Design and Implementation of Microblog Database Based on NoSQL and Relational Database

Ruiying Xiong[1, a *], Jinya Xu[1,b] and Yanran Huang[1,c]

[1]Department of Information Management, Chengdu Neusoft University, Chengdu 611844, China

[a]XiongRuiying@nsu.edu.cn, [b]XuJinya@nsu.edu.cn, [c]HuangYanran@nsu.edu.cn

* The corresponding author

Keywords: Relational database; Non-relational database; NoSQL; Database design; Microblog

Abstract. In recent years, in massive data environment, such as large-scale data mining, business intelligence and social networking applications, relational database has not been able to meet the needs of emerging applications in scalability and concurrent access. On the basis of micro-blog business, the advantages of both the relational database and NoSQL are adopted to design and implement the micro-blog database. Micro-blog data using MySQL and Cassandra these two kinds of database products sub-treasury storage. The experiment shows that, compared with only using relational database for storage, using both MySQL and Cassandra for storage, can solve the storage problem of massive data in micro-blog with efficient access speed, which greatly improves the access efficiency of user requests under high load.

Introduction

At present, microblog as a new media, has become one of the hottest information dissemination tools, and its scale is gradually expanding. In recent years, the activity of the microblog existing user is obviously improved, and the number of registered users has also increased dramatically. According to the 2017 microblog user development report, as of September 2017, the monthly active users of microblog were 376 million, up to 27% compared with the same period in 2016, of which the mobile terminal accounted for 92%, and the daily active users reached 165 million, and increased by 25% over the same period of last year [1]. So far, Sina microblog's user group has successfully surpassed Twitter to become the largest independent social media company in the world. It can be seen that in daily life, microblog has become an indispensable source of information acquisition and the way of communication. However, the development of microblog has also brought new problems. The storage of massive data in microblog and massive users' High concurrent access are the urgent problems to be solved.

Traditional relational databases have some difficulties in storing massive amounts of data and accessing high concurrency, but if NoSQL database is used only, there is a problem of data consistency and security. Therefore, when designing database, we should fully consider the advantages of relational database and NoSQL database. Using the combination of relational database and NoSQL to design and implement micro-blog database is the best choice. Due to the combination of the NoSQL database and the relational database, the scholar Ramon Lawrence proposed a database storage pattern design scheme, which combines MySQL and MongoDB in his paper [2].The scholar Stumptner R,Lettner C,Freudenthaler B, a combined relational and NoSQL data processing approach is proposed to reduce data volume and work load of the relational part and enable the integral solution to process huge amounts of data in their paper [3].

Based on the existing research, this paper uses NoSQL and relational database to design and implement the microblog database. This can not only solve the storage of massive data in microblog, but also provide users with efficient access speed, and You can use the strong consistency and high security features of the relational database to meet the security requirements of users and information, and provide certain references for Sina Wei, Tencent, Netease, and Sohu Weibo.

Relational Database and NoSQL

Relational databases use a two-dimensional table structure to store data, which is defined as the relational model. In order to achieve massive data storage and high concurrent access, NoSQL database uses unstructured data model instead of relational model [4]. The main representative products of relational database are Oracle, SQL Server and MySQL. NoSQL (Not Only SQL) refers to non-relational databases, whose main products are Cassandra, MongoDB, CouchDB, and Redis. The main difference between them is as follows:

Data Table vs. Data Set: The main difference between relational databases and NoSQL is the way of data storage. Relational databases use tables to store data. Data in non-relational databases is usually stored in datasets. It is not suitable for storing in the rows and columns of data tables. The specific way of data storage can be selected according to the characteristics of the business itself, such as columns, key-value pairs, documents, and graph structures.

Predefined structure VS dynamic structure: Relational database must define the table and field structure before adding data while in NoSQL, data can be added anywhere and anytime, without the predefinition of tables. Moreover, it is very easy to adapt to changes in data types and structures.

