FC-SAN 中的数据放置和访问路径选择的代价模型*

李 超+, 周立柱, 邢春晓

(清华大学 计算机科学与技术系,北京 100084)

A Cost Model for Data Placement and Access Path Selection Problem in FC-SAN

LI Chao+, ZHOU Li-Zhu, XING Chun-Xiao

(Department of Computer Science and Technology, Tsinghua University, Beijing 10084, China)

+ Corresponding author: Phn: 86-10-62789150, E-mail: li_chao00@mails.tsinghua.edu.cn, http://dbgroup.cs.tsinghua.edu.cn

Received 2002-11-15; Accepted 2003-09-09

Li C, Zhou LZ, Xing CX. A cost model for data placement and access path selection problem in FC-SAN. *Journal of Software*, 2004,15(5):741~751.

http://www.jos.org.cn/1000-9825/15/741.htm

Abstract: Network storage methods (e.g. FC-SAN) with virtual storage technology are becoming a powerful substitution of DAS in digital libraries and other massive storage applications. However the efficiency of FC-SAN virtual storage strongly depends on some attributes of the stored documents. In some cases, FC-SAN may even perform no better than the sharing data on LAN. This paper illustrates this point by a study on the data placement and access path selection issues in a network storage environment. The paper first presents a linear time-consuming model of data access through the analysis of the virtual storage principle and then gives a decision-making method for data placement and access path selection problem. In the development of this method, the concept of equivalent of virtual storage cost is defined to evaluate the data placement cost in the FC-SAN virtual storage environment. Finally, the theoretical analysis, methods, and assumptions are proved by the experimental results based on a massive storage subsystem in a digital library prototype.

Key words: networked storage; FC-SAN; virtual storage; data placement; access path selection

摘 要: 网络化存储通过引入网络的概念将存储独立于服务器甚至通信网络,已经成为传统存储方式的有力替代者.然而,FC-SAN 虚拟存储方式的存储性能依赖于存储对象的某些属性,在某些情况下,其性能甚至不如传统的 LAN 数据共享方式.就 FC-SAN 虚拟存储方式中的数据放置和访问路径选择对这一问题进行了研究.首先通过分析虚拟存储原理提出了一个数据访问耗时的线性模型;然后,就数据放置和访问路径选择提出了一个决策方法;并在进

* Supported by the National Natural Science Foundation of China under Grant No.60221120146 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.G1999032704 (国家重点基础研究发展规划(973))

LI Chao was born in 1978. She is a Ph.D. candidate at the Department of Computer Science and Technology, Tsinghua University. Her current research interests include database, massive storage systems, and digital library. **ZHOU Li-Zhu** was born in 1947. He is a professor and doctoral supervisor at the Department of Computer Science and Technology, Tsinghua University. His research areas are database technology, digital library, massive storage systems, and data warehouses. **XING Chun-Xiao** was born in 1967. He is a professor at the Department of Computer Science and Technology, Tsinghua University. His research areas are database, digital library, distributed multimedia systems, and digital rights management.

一步探讨这一方法的过程中,定义了“虚拟存储代价当量”的概念,用以评价 FC-SAN 虚拟存储环境中的数据放置的代价,从而为评价以及如何选择数据放置和访问路径提供了一种定量的手段.最后,在数字图书馆的一个海量存储原型系统中对上述的理论分析、各种条件进行了实验验证,并结合实际给出了“虚拟存储代价当量”的计算方法,验证了所提出的方法的有效性.

关键词: 网络化存储;FC-SAN;虚拟存储;数据放置;访问路径选择

中图法分类号: TP333 文献标识码: A

1 Introduction

Digital library is a data-intensive application that includes massive documents of various types (text, video, audio, picture, etc.), so a massive storage system is the key component for large-scale digital libraries. The development of digital library leads to the dramatically increasing requirements for massive storage systems, i.e. scalability, reliability, security, high-availability, and efficiency and virtualization of the management. However, the traditional storage method DAS (directly attached storage) is server-centric, and this brings so many limitations that it can no longer meet these requirements of the massive storage system.

