Development of Wide Area Distributed Backup System by Using Agent Framework DASH

Yafei Zhou
dept. Computer Science
Nagoya Institute of Technology
Nagoya, Japan
cls14052@ict.nitech.ac.jp

Takahiro Uchiya
dept. Computer Science
Nagoya Institute of Technology
Nagoya, Japan
t-uchiya@nitech.ac.jp

Somayya Madakam
dept. Information Technology
FORE School of Management
New Delhi, INDIA
somayya@fsm.ac.in

***Abstract*—With the popularity of digital technology, digitization of data files has become extremely common, for ensuring safety and convenience of data storage, geographically dispersed and distributed backup systems are widely used for individuals. But individuals are sensitive to storage capacity in most situation, this research will adopt duplication exclusion technique, to avoid excessive storage of duplicate data, ultimately achieve savings in storage capacity. The effectiveness of proposal will be verified through experiment.**

***Index Terms*—backup, distributed, agent, CDC, storage**

I. Introduction

With the popularity of network, more and more individuals data is adopted digital storage, but digital data facing many risks, such as operation mistake, virus and hardware malfunction. For avoiding these risks, data backup is necessary, and for adding security, distributed backup systems are widely used.

But individuals usually require simplified operations and low resource occupation. In order to achieve this goal, agent technology is an effective solution. This research will adopt duplication exclusion technique to save storage capacity and use WHOIS retrieve service in order to dynamically select remote node.

II. DASH Agent Framework

This chapter will explain agent framework that used in this research. Agent technology is an integrated computer system that operates in a specific environment. Agent framework can provide components to assist developers build customized agent systems. In this research, DASH [1], Distributed Agent System based on Hybrid architecture, is adopted.

A. Overview

DASH includes two main parts: The management server, be responsible for generating and managing agents with repository; The agents and workplace, are responsible for processing specific tasks.

When user requests service, framework will send the request to management server, then, repository will be based on content of request to build agent, and to organize the agent that processes user's tasks on the workplace. The schematic diagram of DASH is Fig.1.

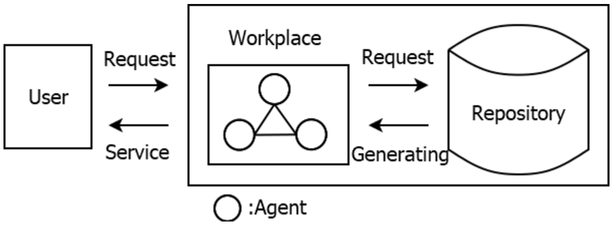

Fig. 1. Repository based agent framework.

B. DASH Agent

The DASH agent will be generated by specific description script, the processing module of agent will control the processing flow based on it. The environment in which agent works is called workplace, it can exchange information with other agent groups and process tasks based on the behavioral knowledge stored in each agent.

Based on the state change of the workplace, the agent will automatically process the task set by script.

C. Name Server

Name server is the management module of DASH, only one name server be used in DASH agent cluster. It storages two main parts information: environmental information and agent information.

III. Existing distributed backup system

This chapter will introduce some representative individuals data distributed backup systems.

A. Dropbox

The Dropbox [2], is a service that Dropbox, Inc. offers. It provides online storage space, user can transfer files using the network between remote server and local file system. In a free account, Dropbox storage space is 2 GB, and in a paid account, storage space is 1 TB.

B. CrashPlan

Crashplan [3],is free software that supports distributed backup for individuals. It provides redundant function, but this function is just the simplest redundant storage method, system will create data copy on each node of the cluster.

C. Problems

The third party distributed backup system only support a small amount of operations, users cannot adjust the details of the system according to their requirement, and it's often difficult for developers to adapt systems to different situations.

But adopting the DASH framework, it's possible to simply modify some specific agents to adapt to different requirements without affecting other parts of the system.

IV. Related Research

The related research [4], composed of three parts: distributed backup system, automatic backup start up mechanism and DASH integration. Because of new proposal based on the related research, so, this section will make the necessary explanations for the related research system and experimental results.

A. Distributed backup system

The backup process include encryption and redundant storage, the restore process include decryption and file combination, system flow will be explained in the following.