Transaction vs. pure extensibility: Relational databases support fine-grained control over transaction atomicity and is easy of transaction rollback. If complex data queries require high trans actionality to control execution plans or data operations need high transactional application, the relational database is the best option in terms of its security, integrity and stability. Although NoSQL database can also perform transaction operations, but this is not its striking advantage. its real value is the scalability of large data processing and operations.

Data VS big data: Reliable storage of data and processing of data is the advantage of relational databases, and the biggest advantage of NoSQL is the processing of large amounts of unstructured data generated by our society or computers every day, that is, the processing of big data. And the scale-out potential of NoSQL databases is limitless because it uses modeless data management.

The Design of Microblog Database

The functions of microblog are numerous. Apart from basic functions such as sending Microblog and browsing Microblog, there are also collections, private messages, searches, and even functions such as shopping, listening to songs, traveling, and games. Here only select two major information for in-depth analysis, one is the most basic user information, including the relationship between the attention and concerns of users, the second is microblog information, including hot topics. Then according to the characteristics of the business, combined with the respective advantages and disadvantages of the relational database and NoSQL, select the most suitable database and database design.

The Design of Relational Database. In microblog applications, taking Sina Weibo as an example, the total number of users has reached more than 700 million, and the monthly active users reached 376 million. However, compared to the number of microblogs published by users, the amount of data corresponding to users is very small. It does not reach the level of massive data at all. In addition, the user's information needs to be protected by security. Therefore, this part of the data should be considered for selecting a relational database for storage. Then design a User table in the relational database, each user occupies a record in the user table to save the user information. The structure of the User table is shown in Table 1:

Table 1 Table structure of User table

Field Name	Ttype of Data	Constraint
UserID	char(30)	Primary key
UserName	varchar(20)	Not Null
RealName	varchar(8)	Not Null
Gender	char(2)	Not Null, The default value is male
RegisterDate	Date	Not Null, The default is the current date
Password	char(20)	Not Null

Table 2 Table structure of Attention table

Field Name	Ttype of Data	Constraint
AttentionID	Int	Auto_increment Primary key
Attend_UserID	char(30)	Not Null
Attended_UserID	char(30)	Not Null
Attend_Date	Date	Not Null, The default is the current date

In the process of using microblog, for interested users, we can pay attention to him and become his fans, so that we can see the dynamics of his publication. In the same way that others are interested in you, you can also pay attention to you and become your fan. Some microblog users will pay attention to thousands of other users. Some users, such as stars, universities, government agencies, and enterprises are usually concerned by thousands of users. Due to concern and attention between users have extremely strong transaction characteristics, so it is necessary to design a Attention table in a relational database to describe the concern and attention between users. The structure of Attention is shown in Table 2:

It is necessary to design a focus table in a relational database to describe the relationship between attention and attention of users. The structure of attention is shown in Table 2:

The Design of NoSQL Database. With the rapid development of microblog, the number of active users has increased dramatically, and a huge amount of microblog information has been generated every day. If you do not consider the NoSQL database and continue to select a relational database for processing, then the storage and access of data will face unprecedented challenges, even may cause web site embarrassment. Therefore, we consider the use of NoSQL database for storing massive amounts of data that are not highly transactional in microblog applications. Then a column family called Microblog is designed in the NoSQL database, and the relationship between microblog and users is reflected in the column family. Specifically, Using the microblogID name/value pair as a row key, Name/value pairs such as content and PublishTime are used as columns. In the column family design, the UserID and the UserName Name/value pairs are also used as the column data to represent the relationship between the microblog and the user, that is, to exchange access efficiency by sacrificing storage space. The column family structure of microblog is shown in Table 3.

