Research on Massive Enterprise Data One-Stop Search Mechanism Based on Information Grid

Shan-shan Wu
Science and Technology on Information Systems Engineering Laboratory,
Nanjing Research Institute of Electronics Engineering,
Nanjing, China
tulipwss@hotmail.com

Wei Jiang
Nanjing Research Institute of Electronics Engineering,
Nanjing, China
30352880@qq.com

***Abstract* —Massive enterprise data one-stop search system and method based on grid network is introduced in this paper. This paper presents the general metadata description model of heterogeneous data resource, the metadata extraction method of data resource, the service encapsulation method and registration publication mechanism of data resource, in order to standard the sharing process of large amounts of heterogeneous data resource. And then, the implement mechanism of data collection and index construction for huge amounts of data resource is given in the paper, by designing the optimized algorithm based on the map-reduce mechanism. Finally, the one-stop search mechanism is given in the paper in order to realize the data sharing and effective user of resources. All kinds of models or methods introduced in this paper, which could effectively handle the problem of data sharing and utilization brought by heterogeneous structures, enormous quantity, dispersive locations and complex contents of enterprise data under the environment of grid, are mainly used to build up unified cloud search framework of enterprise data.**

***Keywords*- Information Grid, One-stop Search, Massive Enterprise Data**

I. INTRODUCTION

In the information grid environment, enterprise data has the characteristics of heterogeneous structures, enormous quantity, dispersive locations and complex contents. Meanwhile the data resource, data processing node and data user requirement are all dynamic changing. In such complex conditions, the uncertainty of data and the uncertainty of demand pose a devastating problem with data sharing and effective use between the enterprises. How to break the old system interactive mode, how to integrate entire network enterprise data resources, how to provide one-stop uniform data search mode, in order to improve the efficiency of data sharing and utilization, are all the problems , which will be solved in the paper.

II. THE DESIGN OF THE SYSTEM FRAMEWORK

The enterprise data one-stop search system described in the paper is composed of the registration publication sharing module， which is in the data source end, and the cloud search framework module, which is in the data center.

The System Framework is as Figure 1.

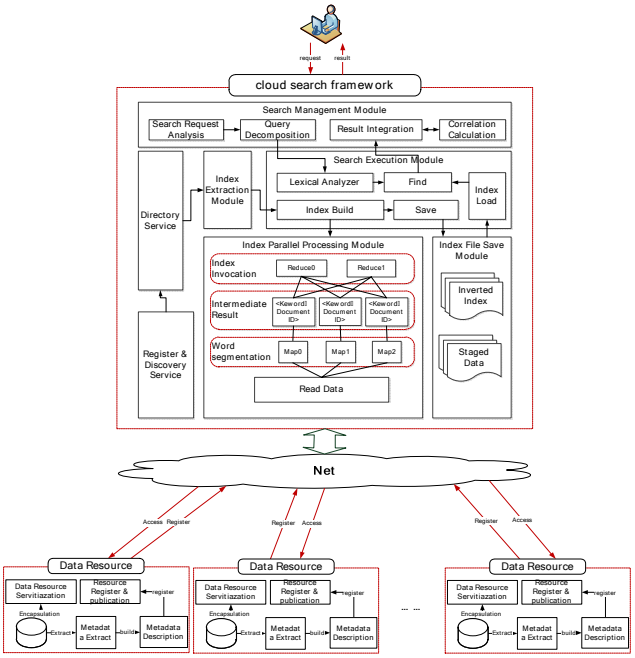

Figure 1. The enterprise data one-stop search system Framework

The registration publication sharing module， which is in the data source end, is used to provide metadata automatically extracting module, general data resources metadata description module, the encapsulation of service access interface of data resources and data resources general registration and publish module. The data resources above-mentioned is composed by the all kinds of enterprise data resource, which are always in form of the database, maybe the Oracle database, SQL Server Database or DB2 database and so on.

The cloud search framework module, which is in the data center, is used to uniform handle huge amounts of data resources, construct the data resources index, providing data integration ability. At the same time, it is used to receive user searching requests, retrieve index in the index database, and return the search results to the user. It includes register and found directory management module, data resources index management module， search management and execution module.

