International Journal of Future Generation Communication and Networking
Vol. 8, No. 4 (2015), pp. 121-130
http://dx.doi.org/10.14257/ijfgcn.2015.8.4.12

Network Design and Implementation of Synchronization Software

Peng Hai-yun and Liu Xin

Department of computer science and technology, Zhou Kou Normal University Zhoukou,China
E-mail: hangfan_2007@163.com

Abstract

The actual demand for network applications, the paper discusses the data backup, real-time data synchronization knowledge; On this basis, considering the multi-user concurrency, the case of multi-user collaboration, to achieve a multi-machine real-time synchronization, proposed a network synchronization system model, describes its design ideas and development methods. The results show that: the system update for the relatively low intensity operations with relatively good efficiency of backup and synchronization synergy.

Keywords: *network storage, real-time backup, network synchronization.*

1. Introduction

It presents the exchange of information between international, intelligent and broadband trends among people through the computer in the 21st century. Local and remote information exchange, file interaction, we need more timely and reliable. Do you encounter such confusion when you hurried to the office computer from home only to find forgotten copy of important documents? To this end developed a web-based data replication network synchronization system that can guarantee the client complete consistency between the data and the way real-time backup to reduce the inconvenience of manual operation and failure risks. In addition, the system also implements real-time synchronization updates. To ensure that the file system level multiprocessor synchronization, improved workplace flexibility, allowing many people simultaneously file system to improve the parallel efficiency. It also means there are multiple file system backups, to further improve the data security and integrity.

2. The Main Mode of Data Backup

In recent years, the rapid development of network storage, remote off-site backup to ensure that when a server because of its damage, the data stored in the data center can quickly recover and become the preferred backup program. To ensure that the data in real-time, integrity and balance between system performance, disk-based remote data backup system have two main ways of working:

Synchronous data replication modes: updated data from the processor is written to the disk system before the local connection, through disk mirroring technology, it will update the data transmitted to the remote disk system.

Asynchronous data replication modes: updated data from the processor is first written to the local attached disk system, and immediately returned to the processor, an I / O write to complete instructions, then the disk mirroring system in a very short period of time, will be updated Data sent to the remote disk system.

ISSN: 2233-7857 IJFGCN

3. System Structure Design

The system uses C / S mode, each client sends to the local update server while remote from the server to download updates. Between the individual clients does not communicate directly. Server is responsible for synchronization between the various client relationships. Ensure that all the client file system consistency. System structure diagram is shown in figure 1.

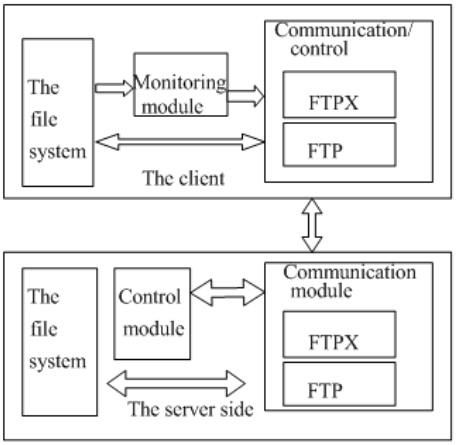

Figure 1. System Structure Diagram

The user on the client to interact with the file system, file system monitoring module monitors the change. If we found its change, we will put the change of the information sent to the communication / control module, communication / control module through information filtering, the information is organized into FTPX command format, sent to the server via FTP.

Server receives the client's order, process the order, and to make a selective response. The client makes the appropriate response in according to the server's response.

Clients are mainly monitoring module, FTPX, FTP client, operation simulation module, the global control module, test module (performance test).

Clients are composed by the FTP server module, the FTPX module, and the log module three components. Three modules completed the backup and collaboration capabilities. The structural chart is as following.

4. Module Implementation

4.1Server-side Processing FTP Server Implementation of Data Distribution

FTP server is the main component of the server. Other modules depend on the module and triggered by the FTP server. (Including the front of the user management module) FTP server is the message the command-driven mode, only when an order is sent to the monitor port, FTP server will start the appropriate services.

FTP server is implemented by CFtpServer, CFtpXQueue, which is a FTPX CFtpXQueue command queue. There is no response command processing functions, processing functions are member functions for the CFtpServer. Between the two classes are friend’s relationships. Now introduce two more important CFtpServer member functions:

Unsigned ClientShell (void *pvParam)

Void FtpXShell(char *pszCmdArg, Client Node *Client)

Client Shell primarily responsible read data from the monitor port; analyze the data header. If it is the FTP order, it will process in the internal function. And if it is FTPX order, it is submitted FtpXShell function to process.

