2010 Second IITA International Conference on Geoscience and Remote Sensing

Research and Implementation on Spatial Data Storage and Operation Based on Hadoop Platform

Yonggang Wang
Institute of Remote Sensing Applications
Chinese Academy of Sciences
Beijing, China
wangyg@bjgtj.gov.cn

Sheng Wang
Beijing Easymap Information Technology Co., Ltd
Beijing, China
wangsheng@easymap.com.cn

***Abstract*—As the deep development of spatial information sharing service, it brings forward high request to usability and expansibility of supporting system. Based on large-scale scalable server cluster, cloud computing brings hopes to resolves the existing difficult problems in the domain of geospatial information service. In this paper, we imported cloud computing technology including MapReduce model and Hadoop platform into the domain of geographic information system (GIS). Those key technology problems in the application of GIS such as spatial data storage, spatial index and spatial operation were described and studied in detail. We evaluated the performance and efficiency of spatial operation in Hadoop experiment environment with the real world data set. It demonstrates the applicability of cloud computing technology in computing-intensive spatial applications.**

Keywords—geographic information system;Hadoop;cloud computing; MapReduce; geospatial service

I. INTRODUCTION

GIS is now playing an important role in many areas of modern city. Space information has become the basic infrastructure of modern digital city and is the integral part of information construction [1]. Generally, GIS functions such as spatial analysis is involved in a lot of vector data (points, lines or polygons) and raster data (satellite or aerial images). This type of data is periodically generated via special sensors, satellites or GPS devices. Due to the large size of spatial repositories and the complicacy of the geospatial models, it take on upper time complexity and require high computing resources. As the deep development of spatial information sharing service, unborn synthetical GIS platform need present more abundant information, respond a great deal of parallel requests and achieve intelligent information processing. This type of information service aiming at Internet is different from prevenient scientific computing and transaction processing. It brings forward high request to usability and expansibility of supporting system and traditional high-performance computing and database systems are unable to meet those requests.

Cloud computing is a new term for a long-held dream of computing as a utility, which has recently emerged as a commercial reality [2]. Cloud computing provide service over the Internet to users based on large scalable computing resources. It is one of the latest trends in the mainstream IT industry. Cloud computing can provide a good developing environment with geographic information technology. It brings hopes to resolves the existing difficult problems in the domain of geospatial information service. Cloud computing can supply a reliable IT infrastructure to GIS effectively. Based on large-scale server cluster, cloud computing can improve system performance, computing and storage capability greatly, and reduce software and hardware cost effectively. Through implementing virtualization management with physical resources, cloud platforms have good system scalability and can supply mass geospatial computing needs.

In this paper, we import cloud computing technology including Hadoop platform and MapReduce parallel computing model into the domain of geospatial information services. We have studied those key technology problems including spatial data storage, spatial data index and spatial operation in the application of GIS. Aiming at the characteristics of spatial operators, we have designed the process flow of spatial data. We evaluate their performance using real spatial data set based on the actual implementations of these spatial algorithms on Hadoop. The experiment results show that MapReduce is applicable for computing-intensive spatial applications.

This paper is organized as follows. Section Ⅱ reviews the related work. Section Ⅲ introduces and describes spatial data object model, spatial data storage and the building of spatial index in detail. Section Ⅳ presents the implementation of the process flow of spatial operators. The experiments in section Ⅴ demonstrate the efficiency of the system. Finally in section Ⅵ, we summarize the contributions of our work and describe the future work.

II. RELATED WORK

Google has developed its infrastructure technologies for cloud computing in recent years, including Google File System (GFS), MapReduce and Bigtable. GFS is a scalable distributed file system which emphasizes fault tolerance [3]. Bigtable is a distributed storage system based on GFS for structured data management [4]. MapReduce is a programming model with associated implementation for massive data processing [5].

Hadoop is an open source implementation of the MapReduce parallel processing framework [6]. It hides the details of parallel processing, including distributing data to processing nodes, restarting subtasks after a failure and collecting the computing results [7]. Hadoop is composed of MapReduce runtime system and Hadoop Distributed File System (HDFS) which is an open source implementation of the Google File System (GFS).

A. MapReduce

MapReduce is rising parallel programming model which proposed by Google for large scale data processing on clusters of share-nothing commodity machines [5]. The model is

978-1-4244-8515-4/10/$26.00 ©2010 IEEE

stunningly simple and it can supports parallelism effectively. The programmer may abstract from the issues of distributed and parallel programming because MapReduce implementation could take care of load balancing, network performance, fault tolerance, etc.

In MapReduce, a program consists of a map function and a reduce function which are user-defined. The map function is applied to every input key/value pair to generate an arbitrary number of intermediate key/value pairs. The reduce function is applied to all values associated with the same intermediate key to generate output key/value pairs. MapReduce is used in a wide range of applications, such as web link-graph reversal, machine learning, statistical machine translation and so on.

