

A Row-column Hybrid Storage Approach based on Massively Parallel Processing

Qiao Sun, Lan-Mei Fu, Peng Wu

Beijing Guodiantong Network Technology Co., Ltd.
Beijing, 100070, China
Email: littlesmart@yeah.net

Jia-Song Sun

E. E. Department, Tsinghua University
Beijing, 100084, China
Email: littlesmart@yeah.net

Abstract-Massively parallel processing architecture, as a common parallel processing server architecture, is widely used in the field of mathematical modeling and database processing which require a lot of computing. This architecture can complete the calls of multiple processors by the coordination, which can process the same command simultaneously. However, this architecture is unable to satisfy the needs of optimizing and time delay when deal with the relational database system of row or column storage. This paper presents a row-column hybrid storage approach based on massively parallel processing, which selects the method by judging the requirements of the pending data. The optimized row storage method is selected with the low time-delay requirement and the optimized column storage method is selected with the complex processing requirement. And with the help of cross-layer during query and processing, both on-line analytical processing and on-line transaction processing can be achieved in one system. The real experiments on massively parallel processing network architecture show that this method can achieve the balance between high throughput and complex processing, and improve the overall performance.

Keywords-massively parallel processing; row-column hybrid Storage; cross-layer query; relational database.

I. INTRODUCTION

Distributed database is physically distributed in the computer network on a number of logically interrelated database collection, the geographical location of the management and control need to different degrees of concentration of multiple centralized database together to form a unified database. At present, distributed research mainly has two directions: one is the non-relational database of heyday development, and the other is relational data of parallel processing. For the database system, the main motivation is to put all the business data together to provide a centralized access to data access; for computer network technology, the data will be decentralized processing, is the opposite operation. Whether it is non-relational distributed database, or parallel architecture based on relational distributed database, are trying to combine both of them [1] [2].

Parallel Processing is a computational method in a computer system that can execute two or more processes at the same time. Parallel processing can work on different aspects of the same program at the same time. The main purpose of parallel processing is to save large and complex problem solving time. In order to use parallel processing, we first need to parallelize the program, which means distribute

working parts into different processes (threads) [3]. As a kind of parallel processing server architecture, Massively Parallel Processing is commonly used in mathematical modeling, database processing and other fields of large computational complexity. It can accomplish the call of multiple processors through coordination at the same time. In SQL and other relational database systems, the storage model is divided into two types of row storage and column storage, each with its advantages and disadvantages, the use of one cannot meet the needs of the simultaneous optimization of reading and writing, so the need to study the ranks of hybrid storage the way.

In this paper, we propose a hybrid layout storage method and query method for the MPP architecture. We judge the demand of the data to be processed and adopt the optimized row storage method under the demand of low latency. Under the requirement of complex processing, we adopt optimized column storage and cross-layer approach in the query and processing, so only need to deploy a set of products can be achieved online analytical processing and online transaction processing. Experiments in the real MPP network architecture show that this approach can achieve a balance between high throughput and complex processing and improve the performance of the overall MPP architecture.

The remainder of this paper is organized as follows. First, we briefly introduce the massive parallel processing architecture. In section 2, we review the method of row and column storage. In section 3, we propose a hybrid storage and implementation method based on massive parallel processing, and prove this method by experiment and evaluation. The effectiveness of the final summary and outlook.

II. INTRODUCTION TO MPP ARCHITECTURE

From the view of system architecture, the current server architecture can be divided into the following three types: Symmetric Multi-Processor (SMP), non-uniform memory access structure (NUMA), Massive Parallel Processing (MPP). Figure 1, Figure 2 and Figure 3 are the basic structure of these three schematics [4].

A typical architecture for parallel processing is a symmetric multiprocessing (SMP) architecture, where a set of processors (multiple CPUs) are pooled on a single computer and the memory subsystem and the bus architecture are shared among CPUs. In this architecture, a computer is no longer composed of a single CPU, while the operating system by a number of processors running a single copy, and shared memory and other resources of a computer.

Although the use of multiple CPU, but from a governance point of view, their performance is like a single machine. The system distributes task queues symmetrically over multiple CPUs, which greatly improves the data processing capability of the whole system. All processors have equal access to memory, I/O, and external interrupts. In a symmetric multiprocessing system, system resources are shared by all CPUs in the system, and workloads can be distributed evenly over all available processors.