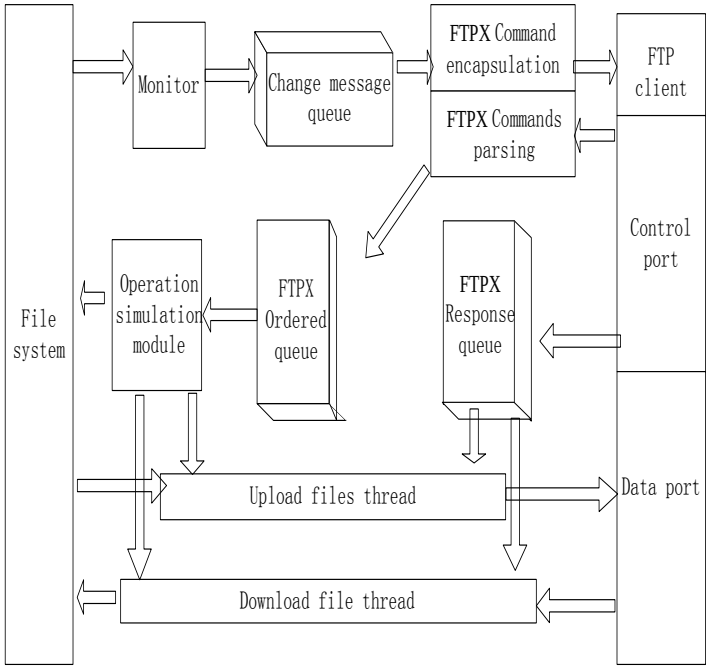

Figure 2. Server Structure Flowchart

To ensure the command reads the processing efficiency, these functions are designed to quickly return to the service routine, and some of the more time-consuming operation is placed in the other thread to do it.This can guarantee that the socket can be read into the buffer data read out in time to avoid the buffer is full and the packet loss.

Be distributed under the header, if not FTPX command, the FTP protocol in accordance with standards implementation.

4.2 Receive the File Server Thread Implementation

To receive files CFtpServer thread function is a member function:.

Unsigned StoreThread (void * pvParam).

When the FTP server receives the "STOR" command, it will start a new thread to run this function.

Initialized value of the variable is determined by the parameters of the function passed in the thread, by the force of the argument into a struct CFtpServer: ClientNode *, the structure records all the necessary information of all interactions with clients.

"All other clients send FTPX sync command", through forward, to the debit user management ClientNode traverse linked list implementation. The command content is just uploading the current success of the command to synchronize the files, notify other clients to download this update.

4.3The Flow Charts of Monitoring Thread

The functionality of the this thread as the main thread. Created for the pending state, need other threads (the main thread) to activate it. When it is activated, loop reads data

from the completion port queue, call subroutine distributed processing. At the same time, to order the current thread state, decide whether to continue to monitor. Because when reading data, calls GetQueuedCompletionStatus function, the function of blocking function, if there is no information in the queue, it will block until a complete change information. If the function returns without reading data, will go wrong. May be caused by file handle off. This thread is not necessary to continue to run, to return. The process is shown in figure 3.