B. HDFS

The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware [8]. A HDFS cluster consists of a name node and one or more racks of data nodes. HDFS is designed to store very large files by splitting them into a sequence of blocks, typically 64 MB in size. The blocks will be distributed across the cluster and replicated for fault tolerance.

The data nodes store HDFS blocks in files in their local file systems. Each data node has no knowledge about HDFS files, as this information is held on the name node. When a data node starts up, it scans through its local file system and generates a list of all HDFS data blocks that correspond to each of these local files. The name node has an in-memory data structure that contains the entire file system namespace and maps the files on to blocks. It also keeps a log file that records all the transactions. The data nodes send regular heartbeats to the name node so the name node can detect data node failure.

III. SPATIAL DATA STORAGE AND INDEX

Compared with the previous data objects concerned by Hadoop platform such as URL address used by web search and all text strings, spatial objects are generally larger and have more complex structure and properties. Each object not only has spatial properties such as points, lines and polygons in itself, but also a collection of the same type of objects have spatial relations such as topological relations and the relationship between spatial clustering. In order to provide a unified data access and interoperability interface for upper applications and simplify the process of data access, we generally do not use the HDFS data management interface directly to access and operate spatial data. But we need to build spatial data objects understandable in application logic and to achieve physical storage and data management of spatial object on Hadoop platform.

A. Spatial Object Model and Storage

1) Spatial Data Object Model

Currently, there are some mature research results about spatial data object model. The most successful achievement is the simple feature model proposed by the Open GIS Consortium (OGC) [9]. Because of the elegance and simplicity of the model, we have adopted it in this Cloud-based geospatial information application. The OpenGIS simple features specifications define various spatial operators, which can be used to generate new geometries from existing geometries. So it has good scalability.

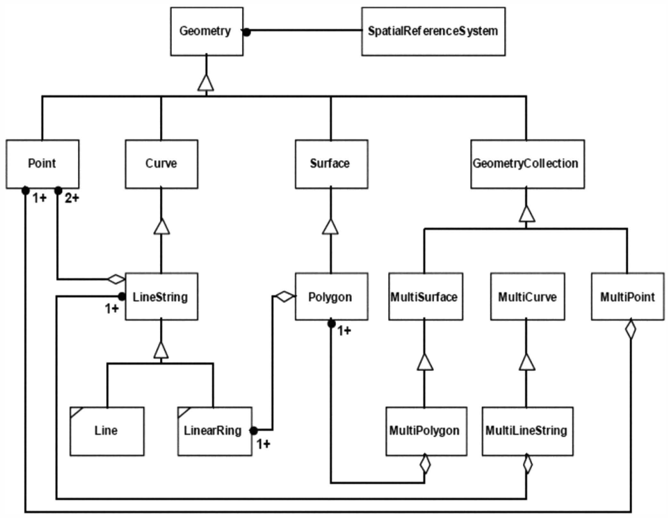

Figure 1. OGC simple feature code model

2) Physical Storage of Spatial Data

On Hadoop platform, non-spatial attribute of spatial data objects can be accessed by common data types such as string or number. But spatial attribute of spatial data can not use general hash file to process because it contains a large number of geometric coordinate information.

In my paper, we propose a method which compress spatial attribute into a binary large object to store. We adopt the persistent coding rules of the object which have been regulated by the OGC and described in their Access and Coordinate Transformation Service Specifications. It is Well Known Text (WKT) and Well Known Binary (WKB). WKT is a text markup language for representing vector geometry objects on a map, spatial reference systems of geographic objects and transformations between spatial reference systems. WKB is used to transfer and store the same information in binary bytes. For example if we have a point object with the x coordinate and y coordinate value is 1, then the equivalent WKT and WKB formats are given in the table below.

TABLE I. WKT AND WKB EXAMPLE

OGC Service Specifications	**Formats**
WKT	POINT(1 1)
WKB	0101000000000000000000F03F000000000000F03F

3) Spatial Data Storage Based on HDFS

For HDFS file system, its operation requires the data file is divided into a lot of small objects which are stored in different processing nodes. And we also need different Map/Reduce processing nodes in order to achieve a distributed storage and computing. So this system needs to segment a large data set into a small discrete documents managed by the HDFS in accordance with the scope and the size of space.