Backup:

- Getting resource information from server, to select redundancy rate and data transfer remote node, after that, encryption agent will encrypt file.
- File combination and redundancy agent will split file (the divided part is called block) to create redundant data.
- Transmission agent will transfer the processed data, and record backup information to system database.

Restore:

- Management agent will search requested blocks of system database, and transmission agent will transfer these blocks.
- File combination and redundancy agent will reorganize the file.
- Encryption agent will decrypt file and finish restore process.

The schematic diagram of system flow is Fig.2.

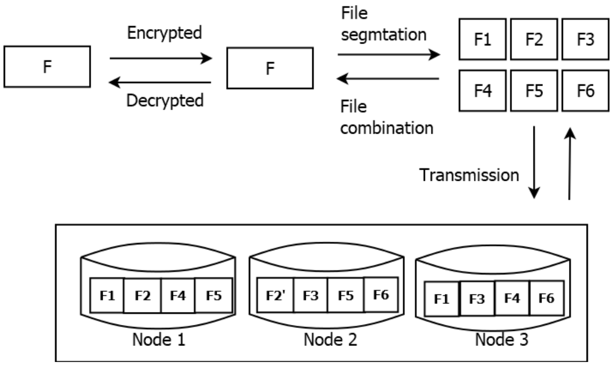

Fig. 2. The related system flow.

B. Problems

Based on the related research, this research will focus on improving the two parts:

In the related system, when files needs to be backed up again due to update, the adopted redundancy algorithm will re-back up the entire file, if some files are updated frequently, backup files will consume a lot of storage capacity.

On the other hand, the different DASH workplaces may exist on one node, and the order of candidate list just based on the simple rule. It means system will make unreasonable backup operations, such as sending all backup files to one node and selecting the node with the smallest storage capacity or the largest transmission delay.

For solving above problems, this research will import repeat exclusion mechanism and more flexible backup node selection mechanism.

V. Proposal

A. Duplicate exclusion

The basic thought is when a file that had been backed up (blocks of this files has been saved on multiple nodes) be updated, just update blocks which include updated content. In order to achieve this goal, proposal will adopt variable-length file segmentation algorithm and content verification mechanism. Other important changes on the original system will also be explained.

1) CDC [5]: For filtering updated blocks, proposal system introduces new dispersal algorithm, called CDC (Content Defined Chunking), this algorithm determines blocks based on content fingerprint (Hash implementation), when some parts of original file be changed, the CDC algorithm can find them, and can keep other blocks are not affected. The database will storage content fingerprint.

The Hash calculation adopted K&R algorithm [6], to improve the efficiency of CDC algorithm.

2) The specific implementation of CDC: In actual situations, if no additional restrictions on CDC algorithm, the results of the CDC algorithm are often unsatisfactory.

The three important parameters are: minimum length, maximum length and window size. The minimum length and maximum length can avoid blocking too large or too small, because of it's hard to predict the Hash match frequency of CDC, and window size will influence Hash match frequency. In most of situations, avoiding the matching frequency be too high or too low is important.

The above parameters were tested in the common range, and combined reference [7], the parameter settings are as follows:

- Minimum length: 4 MB
- Maximum length: unlimited
- Window size: 32 Byte

3) Combined with the original system: The first update is introducing the duplicate exclusion mechanism into the backup process, in the backup process of section IV.A, the duplication exclusion process will be executed between steps two and three. There are two different situations in duplication exclusion process, if data have not backed up, then upload all data directly, Otherwise, to process duplication exclusion first, upload necessary data afterwards.

After that, only changed blocks what will be transported. And the encryption process is changed to after duplicate execution. The reason will make explanation in the following.

The other part is the block allocation method, because of dispersion algorithm changed, the number of blocks will far exceed the number of nodes, the original function is expired.

So the new process will be adopted: Generating blocks first, and using every block number modulo number of nodes, the remainder is the node number that block will be saved.

