Security control heterogeneous big data cloud storage system based on adaptive cache

Jianbao Zhu*, Jing Fu, Yuwei Sun, Ye Shi, Yu Chen, Weichao Deng
Nantong Power Supply Branch of State Grid Jiangsu Electric Power Co., Ltd., Nantong, Jiangsu, 226006, China
*Corresponding author’s e-mail: yuxinchunntpower@163.com

Abstract. In order to improve enterprise information security and the performance of storage system in data processing process, the application of adaptive memory cache mechanism in remote sensing data storage is studied. Firstly, alluxio is used to build a unified virtual file system, and the caching mechanism based on spatial relationship and historical data access inference is studied. Secondly, Presto based on memory computation is introduced as the storage component of remote sensing data meta-data, providing real-time read-write support for massive remote sensing data meta-data and advanced semantic information. Through experiments, the process of adaptive memory buffer remote sensing data storage system takes less time than the traditional storage method.

1. Introduction
In a modern society with developed technology and information circulation, people's exchanges are getting closer and closer, and life is becoming more and more convenient. Big data is the product of this high-tech era. Nowadays, network information technology has a very wide application rate in the society, and the level of information science and technology has also been significantly improved. It has promoted the rapid development of enterprises in all walks of life in our society. Of course, in its development. There are also some hidden dangers of security. Moreover, because network security problems may cause great economic losses, resulting in adverse effects of enterprises, there are two main reasons for these effects. One is information technology, the second is the reason for enterprise network management. This shows that in addition to the information technology to solve the problem of enterprise network security, enterprise information management needs to pay attention to network management issues. For this reason, it is important to pay attention to the information security of enterprises, continuously research modern information security technology management, and establish a sound information security management framework, which is of great significance to enterprise development. With the development of remote sensing technology and the use of high-resolution series satellites, the data received show the characteristics of multisource heterogeneity and mass. The large remote sensing data storage research mainly divided into the following several ways:

1) Distributed storage, such as the way of HDFS is one centralized way. In order to improve the ability of fault tolerance of the system and expansion, such as publicity design of remote sensing image based on hadoop business management system. According to the data heat using tertiary storage mode, which can improve the reading and writing speed of HDFS in remote sensing image storage.

2) Mainly study the storage and organization mode of massive remote sensing data to improve the access efficiency of remote sensing images in practical applications. Nie pei proposed a distributed storage method for remote sensing images, designed an image storage model for distributed
environment [6]. It provided better aggregation of data, and designed a configurable data partitioning strategy.

In the design and application of remote sensing big data system, the efficiency of storage system has a very important influence on the operation efficiency of the whole application system. Especially in distributed remote sensing data processing workflow, since the input of the latter process is the result of the forward process. It leads to frequent storage system access, increases the running time of the process, and increases the load of the storage system.

In this paper, alluxio is introduced as a tool to establish a virtual unified file system structure and to accelerate distributed memory cache. alluxio can carry out distributed adaptive cache of remote sensing image data across nodes, which can avoid the frequent data being written and read consuming extra time and resources. In addition, many spatially adjacent remote sensing images have a high probability of being processed together, so multisource heterogeneous remote sensing data is constructed based on meta-information and high-level semantic multidimensional index structure, and in the storage system. Recording the access history of the data, adaptively buffering the required data according to the existing information, can effectively improve the hit rate of the cache system.

2. Composition structure of the system
By establishing the unified file system UFS, the storage layer can automatically adapt and integrate existing disk arrays. Meanwhile, a distributed file system based on HDFS is established. The structure of the system is shown in figure 1:

![Figure1. Structure chart of memory-based RS data storage system](image)

Due to the continuous reduction of unit memory price, alluxio virtual storage system based on the concept of using memory as hard disk is gradually welcomed by the industry. alluxio can unify various storage systems into one file system, provide storage services for top applications such as Spark, and use distributed memory caches based on file system access. The use of UFS to establish a virtual unified file system can ensure that the existing data storage system can be seamlessly upgraded to the new storage system, also can use the data source of the newly established distributed storage system.

HDFS is a distributed file system, which is characterized by the need to build a reliable and stable file system with good scalability on ordinary hardware. HDFS has a highly fault-tolerant mechanism to ensure the reliability of file system data deployed on cheap machines. Presto is an interactive SQL query engine developed by Facebook that is completely based on memory and distributed parallel computing. Presto is a kind of MPP architecture (Massively Parallel Processing). It can reduce the difficulty of remote sensing data retrieval and making the system more user-friendly.