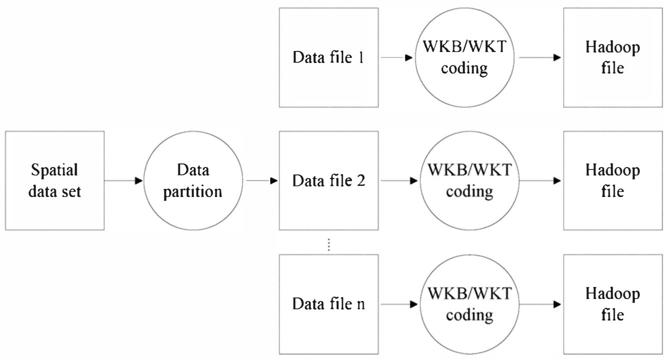

Figure 2. Spatial data file management based on Hadoop

B. *Spatial Data Index*

For spatial data through HDFS document management, if not build spatial index, then the efficiency and speed of data access will be greatly affected and the spatial operator can not be achieved. But the spatial index is a global data index, while Hadoop is a distributed data processing model. There is some difference between them. So we proposed a hybrid index strategy. In the process of data preprocessing, spatial search and spatial operation, we choose spatial index. For attribute data query and dynamically generated data we achieve it through distributed search function provided by Hadoop.

Currently, spatial index mainly adopt Quad-Tree and R-tree algorithms. A Quad-Tree is a tree data structure in which each internal node has up to four children. It is commonly used to partition a two-dimensional geographical space by recursively subdividing it into four quadrants or regions. The regions may be square or rectangular, or may have arbitrary shapes. All forms of Quad-Tree divide space into adaptable cells which has a maximum capacity. The tree directory follows the spatial decomposition process. R-Trees are tree data structures that are similar to B-Trees, but are used for spatial access methods. The data structure splits space in the hierarchically nested way, and possibly overlapping minimum bounding rectangles (MBR). Each node of R-Tree has a variable number of entries which are up to some pre-defined maximum. Each entry within a non-leaf node stores two pieces of data, which include a way of identifying a child node and the bounding box of all entries within this child node. After the creation of Quad-Tree and R-tree, the fast lookup of static data is supported.

To achieve dynamic spatial data index on Hadoop, we need to divide index data into the basic unit handled by each node. Then the same Map/Reduce node can simultaneously handle the data file and its corresponding index file. After generating dynamic results through query or spatial operations, we can get the data corresponding index file which is incorporated into the unified management with data processing results.

For each index file, in order to achieve local data aggregation we adopt the spatial cluster method which is the index form of Hilbert ranking. The so-called spatial cluster is those grids close to each other. The numbers generated after Hilbert ranking are also close to each other. For each Hadoop spatial data file which needs to create index, spatial index can be generated in its MBR space.

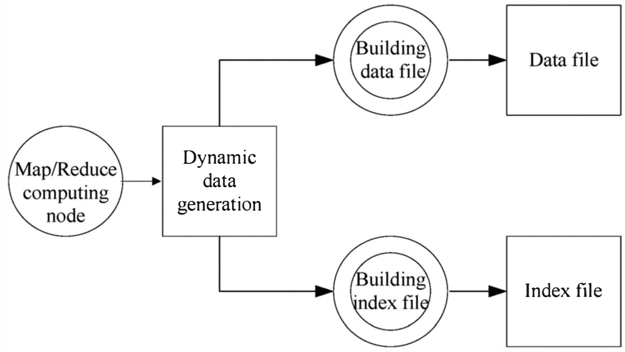

Figure 3. Dynamic index file generation on Hadoop

IV. IMPLEMENTATION OF SPATIAL OPERATOR

Space operator is a series of spatial operation methods and the basis of spatial analysis and processing. Space operators include the relationship operation among space objects such as space intersect, joint and overlap. For the complete description of spatial operator, the so-called nine-intersection model can define. This model describes nine spatial relationships between spatial objects such as disjoint, touch, contain and cover.

In MapReduce model, in order to achieve distributed computing spatial operators have to be divided into two stages including Map and Reduce to realize. Aiming at the characteristics of spatial operators, we have designed the following process flow of spatial data.

- Data Preprocessing. In order to achieve a single node for processing, we need do preprocessing to two data sets involved in spatial operator. In accordance with the foregoing spatial data processing and indexing, data and their index data files are generated into Hadoop data file and distributed in different processing nodes. Data files mainly include the data objects which are stored in the way of (K, V). Among them, K includes the unique ID of spatial data and V contains the geometric data, attribute data, data sources and so on.
- Map Process. Any two spatial objects which need to be addressed are filtered firstly by the map process. That means through judging by the index data MBR, those spatial objects which clearly do not meet the requirements are filtered out. And the spatial objects which potentially meet the operational requirements of spatial operators are preserved and generated into intermediate object files.
- Reduce Process. After receiving the intermediate file generated from the map process, we can choose appropriate algorithms to do spatial relations processing with two data objects. The results are generated object file. In the actual processing, for implementation process of each operator, the technical flow is not the same.

V. PERFORMANCE EVALUATION

A. *Experiment Environment and Data Sets*

In order to verify the technical feasibility, we have established a Hadoop test environment. And aiming at the basic technical process, we have developed the corresponding test and verification procedures so as to test the applicability of Hadoop in geospatial processing.