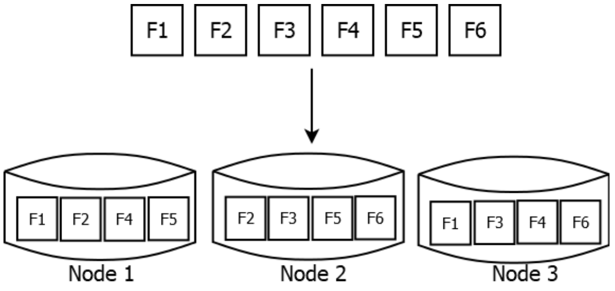

Fig. 3. Blocks allocation method.

In Fig.3, source file is divided into six blocks and there are three backup nodes, left two blocks in every node are based on remainder, the right two blocks are redundant storage, in this example, to select any two nodes will complete file recovery.

Because of encryption process will change the segmentation result, and blocks allocation based on segmentation result, so, the encryption process must be completed after the CDC algorithm.

B. Backup node selection

For avoiding unreasonable backup node selection, the WHOIS [8] retrieve service was introduced into the system to support backup node selection.

1) Google geocoder [9]: Based on the judgment of node position, system can use Google geocoder to calculate distance between local and remote node. WHOIS can get the node location name by IP address, and use this name can get it's latitude and longitude by Google geocoder, after that, the distance from the node can be calculated by the spherical trigonometry method [10].

2) Backup node selection scheme: The backup node selection scheme is based on distance, number of times the node has been backed up and remaining capacity of the node, the detail is following:

- Determining the number of backup nodes required, recorded as n .
- Calculating the distance between the local and other nodes, and selecting remote nodes to avoid storage in too close nodes.
- Selecting at least n nodes with fewer backups times. If candidate node more than n, then, further screening by distance.
- If candidate node less than n, then from the nodes with more backups, to select nodes with a large remaining storage capacity.

C. DASH agents configuration

For managing and executing functions of proposal, DASH will provide four main agents:

- Split/Joint Agent: It will surveillance the workplace, when new files move or modify in the workplace, it will read these files data and create blocks of these files. When some blocks will move into local workplace from remote nodes, this agent will stitch these blocks to restore file. The functionality of CDC is integrated in this agent.
- Duplication Agent: When new file's blocks will be uploaded, duplication agent will search the file backup message in the database, if it not existed, all blocks will be uploaded, otherwise, this agent will process these blocks to remove the duplicate part.
- Encrypt/Decrypted Agent: To encrypt upload blocks or decrypt restore blocks.
- Transfer Agent: This agent will select the appropriate remote backup nodes dynamically, and decide how to allocate redundant data. The functionality of backup node selection and blocks allocations method are integrated in this agent.

D. System Overview

Merging duplication exclusion and redundant storage mechanism, can give the overall flow chart of the system in Fig.4.

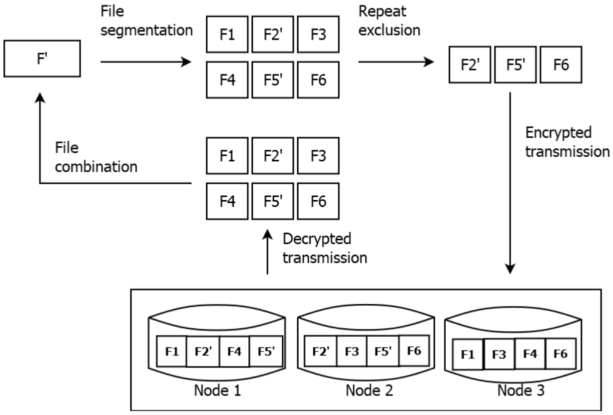

Fig. 4. The system flow.

The other parts of Backup/Restore process that not explained in this chapter are same as related research.

VI. Experiment and evaluation

A. Evaluation of duplicate exclusion

The experiment of duplicate exclusion focus on duplicate exclusion rate and backup time.

The tested file detail is in the TABLE I.

The duplicate exclusion experiment adopted local a computer, the detail of it in the follows:

TABLE I
TESTED FILES

File size	Original file	Changed content
about 500KB	pptx file with 15 slides	5 new slides be added
about 10MB	about 1 minute MP4 file	new sound track be added
about 100MB	about 25 minute MP4 file	new sound track be added
about 1GB	about 25 minute MP4 file	new sound track be added

Local computer:

- OS: Windows 7 Home Premium 64 bit
- CPU: Intel(R) i7-3770K CPU @3.5 GHz
- Memory: 8GB

The experiment just test the duplicate exclusion algorithm's effect and efficiency, in the first time, no any backed up blocks in nodes, and the next backup will use the changed file. The duplicate exclusion rate is the number of not changed blocks divide number of all blocks after second backup process completed, the result in TABLE II.

