The Massive Remote Sensing Data Organization and Management Strategies

Hou Wei[1], Zhang Yuheng[1]

[1]*Harbin space star data system technology Co., Ltd., 150028 HARBIN, CHINA*

Abstract. With the continuous development of earth observing technology of China, Remote sensing image data has characteristics of large volume, multisource, multi-type and multi-resolution, finding a way to store, manage, and publish the data is a big challenges. This paper proposes a massive Remote sensing tile data organization and management strategy based on spatial database, design a data distribution and node management strategy to solve the massive remote sensing data organization and management problem.

1 Introduction

With the continuous development of earth observing technology of China, Earth Observation System (EOS) will form the characteristic of temporal and spatial coordination, all-weather, all-time, and it can also obtain massive remote sensing imagery[1-2]. Remote sensing image data has characteristics of large volume, multisource, multi-type and multi-resolution, finding a way to store, manage, and publish the data is a big challenges. The research on the storage of massive remote sensing image data presents a new research trend: First, the storage architecture of data present smaller and denser and the way of data storage present more discretization. The scale of data storage is gradually developed from the original single storage mode to the large-scale cluster. In addition, the research presents a special-purpose Remote Sensing Data management system according to the characteristics of remote sensing image, which can gradually replaces the leading role of the Relational database management system (RDBMS) in the area of massive data management. In this development trend, the method of distributed and high performance parallel processing will be the first solution to the comprehensive management of massive spatial information data[3-4].

2 Current Situation on the Research of Massive Image Data Management System

The representative massive image data storage management systems mainly contain Google Earth/Maps[5], the open-source geographical science software World Wind[4] launched by NASA, the online map service Bing Maps[6] released by Microsoft, online public atlas TerraServer[7-9], and national geographic information public service platform Map World[1].

Table 1. CPMPAROSPM OF EACH IMAGE STORAGE MANAGEMENT SYSTEM.

Management system	Data storage organization	Physical storage architecture	Spatital reference	Data format	Data storage management
Google maps	Quadtree-based tile layer	Distributed server cluster storage and cloud computing	Mercator Projection	JPEG, PNG	BigTable and GFS
World Wind	Tile layer based on the spherical grid	Centralized server cluster storage	Plate caree projection	JPEG, PNG	Distributed file system
Bing Maps	Quadtree-based tile layer	Distributed server cluster storage and cloud computing	Mercator projection	JPEG, PNG	Windows Azure and SQL Azure
TerraServer	UTM zone division and data tile	Centralized server cluster storage	UTM	JPEG, GIF, TIFF	Blob-based SQL Server database
Map World	Pyramid tile based on equal intervals of the latitude and longitude	Distributed server cluster storage	CGCS2000	JPEG, PNG	Commercial database and file

[a] HouWei: 892118874@qq.com

Comparison of each image storage management system is shown in Table 1. The management of tile data mainly base on multi-resolution pyramid of spherical coordinate system, physical storage is classified as distributed and centralized storage sever cluster, general data format supports TIFF, JPEG and PNG, data storage has file storage and BLOB of relational database.

Based on current remote sensing data organization format and the development trend, organizing massive global remote sensing data with latitude and longitude to establish a geographic Lat/Lon projection spherical subdivision system , is an effective way to unitedly store massive global remote sensing data. Meanwhile, as tile data stored by relational database, storage capacity is restricted to DBMS, database technology will reach bottleneck once the data is too big, and relational database is not suitable for unstructured big data due to characteristic; if using pure file system, remote sensing data will be stored in file format to meet the demand of mass storage, but the organization strategies of massive tile file will directly affect the storage and retrieval efficiency.

3 The Massive Remote Sensing Data Organization and Management Strategies

To resolve PB-level remote sensing data organization and application, proposes a massive Remote sensing tile data organization and management strategy based on spatial database, establish distribution rules for date between multiple nodes,and the multisource heterogeneous remote sensing data management.

3.1 Data Organization and Management

This paper designs the Multi-scale organization management model for Remote sensing spatial data guided by the "horizontal multi elements, vertical multi-level and multi-scale". As shown in figure 2-1,it means the horizontal uses high-resolution remote sensing data as the basic unit to realize the data organization of different spatial geographic resource types and the vertical manages with multilevel resolution. Which achieves horizontal and vertical management functions for different levels of different types of resources. The management of spatial data resources at the national, provincial and municipal levels takes the county as the basic unit, and the logical relation of the national, provincial and municipal levels is managed by the regional characteristics. Different types of resources are established at the county level according to the spatial region. A resource builds different types of galleries according to scale, time, etc..That is the basic geographic spatial spatio-temporal data at the county level visit the five granularities(the resource, the map depot, the layer, the layer set and the surface features) which are divided by the resource authorization granularity to mange data.

