Research and Application on Optimization of Multi-thread Download Technology for Enhanced Search Engine

Yajuan Sun,Hong Lin
Network and Information Center
North China Electric Power University
Beijing, China
syj@ncepu.edu.cn

Baohui Wang
School of Software
Beijing University of Aeronautics and Astronautics
Beijing, China
wangbh@ncepu.edu.cn

***Abstract*—Multi-threaded file download as the key technology of content acquisition system for search engine, determines the efficiency and timeliness of content acquisition. In this paper, we do the research on optimization technologies which include multithreaded download based on P2SP, task scheduling based on MapReduce and download based on the protocol adaptation, designed to improve enhanced search engine efficiency. At last the result shows that the optimization method is successful for content acquisition.**

Keywords- search engine, multi-threaded download, P2SP, MapReduce, protocol adaptation

I. INTRODUCTION

The content acquisition is the basic work for search engine, with the rich type of network resources, which should support not only the collection of data such as text, web pages, also multimedia data acquisition. With the growing of huge amounts of data on Internet, the data collection must achieve real-time, which requires completing download of data in the possible shortest time, and as a data transmission technology, multi-thread download technology to efficiently complete the download tasks will directly affect the quality of the acquisition.

"Key technology research and demonstration for the enhanced search engine" is a subject for National Science and Technology Support Program, which number is 2011BAH11B01. The topic is to study the key technology as real-time multimedia information collection, security filters, audio and video index, results focus, collaborative recommendation, collaborative optimization enhanced search engine, in order to achieve multimedia information fusion, focusing, filtering, recommendation, supporting the three-screen integration services for mobile phone, computer and TV, supporting safe and reliable search service.

Content acquisition system of search engine is mainly responsible for collecting a variety of video information from the network, in which network crawler is responsible for downloading the files to the local server. After downloading, streaming media generally generates file format such as FLV, AVI, RMVB, and WMV. These files some are complete, some are stored as block on a different server, and there are several different mirrors for accelerate download for clients. File download system belongs to the content acquisition system as a sub module, generally which should have the ability to quickly complete a streaming download. In the cloud platform environment, the server is distributed deployed, Internet network environment is complex and changing, resources often occurs failure. File download system overall requires spending minimal cost, in the possible shortest time, to complete the fastest resource file downloading.

II. SYSTEM ARCHITECTURE FOR MULTI-THREADED FILE DOWNLOAD

Multithreaded download includes download queue management, download task management and download resource management. It is distributed structure as Master/Slave, which is divided into three parts, the control node and MongoDB storage node. Its architecture is shown as Fig. 1.

The main functions of the control node is a distribution of tasks, the task scheduling management and the coordination of download node using MapReduce model to performs scheduling. The main function of the download node is file download after the protocol adaptation. The MongoDB storage node is indexing and storage on the slice information, maintaining task information of the control node.

The dual backup of the control node can guarantee the reliability of the system. The MongDB storage nodes using a distributed deployment, constitutes a large storage index cluster [1].

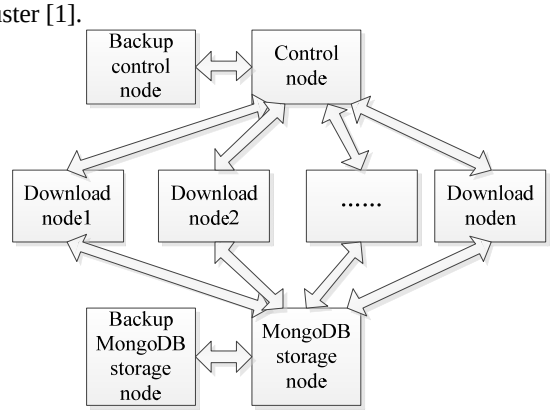

Figure 1. the system architecture of multithreaded download

III. MULTI-THREADED FILE DOWNLOAD SYSTEM OPTIMIZATION TECHNIQUES

The architecture of the whole system is designed and built by multi-threaded download as core technology, in this paper, we propose three kinds of optimization methods, the multithreaded download based on P2SP, the task scheduling based on MapReduce, the download based on protocol adaptation, to achieve the purpose of improving resource download speed by the combination of these methods and parameter configuration.