3. Reading and writing process analysis of remote sensing data
The use of the remote sensing image storage system supports data exchange with the storage system using traditional methods such as NFS and FUSE, and also supports the use of HDFS and Alluxio APIs, regardless of the method used, because the data is in the adaptive cache. Storage management system's unified name-space, so you can get higher comprehensive performance than traditional storage methods Data can be organized in the traditional way of folder organization, and SPATS [9] (spatial-temporal-spectral) data structure with integrated organization management of 4 dimensions of time-space-spectrum. This way can increase the cache hit ratio under specific application scenarios.
3.1 Data Writing
When the remote sensing image data is written, different caching methods can be used for different levels of data as needed to improve the data writing speed or the reliability of the data as needed. For the key meta-data and important production results, the data must not be lost. For the writing of intermediate data, the data writing speed should be increased as much as possible. In the case of node failure, the fail-over mechanism should be automatically enabled to ensure a high success rate of data writing.

The caching methods of data writing are mainly divided into the following ways:

1) CACHE_THROUGH mode. Remote sensing image data is written synchronously to the cache system and the underlying storage system to ensure that the data will not be lost.

2) MUST_CACHE mode. Remote sensing image data is only written to the cache system, but it will not be written to the underlying storage system. In this case, the writing speed of data is only slightly slower than the memory writing speed, which has high performance and can be used as a temporary file writing method.

3) THROUGH mode. Remote sensing image data is written synchronously to the underlying storage system, but it is not cached when written. This situation is suitable for writing archival images because archival images are not used temporarily after inhalation.

Before the data is written, the corresponding basic meta information, such as coordinate system, longitude and latitude information can be automatically extracted, but some other descriptions of remote sensing images, such as data sources, may be lost, which requires manual input when warehousing or after warehousing.

For advanced semantic features of the image, such as NDVI and main ground object types, online extraction is carried out according to the load of the system when the image is stored in storage. Figure 2 shows the flow of data warehousing, which is cached to the node where the broker is.

3.2 Data Reading
Due to the application of the memory-based distributed database system Presto and the rich description of remote sensing images when data is written, traditional query methods based on coordinate ranges and data source types can be used. It can also use advanced semantic information such as vegetation index to query, which can provide users with more targeted data from the mass remote sensing image. After the image retrieval is completed, it will enter the file reading stage. Due to the existence of cache system, when reading remote sensing image data, the client does not need to care about the specific storage location of the data. It only needs to read from the virtual unified file system in the traditional way. The following situations may occur during reading.
1) The data is already in the cache of this node, so the reading speed is equal to the reading speed of memory.

2) If the data is in the cache of other nodes, it is directly read from the cache of other nodes. And the reading speed is limited by the maximum speed of the network.

3) If the data is not in cache, the data will be read from the underlying file system. And the reasoning will be carried out according to the conditions at random airport to decide whether to cache the data, and further decide whether to asynchronously read other related data in advance to improve the hit ratio of subsequent cache system.

When reading a specific remote sensing image file, it may also occur that part of the data of the file has been cached and the other part needs to be read from the underlying storage. In general, the adaptive remote sensing image cache system improves the reading performance obviously.

4. Image caching strategy

With the advent of the era of big data, the rapid increase in the amount of image data has brought great pressure to the storage and transmission bandwidth, so the application field of image collection and storage has become very extensive. Although image acquisition is a kind of data acquisition, it has its particularity:

1) High speed. In order to satisfy the real-time and full time video, the data collection frequency rate of the data to be obtained should be up to several MHz.

2) Frame image storage. In the image, the frame is a single bit, and only the completed frame has complete information, so the whole frame data needs to be saved every time.

3) Strict timing. The general image sensor is fixed timing output. The corresponding image data is input at the corresponding time, otherwise the data is lost. This feature creates a lot of pain for image collection.

The following are some common methods for comparison:

1) Cache with SRAM. The advantage is that the interface is easily accessible and addressable. However, it is unable to access at the same time, with high power consumption, limited capacity of a single chip, and the current 512KB single SRAM is close to the maximum. Power consumption, volume, and cost become higher when multiple chips are used.

2) Cache the rows using FIFO. The advantage of it is that the speed is fast, the interface pin is few, and can save at the same time. However, the volume is small. It is often seen below 1.6kB, and the current 64KB single piece SRAM is nearly the largest.

3) Cache BUFFER, which is very similar to FIFO, and can be stored quickly. However, the capacity and price reach the level of SRAM, and the power consumption, volume and cost become higher when the capacity is larger.