By introducing the concept of networks, network storage makes storage independent of servers or even communication networks and becomes the powerful substitution of the traditional storage method DAS[1,2]. There are three main network storage methods nowadays: NAS (network attached storage), FC-SAN (storage area network based on Fibre channel) and IP-SAN[3] (storage area network based on IP storage technologies such as iSCSI, FCIP, iFCP). But today, FC-SAN has many advantages that the other two could not be compared with, e.g. high-speed data transmission, great scalability and flexibility, high-availability, centralized management, massive data access, LAN-Free Backup, remote mirror, and disaster recovery, so FC-SAN still takes up the high-end market of the networked storage. Many large organizations, in possession of massive business data, have chosen FC-SAN, e.g. eBay[4], Old Dominion University[5], and Carolina Power & Light[6].

As any kinds of servers or storage devices can be involved in a FC-SAN environment in which any server has an access to any storage device. In order to utilize the potentials of FC-SANs we need to solve the heterogeneous shared storage problem. It is believed that standardization, interoperability, and openness are vital to the success of the industry[7]. This commitment has led industry and research to focusing on developing virtual storage systems to run on top of as many existing standards as possible by using only standard extensions and interfaces to this existing technology. There are several such kind of systems which are very similar to each other, such as HP’s DirectNFS, Tivoli’s SANergy, EMC’s Celerra and Storage Tank. There are also other related work, e.g. Veritas Cluster File System.

Obviously, there are at least two strong points of this virtual storage solution for FC-SANs: (1) we can manage and utilize FC-SANs without a long time waiting, by leveraging the popular existing operating systems and file systems; (2) the storage resource usage of FC-SANs is transparent to end users so that they can manage them without any special trainings, and all the tools and applications can run smoothly without any modifications or any incompatible worries. However, how to efficiently utilize the FC-SANs in applications with this virtual storage solution is still an open problem. This paper focuses on the data placement and access path selection in this storage environment, which is the first thing to considerate in any storage environment deployment.

In developing digital libraries supported by FC-SAN virtual storage systems, its dominant easy-to-manage and easy-to-use feature of virtual storage often leads to the tendency to deploy it to achieve a better performance in the large-scale digital library. In fact, this practice strongly depends on some attributes of the stored documents. In some cases, FC-SAN may perform no better than LAN storage. As a result, for a storage system of the coexisting

FC-SAN and DAS, as documents may vary dramatically, it is desirable to select the most proper one according to some criteria. This paper illustrates this point by a study on the data placement and access path selection issue in the network storage environment.

The paper first presents a linear time-consuming model of data access through the analysis of the virtual storage principle and then gives a decision-making method for data placement and access path selection problem. In the development of this method, the concept of equivalent of virtual storage cost is defined to evaluate the data placement cost of the FC-SAN virtual storage environment. After that, all the theoretical analysis and assumptions are proved by the experimental results getting from one ordinary FC-SAN virtual storage environment. This environment is the core of the massive storage system prototype in our digital library project. Finally, we discuss the future work and draw the conclusions.

2 FC-SAN Virtual Storage Basics

Before introduction of the principle, some basics about FC-SAN environment should be talked about[8].

In Fig.1, the thick black curves represent FC connections and the thin gray curves represent LAN connections. The FC-SAN cloud means various structures composed of FC-switches, FC-hubs, kinds of bridges, and FC connections. The Ethernet cloud has the same meaning as the clouds in common LAN pictures. The storage devices, servers (shown as Host and MDC), and FC-SAN cloud constitute the FC-SAN, and the servers have both FC and LAN connections.

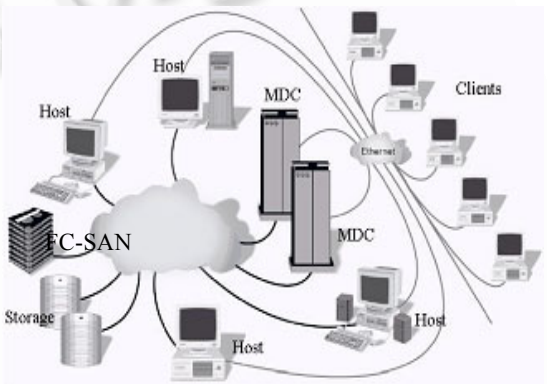

Fig.1 An example of FC-SAN environment

In FC-SAN virtual storage environments, metadata means the mapping information between the physical storage and the logical virtual storage of the storage resources. Servers fall into two categories according to their different roles, i.e. Host and MDC (meta data controller). The servers being destined for MDCs take charge of the storage and management of the entire or part of the metadata for the storage resources, and the other servers are all hosts who have no responsibility for the metadata.