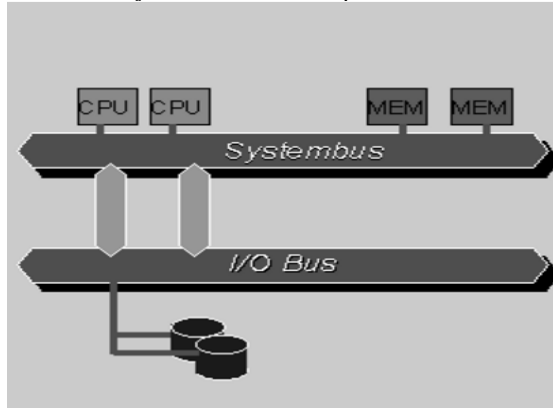


Figure 1. Basic structure diagram of symmetric multi-processing.

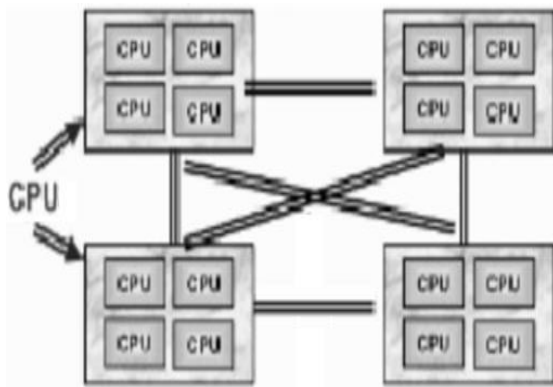


Figure 2. Basic structure diagram of non-uniform storage.

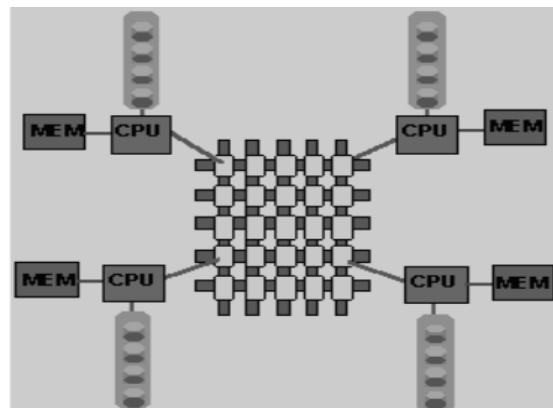


Figure 3. Basic structure diagram of massive parallel processing.

SMP is limited in scalability, NUMA technology can be effectively extended to build large-scale systems. NUMA can be dozens of CPU or even hundreds of CPU combination in a server. NUMA is characterized by a number of CPU modules, each CPU module consists of multiple CPU, and has a separate local memory, I/O slots and so on. Because the nodes can connect and exchange information through the interconnected modules, each CPU can access the memory of the whole system, which is the important difference between NUMA system and MPP system. Obviously, accessing local memory will be much faster than accessing memory in remote memory or other nodes within the system, which is the origin of non-uniform storage access NUMA. Because of this feature, in order to better play system performance, the development of applications need to minimize the information exchange between different CPU modules. NUMA can solve the original SMP system expansion problem, but NUMA technology also has some defects, because access to remote memory far exceeds the local memory, so when the CPU number increases, the system performance cannot increase linearly.

Massively Parallel Processing is commonly used in mathematical modeling, database processing, and other fields, MPP through the coordination to complete a number of processor calls, the same command can be processed at the same time, and its main feature is multiple processors simultaneously perform computations in parallel, with real-time communication between the processors. Through the network real MPP interface and its management, can under the same load, with a number of price advantage of the server to parallel processing to replace the traditional server. In contrast, the traditional server architecture SMP mode [5] will be a number of processors and a centralized memory connected to the server under the SMP structure usually contains eight processors, the reason is that when the storage interface Saturated, the increase in processor cannot get a greater advantage.