Table 3 Column family structure of Microblog

Row Key	Column	Column	Column	Column
MicroblogId1	Content	PublishTime	UserID	UserName
	Value	Value	Value	Value
……	……	……	……	……
MicroblogIdn	Content	PublishTime	UserID	UserName
	Value	Value	Value	Value

In general, microblog users will send a lot of microblog on a hotspot, such as two sessions, house prices and so on. For the hot information involved in microblog, in order to improve the efficiency of the query, a column family named Hots is created to reflect the relationship between the hotspots and the microblog. Specifically, the Hots column family are divided by day, with the Name/value pairs of the HotName + date as the row key, and each microblog associated with the hotspot will add a column to record microblog information related to the hotspot, specifically the Name/value pairs of the MicroblogId and the Content form a super column. The column family structure of the Hots is shown in Table 4.

Table 4 Column family structure of Hots

Row Key	Super Column			Super Column	
HotName1+date	Column	Column	···	Column	Column
	MicroblogId1	Content	···	Microblog Idn	Content
	Value	Value	···	Value	Value
……	……	……	···	……	……
HotNamen+date	Super Column			Super Column	
	Column	Column	···	Column	Column
	Microblog Id1	Content	···	Microblog Idn	Content
	Value	Value	···	Value	Value

In the above, the users, users' attention, microblog, and hotspot search in microblog applications were analyzed in detail as examples. Then, based on the characteristics of the business, combining the advantages and disadvantages of both the relational database and the NoSQL database, a scheme for designing User and Attention two tables in a relational database and designing Microblog and Hots in the NoSQL database is given.

The Implementation of Microblog Database

Through the above design, microblog database can store the microblog data by a storage architecture combining MySQL and Cassandra. The specific implementation is divided into two parts:

The first part is the storage of user-related information. The data of this part is stored by using the relational database MySQL. Because the amount of the users’ corresponding data is not enough and users’ information needs security protection, the users’ data is selected and stored in a relational database. And MySQL, as a free relational database product, is so powerful to be the best choice. The specific operation is to use the CREATE DATABASE command in MySQL to create a database named MicroBlog, and then use the CREATE TABLE command to create the User and Attention tables in this database.

The second part is the storage of microblog-related information. The data of this part is stored by using the NoSQL database Cassandra [5]. Because the large amount of microblog data hasreached a large amount of data and the transactional requirements are not so high, themicroblog data selects NoSQL to store. Meanwhile, Cassandra database stores data in columns, which means its most important feature is to store the structured and semi-structured data, and to facilitate data compression. Therefore, it has very large IO advantages for a particular column or column of queries. The specific operation is to create a key space named MicroBlog in Cassandra using the CREATE KEYSPACE command, the key space is used to store the column family, like the database in the RDBMS, and then use the CREATE COLUMN FAMILY or CREATE TABLE command in the key space to create the Microblogs and Hots column families.

In order to store the user-related information in MySQL and micro-blog related information in Cassandra, it uses Memcached+MySQL+Cassandra storage structure. At first, the data is stored in the cache Memcached, and then the database is selected according to the type of data stored. At last, the use-relevant information is persisted to be stored in the My SQL database via JDBC, and the information about Microblog is persisted to be stored in the Cassandra database via Thrift. While accessing to the data, it first reads from the cache Memcached. If it hits, it returns the result; otherwise, it determines the type of the accessed data. The user-related information is queried by MySQL database through JDBC, and the information related to the microblog is queried by thrift for the Cassandra database. And then returns the query results and writes the query results to the cache Memcached.

Experimental results and analysis

Experimental data is collected through the Sina Weibo Open Platform API. The specific contents are as follows: 9,422 microblog records on the influenza, 17,192 records on the two sessions issues, 2,2012 microblog records on house price, and 46,418 microblog records on smog. In order to accurately test the query efficiency of MySQL and Cassandra, this experiment does not involve Memcached visit but directly performs database access. When the code is written, querying a microblog record of a hot spot in the MySQL database requires only a single table query in the microblog-hotspot table; the query in the Cassandra database only needs to query the Hot column family because the content of Weibo has been stored in the hotline family, which reflects the characteristics of Cassandra's efficiency at the cost of storage space.