A. Multi-threaded download based on P2SP

In the file download system, in order to complete flow media downloads efficiently, we need to adopt the mechanism downloading one resource with multiple tasks and multiple servers at the same time, which is P2S mode. Multiple servers starting multiple threads at the same time to different resources server to download, can maximized use the current network environment, but the section data downloaded need to be integrated into the together, which is P2P mode. At last the data are transfered to the storage server, and ultimately to merge. This general download module is designed and developed based on P2SP mode [2]. System workflow based on P2SP mode is shown as Fig. 2:

The detailed steps are as follows:

1) File download executing end access the Task Manager by HTTP protocol to get the download task information t.

2) Peers get the Source Server address of the media according to the obtaining URL, ready to initiate a request, and send Join request to Tracker Server, receive task scheduling.

3) Tracker Server returns some Peer nodes with the adjacent file download location to this Peer according to request file downloading position of the Peer. Due to the presence of NAT, Tracker generally returns Peer nodes in the same LAN or with the public IP address.

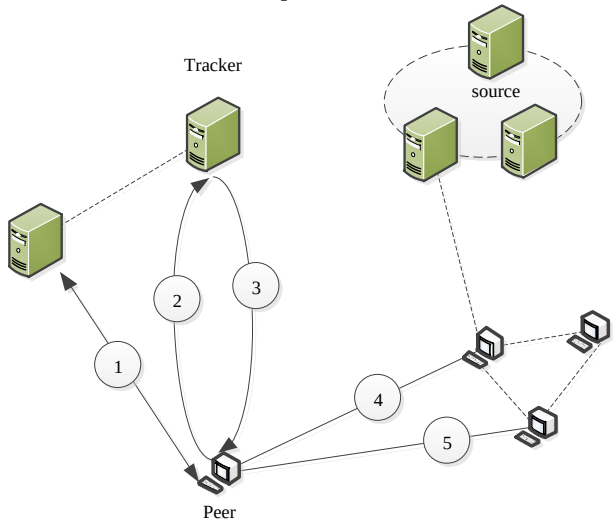

Figure 2. Workflow based on P2SP mode

4) Initiate a connection to a node in Peer List, create Partner relationship, obtain necessary SDP information and media length information to complete RTSP interacting from Partner node.

5) Send scheduling data request to the the Partner node according to Partner Buffer Map.

In the download process, each file download executing end needs to find one of their own peer group, or create a new peer group when not find a peer group. Because the services provided by each peer group are not all the same, we will usually organize the peers to the same purpose through a peer group.

B. MapReduce-based task scheduling

As the task management control node, MapReduce model performs tasks allocation (map task) and resources slice merger (reduce task) [3].

File download node is a large cluster composed by a large number of servers, and the specific implementation process of download is that loading into the system by MapReduce functions program, then completing the download of every file slice, cycling merging files, and eventually completing file merge. The detailed implementation process is shown as Fig. 3.

There are six steps for the detailed MapReduce process of file download.

1) User program get the size information of the file to be download according to the URL, and then as the input conditions, the file is cut into M pieces of fragmentation that size are between 16M - 64M (in Fig3, to be downloaded slice 0,to be downloaded slice 1, ...). These M pieces of data task are distributed to multiple download node servers in the cluster, waiting to start the download process.

2) The master as the master node of task management will assign the task to download node workers. The tasks include M map tasks and R reduce tasks, the master will find the idle worker based on the network status, thread priority, download node status, and assign map or reduce tasks to it.