We use Hadoop cluster of ten nodes configuring the RedHat Linux operating system as the test platform. Among them, five machines have 4G memory and dual-core CPU and each machine runs five map and reduce process. Another five machines set with 2G memory and each machine runs three map and reduce process. The Hadoop cluster configration is given in Fig. 4.

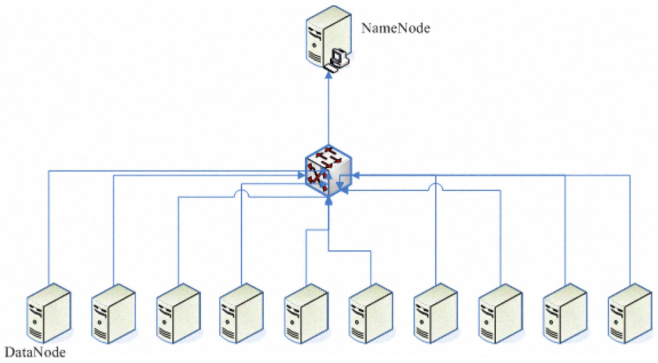

Figure 4. Hadoop Cluster Configration

We choose the vector data of a city to do the test. Test data sets include building information layer and road information layer. Building information layer includes more than two million building polygons information and road information layer include more than five thousand road line data.

B. Experiment Results

Test mainly verifies the efficiency and accuracy of data query, and the performance of data analysis through spatial operators. To compare the performance difference between Hadoop platform and the existing spatial database, we select OracelSpatial as control platform for the same query and analysis operations. The statistical result is given in the tables below.

TABLE II. ATTRIBUTE QUERY

Type	Record Number	Query	Average Time (Hadoop)	Average Time(DB)
building	2013562	building code	294ms	247ms
road	5492	road name	124ms	143ms

TABLE III. SPATIAL QUERY

Type	Record Numbe r	Query	Averag e Numbe r	Average Time (Hadoop)	Average Time (DB)
buildin g	201356 2	Rectangle query (1km*1km)	4826	391ms	432ms
road	5492	Rectangle query (1km*1km)	32	307ms	329ms

In order to verify the spatial operator, we have designed the test of calculating road intersection. That is calculating the intersection of any two roads and outputting the result. Through calling Map/Reduce calculation function of InterSect operator, the calculated number of nodes is 24526. To complete this calculation process Hadoop platform need take 126 seconds, while the OracelSpatial platform takes 284 seconds.

From the above results we can learn that Hadoop platform and database platform have similar computing performance in spatial query, but Hadoop has a distinct advantage in spatial operations.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we import cloud computing technology such as MapReduce programming model and Hadoop platform into the domain of geospatial information services and study the basic key technology problems such as spatial data storage, spatial index and implementation of spatial operator.

Using real geographic data set, this paper also presents a performance evaluation of spatial operation comparing Oracel Spatial with Hadoop platform. The results demonstrate the feasibility and efficiency of the MapReduce model and show that cloud computing technology has the potential to be applicable to more complex spatial problems.

However, my paper just does some preliminary research with the technology problems including spatial data storage, index and operation. To reach the aim for geospatial cloud service, a lot of work need be done such as implementation of spatial analysis, building of geo processing workflow and mass spatial data storage. These are our future interested work.

REFERENCES

[1] Longley, P.A., Goodchild, M.F., Maguire, D.J., Rhind, D.W., Geographic Information Systems and Science. John Wiley & sons, New York, 2001.

[2] PARKHILL,D, The Challenge of the Computer Utility. Addison-Wesley Educational Publishers Inc., US, 1966.

[3] G. Sanjay, G. Howard, and L. Shun-Tak, “The google file system,” In SOSP’03: Proceedings of the 19th ACM Symposium on Operating Systems Principles, pp. 29–43, 2003.

[4] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A.Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber, “Bigtable: a distributed storage system for structured data,” In OSDI ’06: Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation, pp. 15, 2006.

[5] J. Dean and S. Ghemawat, “MapReduce: Simplified Data Processing on Large Clusters,” In OSDI’04, 6th Symposium on Operating Systems Design and Implementation, Sponsored by USENIX, in cooperation with ACM SIGOPS, pp. 137–150, 2004.

[6] The Hadoop Project, http://hadoop.apache.org, 2008.

[7] A. Bialecki, M. Cafarella, D. Cutting, and O. Malley, Hadoop: a framework for running applications on large clusters built of commodity hardware, Wiki at http://lucene.apache.org/hadoop, 2005.

[8] Dhruba Borthakur, The Hadoop Distributed File System: Architecture and design, Document on Hadoop Wiki, 2008.

[9] OGC, http://www.opengis.org/techno, 2009.