The register and found directory management module is used to realize all kinds of data resources registration and publication management and data resource directory management, in order to manage unified and publish the metadata of all kinds of data resources in the information grid;

The data resources index management module consists of three parts: the index extraction module, the index parallel processing module and the index file storage module. The index extraction module is used to fetch all kinds of data resource metadata information in the Register & Discovery Service, interact with the data resource encapsulation module, according to the metadata information of all kinds of data resource, and access the data content which can be shared in the information grid. The index parallel processing module is used to construct the index of all kinds of data resources in the distributed parallel processing approach of the cloud computing. And it provides organization management and integration ability of all kinds of data resources in the information grid. The index file storage module realizes the index storage in the distributed file storage way. The index, which is constructed by the index parallel processing module, includes the inverted index of all data resources and the data information which is disposed.

The search management and execution module is mainly used for receiving data search requests coming from the browser, and returning the search results to the user. It is composed of the search management module and the search execution module. The search management module is mainly used for receiving and analyzing user search request, query decomposing query request, interacting with search execution module, receiving the search results returned from the search execution module, and doing correlation calculation and results integration. The search execution module is the core processor part of the whole index building and executing management. It is used to realize the lexical analysis of search requests submitted by the search management module, loading index information stored in the index file storage module, fetching index and returning the index result.

III. THE MODEL AND METHOD

A. *Enterprise Data Resource Metadata Description Model*

The enterprise data resource metadata description model is mainly used for specification and form a unified metadata description of the enterprise data resources in information grid. It provides the unified management mechanism for the registration of publishing the data resources. It includes Data the resource description metadata, the data resource access metadata and the data resource structure metadata.

The enterprise data resource metadata description model is shown in Figure 2.

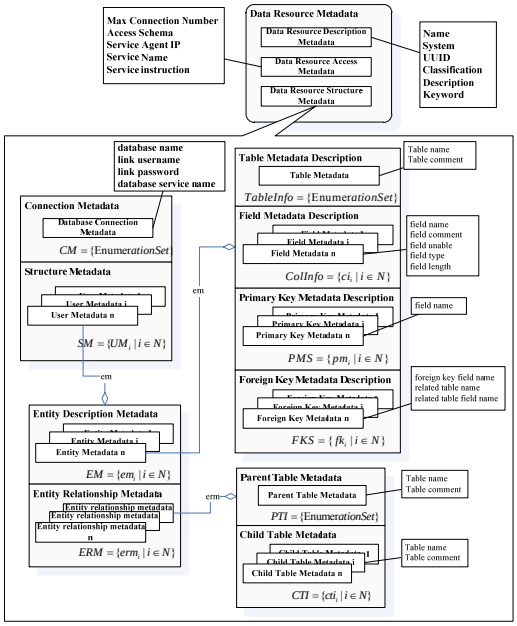

Figure 2. Enterprise Data Resource Metadata Description Model

The resource description metadata includes data resource name, belong system, UUID, classification, description and keyword.

The data resource access metadata includes max connection number, access schema, service agent IP, service name and service instruction.

The data resource structure metadata is composed by the connection metadata and the structure metadata.

The connection metadata (CM) is composed of the database name, the link username, the link password and the database service name, and these four items are the necessary information to establish a database connection.

The structure metadata (SM) is composed of several user structure database metadata. And one user structure database metadata can be divided into entity description metadata (EM) and entity relationship metadata (ERM).

Entity description metadata (EM) indicates the description of the database table under the database users. A database user always has a lot of database tables, so the user metadata entity description metadata is made up of a set of entity description metadata. The entity description metadata is covering all of the basic database table information. It is composed of the table metadata description, the field metadata description, the primary key metadata description and the foreign key metadata description four parts.

The table metadata description describes the table name and table comment information. The field metadata description describes all field information of the table. The field information includes field name, field comment, field unable, field type and field length. The primary key metadata description describes the primary key fields defined in the table. Usually, it is the set of field name in the table. The foreign key metadata description describes the relationship between the table and other tables. It uses the set of the

foreign key field name, the related table name and the related table field name to describe.