After the above work was completed, open Weibo and use the MySQL database and the Cassandra database to search for influenza, two sessions, house prices, and smog. The experimental results are shown in Table 5. The experimental results are compared. The comparison is shown in Fig. 1.

Table 4 Hotspot corresponding microblog records and execution time

Hotspots	microblog record snumber	Use MySQL database execution time	Using Cassandra database execution time
influenza	9422	809.452423ms	733.346152ms
Two sessions	17192	1022.235612ms	911.131265ms
Housing price	22012	1180.142674ms	983.593522ms
Smog	46418	1842.256432ms	1124.328519ms

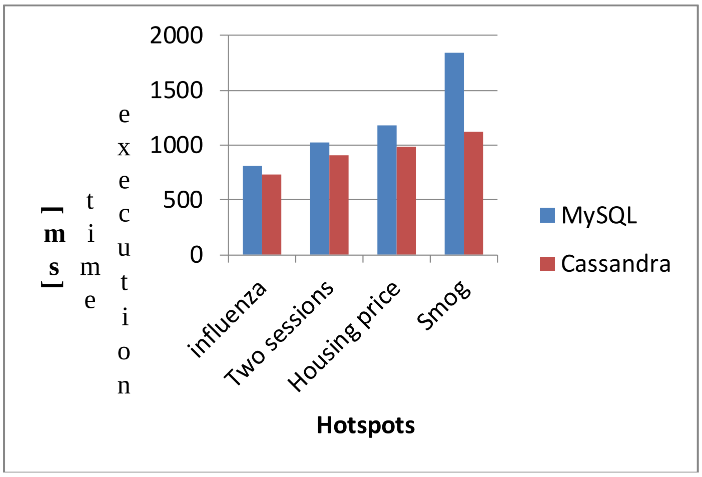

Figure 1. Finite MySQL and Cassandra query efficiency comparison chart

From Fig. 1, we can see that querying the same number of records, the Cassandra database takes less time than the MySQL database, indicating that the query efficiency of using Cassandra database to store data is better than using MySQL database. With the increase of the number of query records, the time difference between using the Cassandra database and the MySQL database is greater. This shows that from the perspective of query efficiency, NoSQL data Cassandra has more obvious advantages than relational database MySQL, especially as the amount of query data increases.

Conclusion

This paper analyzes the advantages and disadvantages of the relational database and NoSQL. Furthermore, based on the microblog's own business, it proposes a design scheme for the microblog data storage repository that combines the respective advantages of relational database and the NoSQL database. Experiments show that the query efficiency of using NoSQL databases to store large amounts of data is high, compared to the relational databases, and as the amount of data increases, compared to the use of relational database storage, the advantage of using NoSQL database becomes more obvious.

This paper proposes to use a relational database and non-relational database to store data in separate repositories to solve the storage of massive data and speed up the access rate, achieving the intended purpose. However, as to separate data repositories, different databases will be accessed in the process of data queries, which will increase the complexity of the application. In the future, a unified query model for relational and non-relational databases will be studied.

References

[1] Sina Weibo Data Center. 2017 Weibo User Development Report [R]. Micro report, 2017.Reference to a book (In Chinese).

[2] RamonLawrence: Proc.International Conferenceon Computational Science and Computational Intelligence,(Las Vegas, Nevada, USA, September 11-13, 2014) Vol. 1, p.285.

[3] R Stumptner, C Lettner and B reudenthaler: Computer Aided Systems Theory(EUROCAST, Germany,2015, p 663-670.

[4] H Y. Liao: Modern Computer, Vol. 43 (2017) No.3, p.115.

[5] S Anand, P Singh and B M. Sagar:Working with Cassandra Databas(Information and Decision Sciences, Australia 2018),.p.213.

[6] Peng L, Fang W. Heterogeneity of Inferring Reputation of Cooperative Behaviors for the Prisoners’ Dilemma Game [J]. Physica A: Statistical Mechanics and its Applications, 2015, 433: 367–378.