In addition, there are many tightly coupled multi-processors in SMP, in which all CPUs share all resources, such as bus, memory, and I/O systems. There is only one copy of the operating system or the administrative database. One of the biggest features is the sharing of all resources. MPP is composed of many loosely coupled processing units, it should be noted that here refers to the processing unit rather than the processor. Each unit of the CPU has its own private resources, such as bus, memory, hard drives and so on. There is an instance copy of the operating system and administration database in each cell. The biggest feature of this structure is not shared resources.

III. ROW AND COLUMN HYBRID STORAGE METHOD

Relational database system storage mode is divided into two types of row storage and column storage [6]. The data in the row storage mode is stored in tuples. All attributes of each tuple are stored together. If you want to query a tuple for an attribute value, you need to read the entire tuple first. Most current relational database systems use row-storage technology. Column stores organize each column of a table

together for storage, and different columns are stored independently. The logical structure of row and column storage is shown in Figure 4 and Figure 5.

Row storage and column storage technology has its own advantages and disadvantages, such as the former data stored in accordance with the tuple, write operation is highly efficient; the latter data tuple must first split into separate attribute columns for storage, write operation Efficiency is low. In addition, the query process generally involves only a few properties, the former must read the entire record, when the property is more costly to read data; the latter only need to read the property column. And when using column storage, you can get a higher compression rate, and row storage is not conducive to compression.



Figure 4. Row storage logical structure.

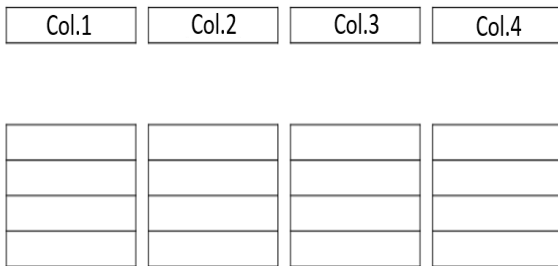


Figure 5. Column storage logical structure.

Row storage and column storage are the mainstream methods of data storage. However, with the introduction of emerging business data such as Internet of Things and mobile Internet, the bottleneck of the traditional method is gradually appeared in the following aspects [7]:

Storage technology single, the existence of short-board technology: physical data can only be selected by row storage or column storage, but the two programs are some technical limitations, cannot take into account the advantages of row and column storage.

- Storage space waste is serious: Mobile Internet, Internet of Things business, including not only dense data, there are massive sparse data, such data takes up a huge storage space, and the current data warehouse is a row storage structure data compression ratio is relatively low, Easy to cause a large storage space waste.
- Some business data storage scheme is not reasonable, the flexibility is poor: business analysis system of large amount of data entities, a large number of attributes, and the characteristics of different characteristics between the columns, and some columns of data dense, high importance for the row to store some column data Sparse, low importance, suitable for column storage.

But are unified by the current line of storage, not according to the characteristics of the data characteristics of the column targeted storage options, the problem in the mobile Internet, Internet of things and other emerging business data is particularly evident.

- Cannot realize the mixture of the sparse data and the dense data, the high-efficient storage: More sparse data of the new business class data introduced by the sub-system, compared with the traditional dense data, sparse data and the model is changed frequently. In practical application, the two types of data usually coexist, and neither the relational model nor the non-relational model can effectively solve the problem of efficient storage of "hybrid data set".
- Investment costs and business support costs remain high: Column storage for online analytical processing, line storage for online transactions, but for online analytical processing and online transaction processing products are independent of each other, the need to deploy two or more sets Products to meet business needs, not only bring higher investment costs, and the actual application needs to do data migration between different products, bringing additional management and maintenance costs.

In fact, the database system based on row storage technology and column storage technology can only satisfy one of read optimization and write optimization, and cannot solve the problem of simultaneous optimization of reading and writing. Therefore, the database system based on hybrid storage is necessary.