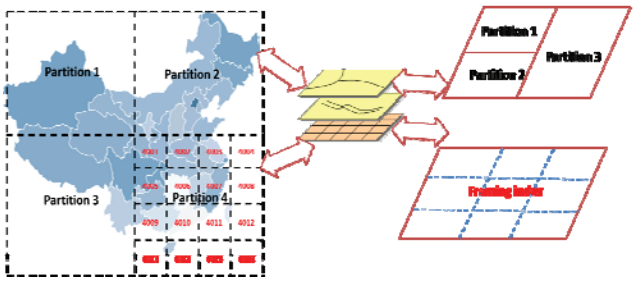

Figure 1. Multi - scale organization and management model of remote sensing spatial data resources.

3.2 Storage Strategy

3.2.1 Distributed Strategy

For remote sensing data users, each query's performance due to the dispersion and equilibrium of remote sensing data distribution. So we should require the data fragmentation as high as possible and satisfy the load balancing requirements by satisfying the data volume of each node. In this way, we can ensure each node machine retrieves the same amount of data as the same amount of data in parallel retrieval, and it can store almost the same amount of data for each node machine.

Use hash functions to do data fragmentation is a common method as hash functions have excellent dispersion to assures the load balance.

Hashing algorithm use hash functions to modulus operator on the key, and put the result as the key's Hash table entry address. If we need to map an object key uniform mapping to nodes, we should use the result of Hash (key)%N to make sure which node to storage.

This paper's distributed strategy draw lessons from hashing algorithm, the formula is:

$$H(key) = (3*row + col)\%K \quad (1)$$

H(key) is Split space number, K is Maximum number of virtual nodes, K＞H(key)＞0, row is the row number of the lower left corner of the tile, col is the column number of the lower left corner. Split space number was at the top of the tile file tree structure.

This gives you any 3 * 3 tile requests, 9 of which have different space Numbers. At last, using fixed pixel size (e.g. 512*512) tile image to correspond each grid tile under each level.

For the mapping of the split space number with the actual physical node, for example, the number of actual physical nodes is n (n < = K), the number of fragments allocated by each node should be between (INT) K/n and (INT) K/(n + 1) and the difference between each node is n. In this paper we design the max number(K) of storage nodes is 255, n is 9, the split space number information is shown in the following table

Table 2. Space sharding example

Physical node number	Application server	Split space number
1	192.168.1.1	0,9,18,27,36,45,…,243,252

2	192.168.1.2	1,10,19,28,37,46,…,244,253
3	192.168.1.3	2,11,20,29,38,47,…,245,254
4	192.168.1.4	3,12,21,30,39,48,…,246
5	192.168.1.5	4,13,21,31,40,49,…,247
6	192.168.1.6	5,14,23,32,41,50,…,248
7	192.168.1.7	6,15,24,33,42,51,…,249
8	192.168.1.8	7,16,25,34,43,52,…250
9	192.168.1.9	8,17,26,35,44,53,…251

When K is 255, each tile's store split space number like 3 table.

Table 3. The tile spacer

0	1	2	3	4	5	6	7	8	…	250	251	252	253	25
3	4	5	6	7	8	9	10	11	…	253	254	0	1	2
6	7	8	9	10	11	12	13	14		1	2	3	4	5
9	10	11	12	13	14	15	16	17	…	4	5	6	7	8
12	13	14	15	16	17	18	19	20	…	7	8	9	10	11
15	16	17	18	19	20	21	22	23	…	10	11	12	13	14
18	19	20	21	22	23	24	25	26	…	13	14	15	16	17
21	22	23	24	25	26	27	28	29	…	16	17	18	19	20
24	25	26	27	28	29	30	31	32	…	19	20	21	22	23
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
242	243	244	245	246	247	248	249	250	…	237	238	239	240	24
245	246	247	248	249	250	251	252	253	…	240	241	242	243	24
248	249	250	251	252	253	254	0	1	…	243	244	245	246	24
251	252	253	254	0	1	2	3	4	…	246	247	248	249	25
254	0	1	2	3	4	5	6	7	…	249	250	251	252	25

The whole table is like a remote sensing data, each of them represent a tile image data, just like the remote sensing is divided and processed to large number of tile image data. We can find that for any 3*3 request, the 9 tiles data involved are stored in different pieces space and the space number is continuous. In this way, we can ensure the uniformity to each node is the highest and most efficient. Even n*n requests, the nodes of each tile distribution are scattered, not concentrated in some nodes, so we can guarantee the efficiency. From the point of each line, each line includes all continuous shard space (0-254), each shard space to store the number of tiles are equal, then each physical node actually stored almost approximate equal number of tiles. So the whole picture of remote sensing image data are processed data generated by the tiles according to the hash distribution strategy of this research will be evenly dispersed to different storage nodes, each tile data volume of a storage node is approximately equal to meet the needs of the query efficiency and load balancing.

3.2.2 Redundancy Strategy Distributed Strategy

Even the distributed strategy based on hashing algorithm have excellent ability to ensure the load balancing of the system, but lack of design for system security and reliability. When one storage node invalid, the node's tile data will lose causing great loss to the system.

