = CostEst(P);
  Target = P;
return Target;
```

The system also adopts two additional optimizations to speed up the query processing. First, each normal peer caches sufficient table index, column index, and range index entries in memory to speed up the search for data owner peers, instead of traversing the BATON structure. Second, for equi-join queries, the system employs bloom join algorithm to reduce the volume of data transmitted through the network. For small jobs, the P2P engine performs better than the Map Reduce engine, as it does not incur initialization cost and database join algorithms have been well optimized. However, for large-scale data analytic jobs, the Map Reduce engine is more scalable, as it does not incur expensive data replications. Based on the above-mentioned cost models, we propose our adaptive query processing approach. When a query is submitted, the query planner retrieves related histogram and index information from the bootstrap node, analyzes the query and constructs a processing graph for the query. Then the costs of both the P2P engine and Map Reduce engine are predicted based on the histograms and runtime parameters of the cost models. The query planner compares the costs between two methods and executes the one with lower cost.

E. Data Sharing

The corporate network is often used for sharing information among the participating companies and facilitating collaboration in a certain industry sector where companies share a common interest. Data sharing and processing poses unique challenges to such a data management system including scalability, performance, throughput, and security. In order to encourage data sharing and prevent the loss or corruption of data, a number of funding agencies and journals established policies on data archiving. Access to publicly archived data is a recent development in the history of science made possible by technological advances in communications and technology. The ability to share the same data resource with multiple applications or users. It implies that the data are stored in one or more servers in the network and that there is some software locking mechanism that prevents the same set of data from being changed by two people at the same time.

The access to multi-businesses data shared in a corporate network needs to be controlled properly. The challenge is for Best Peer++ to provide a flexible and easy-to-use access control scheme for the whole system; at the same time, it should enable each business to decide the users that can access its shared data in the inherent distributed environment of corporate networks. Best Peer++ develops a distributed role-based access control scheme. The basic idea is to use roles as templates to capture common data access privileges and allow businesses to override these privileges to meet their specific needs. Note that Best Peer++ does not collect the information of existing users in the collaborating ERP databases, since it will lead to potential security issues. Instead, the user management module of Best Peer++ provides interfaces for the local administrator at each participating organization to create new accounts for users who desire to access Best Peer++ service. The information of the users created at one peer is forwarded to the bootstrap peer and then broadcasted to other normal peers also. In this manner, each normal peer will eventually have enough user information of the whole network.

II. CONCLUSION

Best peer++, a system which delivers elastic data sharing services by integrating cloud computing, database, and peer-to-peer technologies. And it has an efficient data to sharing different nodes and it effectively handle typical workloads in a corporate network. A system which enables the shared data network-wide visible and supports efficient analytical queries over those data. which periodically extracts data from the internal production systems of each company for subsequent querying. such a warehousing solution has some deficiencies in real deployment. The proposed system is used to rectify the existing problems, and also it reduced the data lost. It easy to access and easily understanding the concept. Everyone to be accessed in the Best Peer++. Then, it is scalable, Flexible and easy to sharing without any data lost. A Best peer++, a system which delivers elastic data sharing services by integrating cloud computing, database, and peer-to-peer technologies. And it has an efficient data to sharing different nodes and it effectively handle typical workloads in a corporate network. at the same time, it should enable each business to decide the users that can access its shared data in the inherent distributed environment of corporate networks. In its early stage, Best Peer employs unstructured network and information retrieval technique to match columns of different tables automatically. After defining the mapping functions, queries can be sent to different nodes for processing. The main concern is the possibly high maintenance cost.

References

- [1] Gang Chen, Tianlei Hu, Dawei Jiang, Peng Lu, Kian-Lee Tan, Hoang Tam Vo, and Sai Wu BestPeer++: A Peer-to-Peer Based Large-Scale Data Processing Platform.
- [2] H.V. Jagadish, B.C. Ooi, K.-L. Tan, O.H. Vu, and R. Zhang. "Speeding up Search in Peer-to-Peer Networks with a Multi-Way Tree Structure," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2006
- [3] H.V. Jagadish, B.C. Ooi, K.-L. Tan, C. Yu, and R. Zhang. "iDistance: An Adaptive B+-Tree Based Indexing Method for Nearest Neighbor Search," ACM Trans. Database Systems, vol. 30, pp. 364-397, June 2005.
- [4] E. Rahm and P. Bernstein. "A Survey of Approaches to Automatic Schema Matching." The VLDB J., vol. 10, no. 4, pp. 334-350, 2001. 1330 IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 6, JUNE 2014
- [5] P. Rodriguez-Gianolli, M. Garzetti, L. Jiang, A. Kementsietsidis, I. Kiringa, M. Masud, R.J. Miller, and J. Mylopoulos, "Data Sharing in the Hyperion Peer Database System," Proc. Int'l Conf. Very Large Data Bases, pp. 1291-1294, 2005.
- [6] A. Thusoo, J. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthonv, H. Liu, P. Wvckoff, and R. Murthy, "HIVE: A Warehousing Solution over a Map-Reduce Framework," Proc. VLDB Endowment, vol. 2, no. 2, pp. 1626-1629, 2009.
- [7] H.T. Vo, C. Chen, and B.C. Ooi, "Towards Elastic Transactional Cloud Storage with Range Query Support," Proc. VLDB Endowment, vol. 3, no. 1, pp. 506-517, 2010.