IV. MPP ROW-COLUMN HYBRID STORAGE APPROACH

The main difficulty in implementing row and column hybrid storage on traditional row data is that the two storage technologies have different methods for organizing documents, so their query execution methods are very different and difficult to unify. A common practice is to use separate DBMSs for analysis queries and transactions. In this mode, the new data enters the on-line transactional processing (OLTP) database management system and then uses the background extraction/ transformation/loading tool to migrate the data from OLTP to the data warehouse for On-line analytical processing (OLAP). The ideal situation is to support the system's high throughput and low latency OLTP requirements, and can be in the transaction data and historical data between the more complex OLAP queries [8]. But the difficulty is that the delay between the two systems to transfer information from a few minutes to several hours, so the application of real-time response type is more difficult to achieve; query processing and storage engine is stored in different layers, that OLTP is Line storage, OLAP is generally stored in the column, the need to submit two-phase methods such as query operations; In addition to the two systems to query each other requires program developers and managers to invest more time.

Student ID	Student Num.	Student Name	Student Address
1	20110788	Zhang San	Dormitory_1
2	20110789	Li Si	Dormitory_2
3	20110790	Liu Xiao	Dormitory_3
4	20110791	Zheng Li	Dormitory_4
5	20110792	Wu Le	Dormitory_5

Figure 6. Examples of row and column hybrid storage.

In this paper, we propose a hybrid layout storage method and query method for the MPP architecture. We judge the demand of the data to be processed and adopt the optimized row storage method under the demand of low latency. Under the demand of complex processing, we adopt optimized column storage and cross-layer approach in the query and processing, so only need to deploy a set of products can be achieved online analytical processing and online transaction processing.

In this method, the metadata storage for the cross-layer hybrid storage, in the database or table to be frequently accessed data using line storage, and other data using column storage. Student information database, for example, the student's name and name is the most frequently queried and access data, and student address and so requires more complex queries, so the former data in accordance with the format of the row store, the latter as the storage Form, as shown in Figure 6.

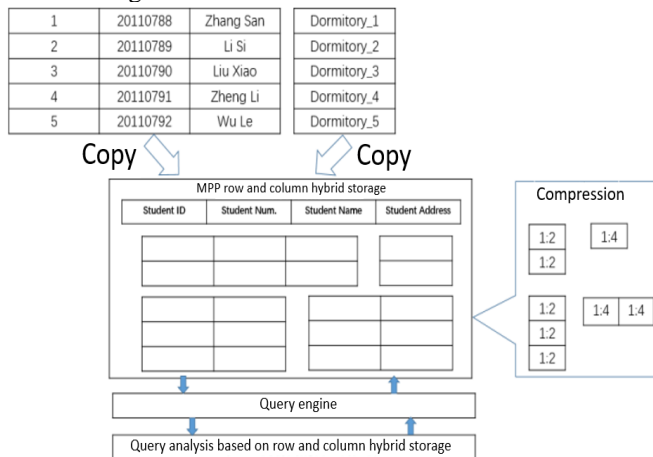


Figure 7. Cross-layer row and column hybrid storage.

As shown in Figure 7, this paper proposes a MPP-based cross-layer hybrid storage architecture, which first receives and replicates data, and then performs a column-and-column hybrid query analysis based on MPP. The steps are as follows: receive query->word/parse->generate syntax tree->quadratic analysis->generate logic plan->query optimization->generate physical execution plan. Query engine part

mainly includes query execution and memory storage engine, where the storage engine includes column storage engine and line storage engine, respectively, corresponding to different storage query function; in the mixed storage part of the ranks, according to the query function on the low delay Or complex processing of the different needs of the data tables are stored in accordance with the different layers, each layer includes row storage and column storage format, respectively, with different compression ratio for quick query and return the results.

V. EXPERIMENTS AND RESULTS

In order to evaluate the above methods, the experiment was carried out based on the general laboratory network environment. Experimental configuration: 10 servers IBM-x3650M4-2U (2CPU, 6-core 12-thread Xeon E5-2620, 48G memory, 2T hard drive) for 7 * 24 hours, the operating system for Ubuntu11.10 and Windows7. Through the simple and complicated queries related to the grid service in the own database, this method and the integrated processing capability based on the row storage/column storage mode method are respectively used to obtain the average query processing time and judge the query processing capability per unit time. The test includes 10 database read operations, each database contains about 5000 records, simulation of 100 users in parallel access to the database, respectively, for these different database storage mode simple query and complex query test results Table 1, Table 2 and Table 3 below. From different query processing time, it can be seen that the row storage model has a lower query processing time advantage in simple query, and the column storage mode is superior to the row storage mode in complex query processing, but still remains high of the time interval. While the hybrid storage model is similar to the row storage model in terms of simple queries, it has a certain query processing time advantage over complex query processing and row storage and column storage, which indicates that the method has some advantages in improving query efficiency effect.