TABLE II
DUPLICATION EXCLUSION RATE(%)

500KB	10MB	100MB	1GB
65.7%	53.1%	25.3%	78.1%

And the result of time requirement is TABLE III.

TABLE III
TIME REQUIREMENT(S)

	500KB	10MB	100MB	1GB
Related research	1.42	3.08	19.07	202.51
First time	3.07	4.94	22.93	260.27
Second time	5.61	5.95	20.69	182.88

The result shows the effectiveness of duplication exclusion mechanism on duplicate exclusion rate, due to the introduction of additional processing, the time requirement increased.

But starting from the second time, the related system will also use same time as first, and the proposal system will save time. The local experiment ignores network transmission time, the more significant experiment results will be shown in system overall experiment.

B. System overall experiment

The experiment of entire system adopts same files on the above, the backup requirement node is 2 and backup nodes are changed to remote nodes, the other conditions keep same. The remote nodes arrangement is in TABLE IV.

TABLE IV
REMOTE NODES ARRANGEMENT

	Available HDD Capacity	Distance
Remote Computer 1	1730GB	510.5km
Remote Computer 2	52GB	347.5km
Remote Computer 3	773GB	0

The node selection result in TABLE V.

TABLE V
NODE SELECTION

First	Remote Computer 1
Second	Remote Computer 2
Third	Remote Computer 1
Fourth	Remote Computer 2

The result is in line with expectations, the node selection mechanism works fine, and the time required is as TABLE VI.

TABLE VI
TIME REQUIREMENT(S)

	500KB	10MB	100MB	1GB
Related research	2.82	10.27	81.66	885.90
First time	2.77	9.83	83.34	894.91
Second time	2.12	6.46	60.70	699.36

In the first time backup, the proposal system basically consistent with the results of the related research, but in second time backup, the duplication exclusion mechanism played a significant role, in many case, the backup time is shorted significantly.

VII. CONCLUSION

Based on the experiment result, it can be confirmed that the proposal is effective with the improvement on the original system. Using the duplication exclusion mechanism, after the file has been updated multiple times, the system can save a lot of time and storage capacity, and node selection mechanism can also avoid backup data centralized storage.

In the future work, reducing memory usage, adopting distributed database and improving system stability are main tasks.

REFERENCES

[1] Takahiro Uchiya, Hideki Hara, Kenji Sugawara, Tetsuo Kinoshita, "Repository-Based Multiagent Faramework for Developing Agent Systems", Transdisciplinary Advancements in Cognitive Mechanisms and Human Information Processing, IGI Global, Ch.4, pp.60-79, 2011.

[2] Dropbox, https://www.dropbox.com/

[3] Online Data Backup - Offsite, Onsite, Cloud - Crashplan Backup Software, http://www.code42.com/crashplan/

[4] Motohiro Shibakawa, Takahiro Uchiya, Ichi Takumi, Tetsuo Kinoshita, "Development of the Distributed Backup System using Multi-Agent Technology", ICCI*CC2013, pp.268-273, 2013.

[5] Athicha Muthitacharoen et al., "A Low-bandwidth Network File System", ACM SIGOPS Operating Systems Review, Vol.35, No.5, pp.174-187, 2001.

[6] Karp,R.M. et al., "Efficient randomized pattern-matching algorithms", IBM Journal of Research and Development, pp.249-260, 1981.

[7] Mie Ogiwara, Mizuki Takaya, Teruhisa Kasuya, Itaru Koikem, Toshiyuki Kinoshita, "Singularity Size Optimization in Data Deduplication", MPS, 2014-MPS-99(2), pp.1-4, 2014.

[8] "JPRS WHOIS / JPRS", http://whois.jprs.jp/

[9] "Google Code Archive", https://code.google.com/archive/p/geocoder-java/

[10] "Spherical trigonometry",
https://en.wikipedia.org/wiki/Spherical_trigonometry