Entity relationship metadata (ERM) is used to describe the constraint relationship between the entities. It also means the relationship between the parent table and the child among all database tables. It is described by an entity relationship set. One entity relationship metadata includes the parent table name, the parent table comment and name list and comment list of its children tables.

B. Metadata Automatic Extraction Method

The metadata automatic extraction method provides the method how to automatic extract the metadata information of the data resources. The metadata automatic extraction tool is designed as the method said. The main procedure is as Figure 3.

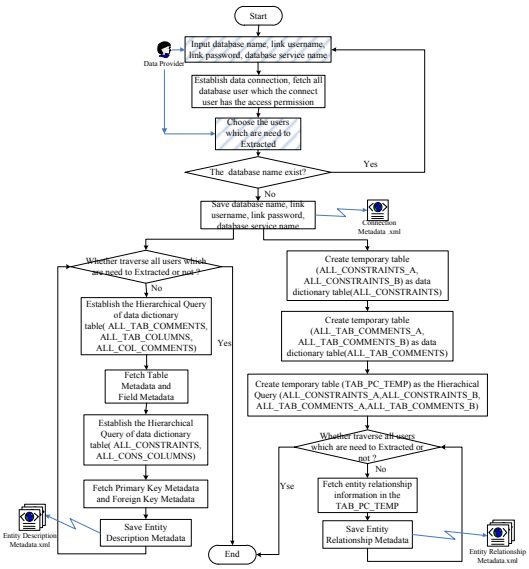

Figure 3. Metadata Automatic Extraction Flow

Firstly, Data provider use Input database name, link username, link password, database service name by the tool. The tool will establish data connection, fetch all database users which the connect user has the access permission. Data provider chooses the users, which are needed to extracted, from the fetched database users. Then the tool creates the new XML file (Connection Metadata .xml) for saving database name, link username, link password, database service name.

Secondly, the tool traverses all users, establishing the hierarchical Query of data dictionary table (ALL_TAB_COMMENTS, ALL_TAB_COLUMNS, ALL_COL_COMMENTS), fetching table metadata and field metadata. And establishing the hierarchical Query of data dictionary table (ALL_CONSTRAINTS, ALL_CONS_COLUMNS), fetching primary key metadata and foreign key metadata, then creating the new XML file (Entity Description Metadata.xml) for saving entity description metadata.

Thirdly, the tool automatically creates temporary table (ALL_CONSTRAINTS_A, ALL_CONSTRAINTS_B) as data dictionary table (ALL_CONSTRAINTS) and creates temporary table (ALL_TAB_COMMENTS_A, ALL_TAB_COMMENTS_B) as data dictionary table (ALL_TAB_COMMENTS). And then, creating temporary table (TAB_PC_TEMP) as the hierarchical query result (ALL_CONSTRAINTS_A, ALL_CONSTRAINTS_B, ALL_TAB_COMMENTS_A, ALL_TAB_COMMENTS_B). The tool traverses all users which are needed to extract, fetching entity relationship information in the TAB_PC_TEMP and creating the new XML file (Entity Relationship Metadata.xml) for saving entity relationship metadata.

Finally, the tool compresses three XML file together and then commits to the center when registering the data resource.

C. The Design of Index Extraction Module

The index extraction module will mainly interact with the register & discovery service, traverse all data resources registered in the register & discovery server, fetching the metadata of the data resource.

The index extraction module execution flow diagram is shown in Figure 4.

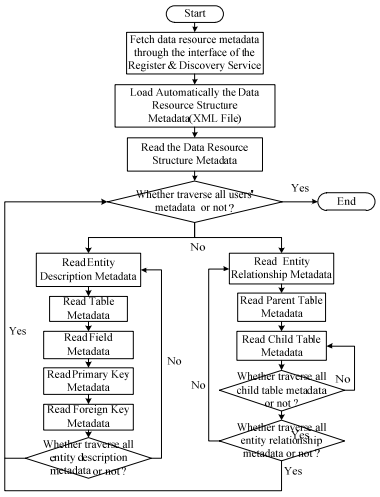

Figure 4. Index Extraction Module Execution Flow

Firstly, Fetch data resource metadata through the interface of the register & discovery server, load automatically the data resource structure metadata (XML File). And then, read entity structure metadata, traverse all user metadata, traverse read entity description metadata and entity relationship metadata respectively. Finally, generate entity array and entity relationship array.