It can be seen from the above analysis that these methods are not suitable for high resolution image acquisition, because of the large amount of high resolution image data. So the required cache capacity will be large and it also hopes to connect with the line bus CPU. Here is an effective approach to conditional random field. CRF is a probabilistic graphical model (PGM), in which CRF is discriminant and optimizes the posterior probability distribution of data labels directly on the basis of available observations. In this paper, the condition random field framework combined with spatial neighborhood information is used to cache remote sensing images. FCM is used to construct the first order potential of CRF, so as to achieve the independent assumption condition to avoid massive heterogeneous remote sensing data and reduce invalid pre-read cache. The spectral information and timing information of multi-source data are used as the second order potential of conditional random field, which can better use heterogeneous multi-source metadata to cache the image data needed by users in time. Gibbs energy function model of conditional random field is:

\[
E(Y | X) = - \sum_{i \in E} \phi_i (y_i; x_i) - \sum_{j \in E, i \in E} \phi_{ij} (y_i, y_j; x)
\]  

(1)
Where \( \phi_i(y_i; \chi_i) \) is the unitary potential, which uses the spatial information between data; \( \phi_j(y_j; y_j; X) \) represent binary potential, spectra of all data in the same and adjacent regions and other high-level semantic information are introduced.

As can be seen from figure 3, the whole process of image caching is to retrieve the original information according to semantics. Then it can locate the appropriate file, apply for access agent, write metadata and judge the cache state. After reading, the cache process is completed by recording the reading process in combination with CRF reasoning.

Maximum posterior probability maximum a posteriori is solved in the condition random field. MAP is equivalent to the Gibbs energy function minimization, as shown in equation 2.

\[
y^* = \arg \max_y p(y|x; \theta) = \arg \min_y E(y|x; \theta)
\]  

(2)

In case of diverse data sources and complex high-level semantics, it is difficult to solve, loopy belief propagation algorithm can be adopted to infer the optimal configuration by LBP.

5. Experiment analysis

The remote sensing data storage system using adaptive memory cache was tested in this paper, and the access experiment was carried out in the network environment with multi-source heterogeneous data sets. At the same time, an experimental comparison is conducted. Firstly, the performance of the storage system proposed in this paper is tested. Secondly, the read-write performance of the storage system that stores image files directly in HDFS and metadata in MySQL is tested.

5.1 Experimental environment and experimental data

The experimental hardware environment is listed in table 1:

| The server and software number | Parameters/version |
|--------------------------------|--------------------|
| The server                     | 8 DELL PowerEdge R730 |
| SATA hard disk                 | 1TB memory          |
| Network detector               | 2 Ac90-264v, 48-62hz input power supply |
| Switches                       | 1 Huawei S1700-24 gr |
| CentOS                         | 7.1                 |
| HDFS                           | 2.7.4               |
| PRESTO                         | 0.208               |
| ALLUXIO                        | 1.6.1               |
| SQL                            | 7.0                 |
| ORACLE SOLARIS                 | 32bit               |

The experimental data contains high score series data, among which the four types of security control data successively reach 2.3Gbyte, 10.2Gbyte, 20.6Gbyte and 40.5Gbyte.

5.2 Experimental results and analysis

Benchmark file system performance index and actual remote sensing data processing process time were compared in the experiment. The benchmark file test system USES IOZone with FUSE mounting mode and ASYNC_THROUGH storage system writing mode in this paper. From the test of the benchmark file system, it can be seen that the writing speed of the cache system is greatly improved compared with HDFS due to the presence of the cache system when writing small files. However, due to the introduction of additional cache layer during reading, the performance is sometimes slightly worse than HDFS. If considering the multi-user and multi-task situation in practical applications, the overall throughput rate is higher than HDFS. The test results are shown in figure 4.
For the writing of large remote sensing image files, the performance of the system in this paper is higher due to asynchronous cache when writing, but the data is not cached when reading, so the first reading needs to be performed from the underlying storage. However, due to the cache read-ahead strategy, the performance is higher than HDFS.

If there are many concurrent production tasks running in the system and the correlation between data is large, it is also proved in the test of the actual generation process. Therefore, the benchmark file test is of reference significance, but it cannot prove that the remote sensing image storage system based on adaptive cache is less effective than the traditional storage mode.