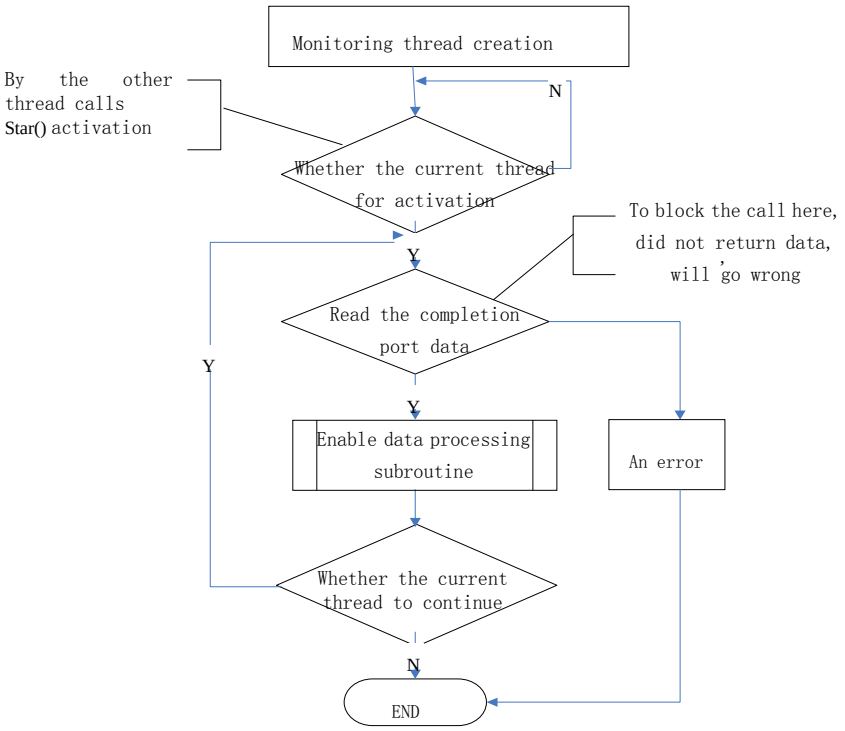

Figure 3. Flow Charts of Monitoring Thread

4.4 The Realization of the Monitoring Module

Monitoring module is mainly composed of ReadDirectoryChangesW API, I/O completion port model to implement.

```
BOOL ReadDirectoryChangesW(
  HANDLE hDirectory,       // handle to directory
  LPVOID lpBuffer,          // read results buffer
  DWORD nBufferLength,    // length of buffer
  BOOL bWatchSubtree,     // monitoring option
  DWORD dwNotifyFilter, // filter conditions
  LPDWORD lpBytesReturned,          // bytes returned
  LPOVERLAPPED lpOverlapped,    // overlapped buffer
  LPOVERLAPPED_COMPLETION_ROUTINE lpCompletionRoutine
// completion routine
);
```

ReadDirectoryChangesW is WINDOWS API, the function to obtain the change of the specified directory, divided into two kinds of synchronous and asynchronous mode, this system USES the asynchronous mode. But because of the API is to realize the limitations, when too many changes in the monitored directory, will cause the change of stored information buffer overflow [8]. Change the phenomenon of information loss.

I/O completion port model, is to create a queue, in a timely manner to changes in ReadDirectoryChangesW information read out, avoid the loss of change information.

In the implementation, the encapsulation of CDirectoryChangeWatcher and CDirectoryChangeHandler two main classes.

Implementation of other modules is not going to introduce.

5. Tested Analysis

5.1 An Experimental Environment

Experiment based on a client/server environment, to use hardware and software components as follows:

1)Hardware devices.

clients with Pentium (R) 4 CPU 2.60GHz, 512M RAM, 60G IDE hard drive, 100M card;.

server with Pentium (R) 4 CPU 2.60GHz, 512M RAM, 80G IDE hard drive, 100M card.

Experiments are 100M LAN network environment.

2)Software.

The work of the system platform is VC ++6.0, where the client runs on Windows XP SP2 operating system, server running on Windows XP SP2 operating system.Test module is installed on the client, it records in real time the transmission speed of the client, while marking the first time the update is complete.

5.2Experimental Methods and Results

5.2.1 Update Latency Test

The test will be the source machine synchronous machines of different types of updates to compare the response speed of the test.Specific experimental method is as follows:.

1)monitoring the source machine to a directory of files of different sizes, from 1M to 600M between.

2)the size of files for different types of operations.

3) test module records the transmission speed of each client.

4)to compare the various situations, synchronous machine with the source machine synchronization delay.

"Property" nature of the delay in updates,such as Table 1, "content" the nature of the delay in updating Table 2.Analysis:

It can be seen from Table 1, for different file sizes "property" nature of the response speed faster than the update. Because the synchronized updated data traffic is smaller

Table 1. “Property” Nature of the Delay in the Update Table

File size	Type of operation	Response delay(seconds)
0~4K	new	1~3
	modify	1~3
4K~10M	New	2~6
	Modify	2~6
	New	5~30

10M~100M	Modify	5~30
100M~600M	New	30~150
	modify	30~150

And it can be seen from Table 2, for different size documents, "contents" nature of the updates are very different response. Because the system uses the full backup, file sizes, the amount of data required for backup are also different.

In addition, with the number of clients' increases, the "Properties" change the nature of the response speed is relatively small update; and "content" the nature of the response speed was significantly slower updates. It was due to the server-side bandwidth.

Can be seen, the system response to small file synchronization more efficient. For larger files the efficiency of the synchronous response is not very satisfactory.