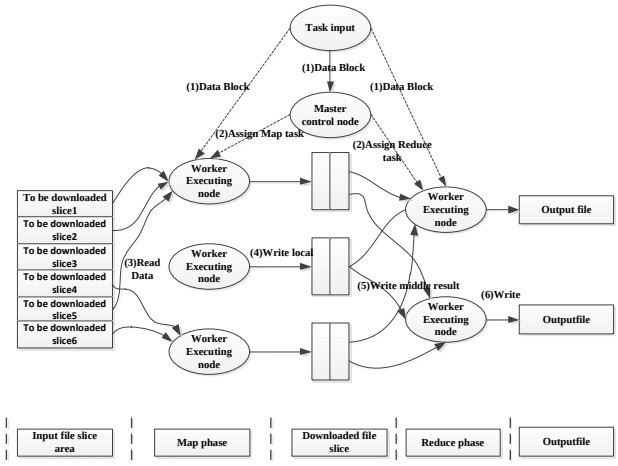

Figure 3. MapReduce task scheduling execution schematic diagram

3) When a worker is assigned to a map task, it will handle the file fragmentation assigned to. The data in InputData are also many pairs of Key/Value form, the worker of Map task will send these pairs to the Map function one by one, a lot of new Key/Value pairs produced by Map function will be cached in memory.

4) Those pairs cached in buffer will continue to be written to the local disk. Because There are R reduce tasks, the pair in disk will be divided into R files according to a certain rule, each district's data are assigned to a specific ReduceWorker. Intermediate files corresponding to each Map Worker, actually are divided into R number of files. The location of these areas will be passed to the master, which will be responsible for these locations to the reduce workers. You can see the master here plays the role of a nexus connection between the Map function and Reduce function which is completed by the master.

5) Reduce Worker get the location of these files slice from the master, then these Key /Value pairs are read from mapWorker by I/O. After finishing reading the data, it is necessary to sort these Key/Value pairs according to Key, getting different pairs together.

6) ReduceWorker conducts iteration of the whole sorting Key /Value data, sends each unique key and its corresponding value sequence to the Reduce function. The result of Reduce function for such input is 0 or a value (such as All Value additive, the sum is the final result). Reduce function will add slice downloaded to the final R output files.

After finishing the map and reduce task, master would call the data saving interface, the MapReduce call of user return, and store this R file information to a NoSQL database.

Here, master has played three roles, assigning the task to the worker, maintaining all the information and the status of the worker, telling the position of Map task results to Reduce task, playing the role of a bridge.

C. Download based on protocol adaptation

Transmission of data resources generally is via HTTP server, FTP server or streaming media server. Ordinary resources can directly use the HTTP or FTP protocol to download, but pure HTTP protocol to download streaming protocol-based resources, cannot accurately obtain resources to slice, which need to unpack it, and ultimately get download data according to the RTP protocol.

Real-time streaming transmission is different to HTTP streaming transmission, which requires a dedicated streaming media server and the transmission protocol. Since TCP requires a large cost, is not suitable for real-time data transmission. Protocol based on HTTP/TCP can be used to transmit control information, and use the RTP/UDP protocol to transmit the real-time streaming media data in the streaming data transmission scheme. The typical protocols for this transmission mode are RTP and RTCP protocol [4].

RTP is at the transport layer, which is built on UDP. Same as the UDP protocol, in order to achieve the transfer function, RTP also have the fixed form of a package. RTP is used to provide time information stream synchronization for end-to-end real-time transmission, but does not guarantee the quality of service. The quality of service is provided by RTCP, which is shown as Fig. 4.

When the application is to establish a RTP session, it will determine the destination transport address. Destination transport address consists of a network address and a pair of ports. There are two ports, one is for the RTP packet, the other is form RTCP packets, which makes the RTP / RTCP data can be sent correctly. RTP data is sent to the even UDP port, while the RTCP data of the control signal sent to the adjacent odd UDP port (even UDP port number+1), thus constituting a UDP port pairs.

RTSP is used to establish and control one or more time-synchronized, continuous audio and video media streaming session protocols. RTSP session command passed between client and server, it can be done such as request to play, start, pause, search, fast forward and rewind of the VCR control operations. RTSP session is usually hosted in a reliable TCP connection, but can also use connectionless protocol such as UDP to send RTSP session commands.

IV. DESIGN OF THE SYSTEM PHYSICAL ARCHITECTURE

Due to the massive crawled data by collecting, it needs distributed deployment for file download application servers.

Task management center should intelligently allocate download tasks according to the region of the source server, or the distance and network conditions, and added to the central database. When the file download execution end receives download instructions, it carries the information read from the central database by the download task, and then downloads the appropriate file fragment from the source server.