D. The Design of Index Parallel Processing Module

The index parallel processing module is composed by the single entity content parallel extraction module, the correlative entity content parallel extraction module and the index parallel building module.

The target of the single entity content parallel extraction module is to crawl the data content from the remote data resource, based on the entity array, which is generated by the index extraction module, and the service interface

information descripted in the data resource access metadata. The whole process is parallel. Input: <server name, entity name>, use map mechanisms, map (server name, entity name) → output <key value1, content>, the entire process using multi-threading, crawling the data content from the remote data resource through the data resource encapsulation agent.

The correlative entity content parallel extraction module's goal is to generate data content set of double or several entities, through crawling the hierarchical data content form the correlative entity, according to the entity relationship array. the whole map-reduce parallel processing as follows: Input:<server name , entity name>, using the map mechanisms, Map (server name, entity name) → output <key value2, content>, the whole process also uses multi-threading and crawl the data content from the remote data resource through the data resource encapsulation agent.

The goal of the index parallel building module is to build index in the map-reduce parallel process way, according to the data content crawled above. The processing is as follows. Input: <key value, Content> document, the key value above including key value 1 and key value 2, Map (): split different key value, performing segmentation of the data content according to the custom index granularity and segmentation methods. Reduce () call Lucence index plug-in to generate the inverted index files, output: inverted index.

E. The Design of One-stop Search Execution Module

The one-stop search execution module receives the user search request in the web browse way. And then it will parse the search request. Firstly, it will have to do the lexical analysis of the search request, generating the key-value pairs. And according to the key-value pairs, it will generate the query composition through constructing logical operation (and-or-not among) the key-value pairs. Secondly, it has to replace respectively the key and the value by their thesaurus in order to generate new key-value pairs, which can improve the accuracy of the search. Thirdly, it will do the index search according to the all key-value pairs and generate index search result set. And then, it will do similarity computation with the index search result set and generate the result sorted list. Finally, it will integrate the results according to the logical operations and return to the user.

IV. Conclusion

All kinds of models or methods mentioned in this paper, which could effectively handle the problem of data sharing and utilization brought by heterogeneous structures, enormous quantity, dispersive locations and complex contents of enterprise data under the environment of grid, are mainly used to build up unified cloud search framework of enterprise data. It can provide timely and effective mass level data query access ability by using the cloud computing mechanism to build enterprise data index and one-stop query methods to provide search ability. It provides the main framework for building the enterprise data center in order to do integration and sharing among all enterprise data resources in the information grid.

REFERENCES

[1] You Chuan-chuan and Zhang Gui-gang, "A Kind of Efficient Search Method Based on Big Data" Computer Science, vol. 40, No. 2, March 2013, pp. 265-269.

[2] Zhu Ming-dong, Guo Zhi-long and Zhang Sheng, "Research on Data Sharing Service System Based on Data Center" Command Information System & Technology, vol. 1, No. 3, June 2010, pp. 18-22.

[3] Zhou Xiao-lei, Zhang Yan-qin and Sun Jin-hai, "Information Sharing Scheme for Network Centric Command Information System" Command Information System & Technology, vol. 2, No. 3, June. 2011, pp. 14-18.

[4] Cao Ju and Yin Zhe, "Clouds Search Optimization" Computer Engineering & Science, Vol.33, No.10, 2011, pp.120−125.

[5] Tang Yu Wang Ying-jie and Fan Ai-hua, "mDHT:A Search Algorithm to Extra-large Volume of Data Based on Open HDFS Platform and Multi-level Indexing" Computer Science, vol. 40, No.2, Feb. 2013 pp. 195–199.

[6] Wu Guang-jun, Wang Shu-peng and Chen Ming, "Massive Structured Data Oriented Storage and Retrieve System" Journal of Computer Research and Development, vol. 49, No. 1, Sep. 2012, pp. 1-5.