5.2.2 Data Transfer Rate Test

The test machine will synchronize the different machine the size of the source file "content" of the nature of the updated data transmission speed comparison test.Small files: 1M The following file size; large files: 100M or more the size of the file.Specific experimental method is as follows:

1)monitoring the source machine directory automatically generate a large file or small file 100.

2)test module records the transmission speed of each client.

3)to compare the various situations, synchronous machine with the source machine synchronous data transfer speed.

Table 2 "content" Nature of the Update Delay

File size	Type of operation.	Response delay (seconds).
0~4K	Rename.	0.95
	delete	1.98
	Property changes.	1.47
4K~10M	Rename	0.90
	Delete	2.25
	Property Changes	1.44
10M~100M	Rename	0.70
	Delete	2.94
	Property changes	1.75
100M~600M	Rename	1.05
	Delete	3.95
	Property changes	1.22

In only one source machine, a synchronous machine case, two clients can use all the bandwidth.

It can be seen from Figure 2, while the small file was transferring, it did not make full use of bandwidth, time data connection primarily used in the creation

and destruction.And it can be seen from Figure 3, while large file transferred, you can make full use of bandwidth.

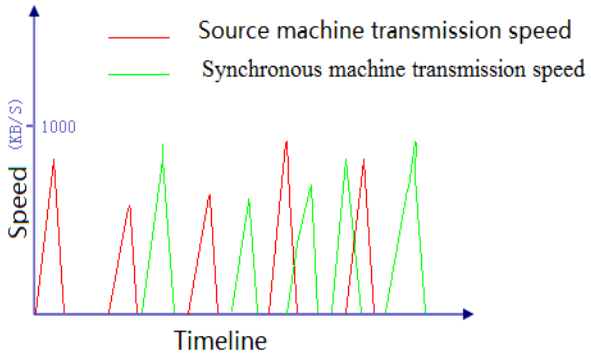

Figure 4. Client File Transfer Rate Curve

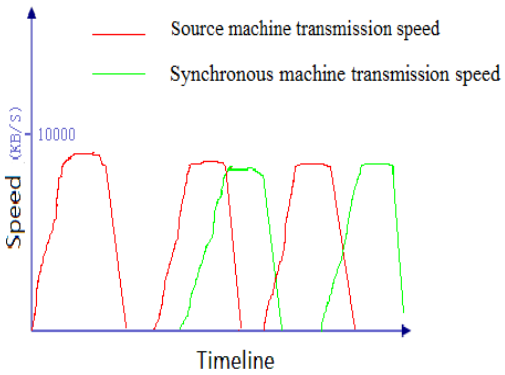

Figure 5. The Client Large File Transfer Rate Curve

When more than one machine simultaneously synchronized with the source confidential (the experimental environment for the two synchronous machines), the transmission rate of the client:.

From Figure 4, for small file transfers, there is sufficient bandwidth between the transmissions speeds of multiple synchronous

machines and the server side is not affected. The synchronized machines just have a slight difference at the beginning of transmission time.

For large file transfers, from Figure 5, more than the transmission speed of synchronous machine interaction. Due to the maximum bandwidth limitations; all of the transmission speed of synchronous machine and it can not exceed the maximum bandwidth.

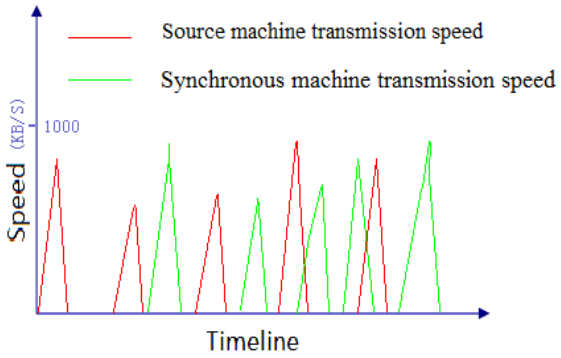

Figure 6. Two Synchronous Machines of Small File Transfer Rate Curve

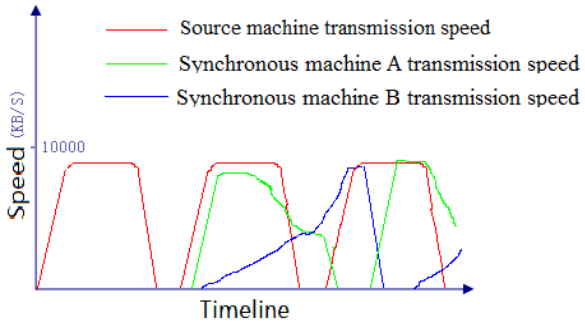

Figure 7. Two Large File Transfer Rate Synchronous Motor Curve

6. Conclusions

Application for the actual needs of the current network design and planning and the network synchronization software system construction and the content implementation; the paper determines the hierarchy of network synchronization system, topology, transmis- -sion mode, and simple synchronization protocols based on the synergistic way. Tests show that to synchronize the size of 1M the following data with the bandwidth in the 100M LAN network environment. This system has good performance.