Using P2P mode between file download execution ends, they maintain contact to the task management center. When finishing slice data download, or when MapReduce task scheduling center is busy, it can conduct small piece of data merging in the internal and timely exchanging to a server, update the status of tasks in the NoSQL, then download other tasks, perform blockbuster data merging combined with MapReduce, eventually finish efficiently downloading.

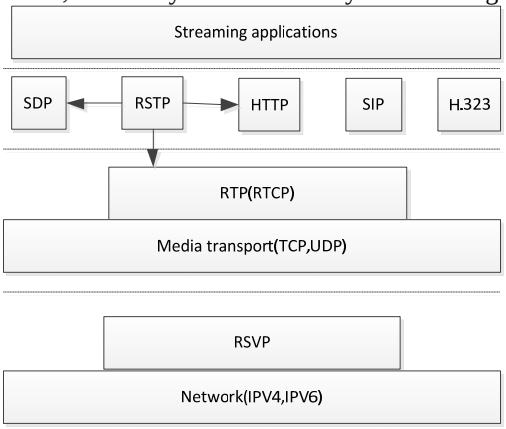

Figure 4. streaming architecture

The physical structure of the system is shown as Fig. 5.

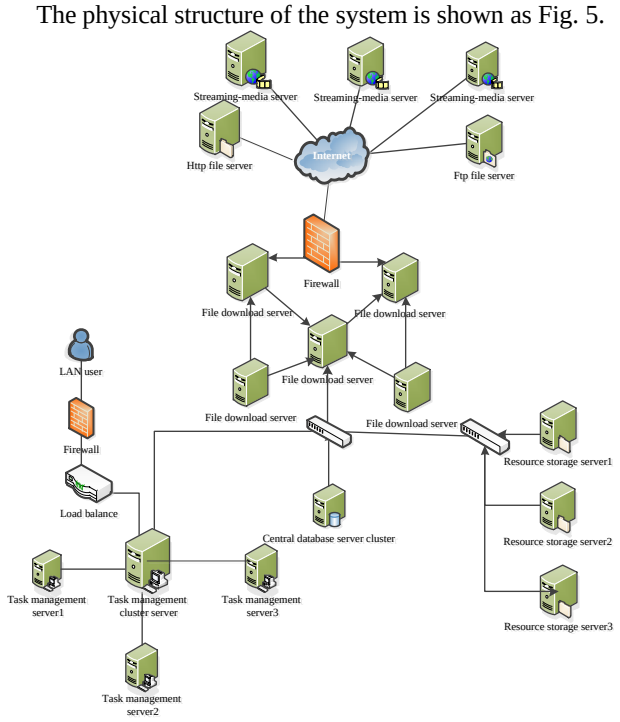

Figure 5. system physical architecture

V. CONCLUSION

According to the result of test, we can see a large number of slices were produce in the download process, and some tasks would switch to other standby server to download because of the network speed, at last successfully completed downloading. The main performance index of file download system is quantity of concurrent task processing and capacity of concurrent merging file, when the download task strength gradually increasing, since in this download system the same file can choose multiple source address to download, the network bandwidth can be used are more extensive, the total download speed are more quickly than other download tools. The optimization method in this paper for file download system in the process of collecting file can effectively improve the efficiency of the content acquisition in enhanced search engine.

REFERENCES

[1] G.L. Wang, “Application Research and program optimization for MongoDB database”, China Science and technology information, 2011, vol 20, pp. 93-94.

[2] F. Hou, “Download System Analysis Based on the P2SP Technology – Example as thunder”, Journal of Henan College of Finance & Taxatio, Nov. 2011, vol. 25, pp. 88-89.

[3] G.Z. Sun, F. Xiao, X. Xiong, “Study on Scheduling and fault tolerance mechanism of MapReduce”, Microelectronics and computer,2007, vol. 24, pp. 178-179.

[4] D. Hu, H. Liu, “The Video Technology on Internet Based on RTP/RTCP”, Computer Application Research, 2002, vol. 10, pp.140-142.