Most of the current distributed storage system tend to use fault-tolerant to increases the system reliability. Fault-tolerant means allow the system to fail and requires that relevant functions and services not be invalidated when the fault occurs.

The basic idea of this paper redundancy strategy is design a new method "copy of spatial data fragmentation" based on hashing algorithm to set up the distributed redundant deployment solutions. This method increase the redundant slice space to the physical node to increase the copy of the system, the fault tolerance of the system can be effectively improved, thus ensuring the reliability and security of the distributed system. The specific method is: add two copies of the spatial lamination number obtained by the hashing model, placing two copies in the previous node and the last node. Then each tile data will be stored in three node, two of them is copy data. When the primary physical node malfunction, the others alternate physical node can provide tile data, each node spatial lamination number have three list to record it.

Table 4. the spatial lamination number corresponding to the i physical storage node

Copy node of i-1	I node	Copy node of i+1
(i-1)-1	i-1	(i+1)-1
(i-1)-1+n	i-1+n	(i+1)-1+n
(i-1)-1+2n	i-1+2n	(i+1)-1+2n
(i-1)-1+3n	i-1+3n	(i+1)-1+3n
…	…	…
(i-1)-1+kn	i-1+kn	(i+1)-1+kn

3.3 Data Organization and Management

The final data that the cloud storage master transmits to the user is the integration of data from the various storage nodes, as the dynamic and autonomous nature of the resources, each node allows self-maintenance of the current situation and correctness of the data, which is bound to cause the data inconsistencies between the nodes. So users need a certain scheduling mechanism when they need data.

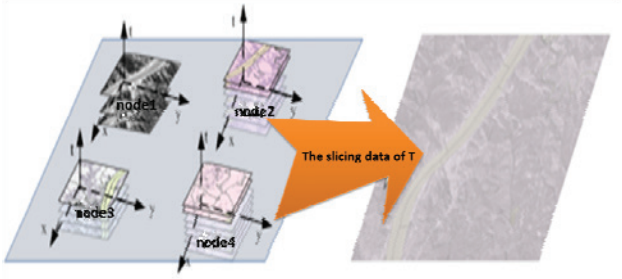

Figure 2. Inter - point spatial - temporal data integration.

As shown in Figure 2, the grid consists of off-site distribution of each network, each network has its own data resources and data organization, storage format. Over time, the frequency of changes of each node data inconsistency. As shown in the figure, the update of the image data is slower in the same time interval, and the updating of the vector data varies depending on the frequency of the change of each node element. So different time points can be formed at different time slices. When the central node needs to provide users with a time slice of the integrated data, the need for the node of the data time cutting, projection, and finally integrated in the same reference, the formation of a time and space data provided to the user.

4 Conclusions

This paper proposes a massive Remote sensing tile data organization and management strategy based on spatial database, design a data distribution and node management strategy to solve the massive remote sensing data organization and management problem. Not only ensures that each node machine stores almost the same amount of data, but also makes the amount of tile data retrieved by each node machine in the same way, which effectively improves the system's concurrent query performance and load balancing. Taking into account the reliability and availability, the design of the "fragment space copy" redundancy strategy, effectively guarantee the system tile data security and system reliability.

References

1. X. F. Lv, C. Q . Cheng, J. Y . Gong, ―Review of Data storage and Management Technologies for Massive Remote Sensing Data.‖ Science China Technological Sciences, vo.54, pp.3220-3232, 2011.
2. Jian Shen, ―Research on Object-Oriented Relational Image Database Technology Based on XML Metadata.‖ Nanjing Normal University, 2008.
3. M. Gibin, A. Singleton, R. Milton, ―An Exploratory Cartographic Visualization of London Through the Google Maps API. ― Appl Spat Anal Pol, vol.2, pp.85-97, 2008.
4. D. G. Bell, F. Kuehnel, C. Maxwell, ―NASA World Wind: Open source GIS for Mission Operations.‖ In Proceedings of IEEE Aerospace Conference. Big Sky, MT, pp.1-9, 2007.
5. B.KARP(2006). The Google File System. [Online]. Available:http://wenku.baidu.com/view/3e4ba4efaea ad1f346933f9f.ht ml
6. I. Foster and C. Kesselman―Globus: A Meta-computing Infrastructure Toolkit.‖ International Journal of High Performance Computing Applications,Vol.15,pp. 359-374,2001.
7. T. Barchlay, W. Chong, J. Gray, ―TerraServer Bricks– a high availability cluster alternative.‖ Microsoft Technical Report, MSR-TR- 2004-107, 2004.
8. T. Barchlay, W. Chong, J. Gray, ―TerraServer SAN-Cluster architecture and operations experience.‖ Microsoft Technical Report, MSR-TR-2004-67, 2004.
9. T. Barchlay, W. Chong, J. Gray, ―Microsoft TerraServer: a spatial data warehouse.” In: Proceedings of the ACM SIGMOD International Conference on Management of Data. Dallas, Texas: ACM, 29(2): 307– 318,2000.