TABLE I. QUERY PROCESSING TIME (S) FOR MPP ROW STORAGE MODE

Simple Query	Different Threads			
	Thread 1	Thread 2	Thread 3	Thread 4
Query_1	2.09	1.48	1.12	2.57
Query_2	3.51	2.62	2.58	3.81
...
Query_10	3.50	3.29	3.52	3.77
Complex queries	Different Threads			
	Thread 1	Thread 2	Thread 3	Thread 4
Query_1	8.15	7.97	8.24	7.85
Query_2	10.29	9.05	10.21	9.43
...
Query_10	11.08	10.57	11.45	12.32

TABLE II. QUERY PROCESSING TIME (S) FOR MPP COLUMN STORAGE MODE

Simple Query	Different Threads			
	Thread 1	Thread 2	Thread 3	Thread 4
Query_1	2.09	1.48	1.12	2.57
Query_2	3.51	2.62	2.58	3.81
...
Query_10	3.50	3.29	3.52	3.77
Complex queries	Different Threads			
	Thread 1	Thread 2	Thread 3	Thread 4
Query_1	8.15	7.97	8.24	7.85
Query_2	10.29	9.05	10.21	9.43
...
Query_10	11.08	10.57	11.45	12.32

In addition, it is found in the experiment that, with the increase of query complexity, the hybrid storage based on MPP ranks has higher success rate than the general row storage or column storage, so it is considered that this method is more suitable for large Multi - concurrent Query Processing in the Peak of Task During Distributed Network System.

TABLE III. QUERY PROCESSING TIME (S) FOR MPP ROW AND COLUMN HYBRID STORAGE MODE

Simple Query	Different Threads			
	Thread 1	Thread 2	Thread 3	Thread 4
Query_1	2.45	1.71	1.05	2.78
Query_2	2.83	2.90	2.24	3.72
...
Query_10	3.08	3.49	3.16	3.85
Complex queries	Different Threads			
	Thread 1	Thread 2	Thread 3	Thread 4
Query_1	5.62	9.92	8.63	10.11
Query_2	5.25	6.14	8.28	5.44
...
Query_10	6.94	7.48	7.13	7.24

VI. CONCLUSIONS

As one kind of distributed database, massive parallel processing technology has begun to be applied to large data computing scenarios. In this paper, the MPP architecture for the ranks of hybrid storage and query methods, the purpose is to read and write efficiency and the balance between the compression ratios. The experiment in the intranet system shows that the method is better than the unmixed column storage and row storage model in the query processing time, and has certain advantages in the execution success rate. The next step will be to further improve the hybrid storage of multi-layer layout model, large-scale distributed database system in the grid to further improve query processing and execution efficiency.

ACKNOWLEDGMENT

This research was financially supported by Science and Technology Project of the State Grid Corporation of China (SGBJDK00KJJS1500180) and the State Grid Information & Telecommunication Group CO.,LTD. (SGITG-KJ-JSKF[2015]0010).

REFERENCES

- [1] Che Wujiang, "Study on Data Locking and Consistency of Distributed Database," Master's dissertation, Hunan: Hunan University of Science and Technology, 2011.
- [2] S. K. Das, D. J. Harvey, R. Biswas, "A latency-tolerant partitioner for distributed computing on the information power grid," Parallel and Distributed Processing Symposium., Proceedings 15th International, 2001:6.
- [3] https://en.wikipedia.org/wiki/Parallel_processing
- [4] <http://www.cnblogs.com/yubo/archive/2010/04/23/1718810.html>
- [5] <http://www.cnblogs.com/viviancc/articles/2310105.html>
- [6] SUN Lin-chao, CHEN Qun, XIAO Yu-ze, et al., "Study on the Hybrid Storage Database System," Application Research of Computers, 2013, 30 (2): 480-482.
- [7] An Yang, Zhao Hongsong, "Study and implementation of large data storage based on mixed-ranks hybrid storage," Communications Management and Technology, 2014 (1).