When the remote sensing image storage system proposed in this paper is applied in the actual production process, the larger the data correlation between the steps before and after the process, the higher the hit ratio of cache, and the more obvious the acceleration of the whole processing process by the storage system, which has more significant practical significance. At the same time, the storage system adopts distributed redundancy, so the security of data storage is guaranteed, and there is no data loss or inconsistency caused by node failure or network fault in practical application.

In actual production process of remote sensing data, though HDFS can be configured with a certain amount of hot data cache, it cannot be flexibly configured according to needs. Meanwhile, the single-task throughput of HDFS is relatively low due to the low performance of the hardware that builds HDFS. The remote sensing data storage system based on adaptive cache proposed in this paper can reasonably use the memory of each node to cache the intermediate data, without affecting the memory usage requirements of the processing module and accelerating the whole production process.

6. Conclusion
In this paper, the main contribution is to introduce a distributed adaptive cache policy as a remote sensing image of the store, through the use of the unity of the virtual file system, is compatible with the traditional way to store. Retain the existing data, and on the remote sensing data storage system based on HDFS significantly accelerated, actual production in the process of remote sensing data, relative to use the disk array production process, using storage system proposed in this paper the process of consumption time reduced by 18% to 35%. The experimental results show that the remote sensing data storage system using adaptive caching strategy proposed in this paper is effective in practical remote sensing data production, and the selection of Presto for metadata storage is reasonable.

The storage system has the characteristics of low construction cost, high reliability, and good performance. At the same time, the storage system also has some shortcomings. If the required data is not strongly correlated and the system does not record the historical data used by the user, the distribution is also distributed. When the cache is used, the system data transmission depends on the
speed of the network. If the infini band network is introduced, it is expected to greatly improve the overall performance of the system.

References
[1] C. Ping, "A second-order SQL injection detection method," 2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, 2017, pp. 1792-1796.
[2] LI H. Alluxio a virtual distributed file system[D]. Berke- ley UC Berkeley,2018.
[3] Li deren, zhang liangpei, xia guisong. Automatic analysis and data mining of remote sensing big data [J]. Journal of surveying and mapping, 2014,43, 12, 1211-1216.
[4] Zhu jianzhang, shi qiang, Chen feng-e, et al. Research status and development trend of remote sensing big data [J]. Chinese journal of image and graphics, 2016,21, 11, 1425-1439.
[5] Zhang Yang, xie bin, wang jingping, et al. Design of remote sensing image business management system based on Hadoop [J]. Computer system application, 2008, 27, 11, 64-70.
[6] Nie pei, Chen guangsheng, jing weipeng. A distributed storage method for remote sensing image [J]. Surveying and mapping engineering, 2008, 27, 11, 40-45.
[7] Jing weipeng, tian dongxue. Research on distributed storage and query method of remote sensing data based on HBase [J]. Computer engineering and science, 2008, 40, 6 992-998.
[8] Zhang lifu, Chen hao, sun xuejian, et al. Design of integrated storage structure of multidimensional remote sensing data spatio-temporal spectrum [J]. Journal of remote sensing, 2017,21, 1, 62-73.
[9] Li guoqing, huang zhenchun. Infrastructure integration, management and on-demand services of remote sensing big data [J]. Computer research and development, 2017, 54:267-283.
[10] Wen dongxin, dong wenjing, cao rui, et al. Asynchronous storage optimization based on Alluxio [J]. Journal of southeast university natural science edition, 2008, 48:248-252.
[11] Dong wenjing, wen dongxin, zhang zhan. Optimization of caching strategy based on Alluxio remote scenario [J]. Computer application research, 2008, 35 10 3025-3028.
[12] YILDIZ O, ZHOU A C, IBRAHIM S. Improving the ef- fectiveness of Burst Buffers for big data processing in HPC systems with eley [J]. Future Generation Computer Systems, 2018, 86(9)308-318.
[13] YANG C T, CHEN C J, TSAN Y T, et al. An implemen- tation of real -time air quality and influenza -like illness data storage and processing platform[J]. Computers in Human Behavior, 2018, 9(28)880-889.
[14] P. Giner, C. Cetina, J. Fons and V. Pelechano, "Presto: A pluggable platform for supporting user participation in smart workflows," 2009 6th Annual International Mobile and Ubiquitous Systems: Networking & Services, MobiQuitous, Toronto, ON, 2009, pp. 1-2.
[15] ZHANG G. JIA X Simplified conditional random fields with class boundary constraint for spectral-spatial based remote sensing image classification[J]. IEEE Geosci- ence and Remote Sensing Letters, 2012, 9(5)856-860