Acknowledgement

THIS RESEARCH IS SUPPORTED HENAN PROVINCE NATURAL SCIENCE FOUNDATION (2010B520036).

THIS RESEARCH IS SUPPORTED NATURAL SCIENCE of CHINA FOUNDATION (61103143).

References

[1] B. Phillips, "Have Storage Area Networks Come of Age", Computer, vol.31, no.7, **(1998)**, pp.10-12.
[2] King, P. Richard, Halim, Nagui, Garcia-Molina, Hector, Polyzois and A. Christos, "Management of a remote backup copy for disaster recovery", ACM Transactions on Database Systems, vol.16, no.2, **(1991)**, pp.338-368.
[3] F. Wen-jun and Y. Yi-qing, "Computer Network Data Backup Management System", Journal of Hangzhou Institute of Electronic Engineering, vol.2, **(2001)**, pp.40-45.
[4] W. Yu-biao, R. Xi-ru and H. Pan, "Incremental database synchronization update mechanism under heterogeneous environment", Computer Engineering and Design, vol.32, no.3, **(2011)**, pp.948-951.
[5] J. Na-na and C. Dong-nian, "Research of synchronized updating of mapping in HBIA model", Computer Engineering and Design, vol.30, no.1, **(2009)**, pp.51-55.
[6] P. Hai-yun and Z. Wen-gang, "Based on WDM of Disk Immunity Systems", International Journal of Security and Its Applications, vol.8, no.2, **(2014)**, pp.659-669.
[7] Z. Wengang, Z. Yu and W. Feng, "Cancer Gene Clustering Algorithm Based on Quantum-Behaved Particle Swarm with Comprehensive Learning Strategy", Journal of Beijing University of Posts and Telecommunications, vol.37, no.4, **(2014)**, pp.59-63.
[8] Lloyd, J. Scott, Peckham, Joan, L. Jian; Yang, Q. Ken, "RORIB:An economic and efficient solution for real-time online remote information backup", Journal of Database Management, vol.14, no.3, **(2003)**, pp.56-73.
[9] L. Ya and P. Haiyun, "Based on the FTP network synchronization software realization of data distribution processing", Computer & Digital Engineering, vol.41, no.6, **(2013)**, pp.965-967.
[10] L. Li-Yu, H. Chua-Huang, "An integrated information system for real estate agency based on service-oriented architecture", Service-Oriented System Engineering, vol.52, **(2008)**, pp.184-189.
[11] L. Yu, "The technology of data copy", Journal of Yuzhou University (Natural Sciences Edition), vol.3, **(2003)**, pp.89-91.
[12] Q. Lv, P. Cao and E. Cohen, "Search and replication in unstructured peer-to-peer networks", rocessing of the 16th ACM International Conference on Supercomputing, ACM Press,**(2002)**, New York.
[13] F. Wen-jun and Y. Yi-qing, "Computer Network Data Backup Management System", Journal of Hangzhou Institute of Electronic Engineering, vol. 2, **(2001)**, pp.40-45.
[14] Yuanhong and Q. Wen, DATA SECURITY AND BACKUP, vol.2, **(2002)**, pp.31-33.
[15] W. Jing, "Notes on Data Backup Technology", Journal of Wuhan Engineering Institute, vol.12, **(2008)**, pp.23-26.
[16] X. Fu, T. Bultan and J. Su, "Synchronizability of conversations among web services", Software Engineering, vol.141, **(2005)**, pp.1042-1055.

Author

Peng Hai-Yun, she received the B.Eng degree in Computer science from Henan University and M.Eng degree in Computer science from Huazhong University of Science and Technology. She is currently researching on computer application technology.

Liu Xin, she received the B.Eng degree in Computer science from South-Central University For Nationalities and M.Eng degree in Control Science and Engineering from Huazhong University of Automation. She is currently researching on computer application technology.