So when hosts desire to access to the storage resources, before they can access the real data through the FC connections, they have to request the corresponding metadata information from the right MDC through the LAN connections first. Meanwhile, in the case of MDCs, if the corresponding metadata is on the MDC itself, the MDC can request the metadata information from itself and then access the real data through the FC connections. But if unfortunately the corresponding metadata is not on the MDC itself, then the MDC should act just as a host does.

3 Theoretical Analysis of FC-SAN Virtual Storage

3.1 Mathematical analysis of data access

From the description of Section 2, the total time interval of one data access from host, T_1, which is from the very beginning when host issues the data request to the very end when host finishes the data access, can be divided into three parts. The first part is the time host getting the metadata from MDC; the second part is the time for dealing with the metadata; and the third part is the time for the access to the real data through FC connections. So T_1 can be expressed as:

$$T_1=t_{Host\ gets\ metadata}+t_{deals\ with\ metadata}+t_{access\ data\ through\ FC\ connections} \quad (1)$$

The first item of Eq.(1) can again be subdivided into three parts. The first part is the time host issues metadata request to MDC; the second part is the time MDC searches for the right metadata; and the third part is the time MDC returns the metadata to host. Eq.(2) shows this subdivision:

$$t_{Host\ gets\ metadata}=t_{Host\ requests\ metadata}+t_{MDC\ searches\ metadata}+t_{MDC\ returns\ metadata} \quad (2)$$

The second item of Eq.(1) can again be subdivided into two parts. The first part is the time the host resolves the metadata, and the second part is the time it issues the data request to the corresponding storage resource. Eq.(3) shows this subdivision:

$$t_{deals\ with\ metadata}=t_{resolves\ metadata}+t_{issues\ real\ data\ request} \quad (3)$$

By replacing the first two items of Eq.(1) with Eqs.(2) and (3), we get T_1 with six sub-items:

$$T_1=t_{Host\ requests\ metadata}+t_{MDC\ searches\ metadata}+t_{MDC\ returns\ metadata}+t_{resolves\ metadata}+ t_{issues\ real\ data\ request}+t_{access\ data\ through\ FC\ connections} \quad (4)$$

From the description of Section 2, analogously, the total time interval of one data access from MDC, T_2, which is from the very beginning when MDC issues the data request to the very end when MDC finishes the data access, can be divided into three parts. The first part is the time MDC searches for the right metadata, the second part is the time MDC deals with the metadata, and the third part is the time MDC accesses to the real data through FC connections. So T_2 can be expressed as the following Eq.(5):

$$T_2=t_{MDC\ searches\ metadata}+t_{deals\ with\ metadata}+t_{access\ data\ through\ FC\ connections} \quad (5)$$

By replace the second item of Eq.(5) with Eq.(3), we get T_2 which is constituted by four sub-items:

$$T_2=t_{MDC\ searches\ metadata}+t_{resolves\ metadata}+t_{issues\ real\ data\ request}+t_{access\ data\ through\ FC\ connections} \quad (6)$$

Comparing Eq.(4) with Eq.(6), we get Eq.(7), i.e. T_1 is constituted by two more sub-items than T_2: the time the host issues metadata request to MDC, and the time MDC returns the metadata to it.

$$T_1=T_2+t_{Host\ requests\ metadata}+t_{MDC\ returns\ metadata} \quad (7)$$

In the above deduction, Eq.(3) has replaced the item *tdeals* with metadata for twice. This action is taken based on the assumption that the time the host resolves metadata and the time the host issues data request to the corresponding storage resource are equal to those of MDC.

This assumption is proved by the test results later. Besides, we can also prove that the items $t_{resolves\ metadata}$, $t_{issues\ real\ data\ request}$, $t_{MDC\ searches\ metadata}$, and $t_{Host\ requests\ metadata}$ can all be treated as tiny constants compared to the items $t_{MDC\ returns\ metadata}$ and $t_{access\ data\ through\ FC\ connections}$. We rename the tiny constants as Δ_1, Δ_2, Δ_3, Δ_4 respectively.

We provide a method to prove the tiny constants Δ_1, Δ_2, Δ_3, Δ_4 as follows: in the steady transmission rate range of both FC and LAN connections, measure the values of T_1 and T_2 with different data file sizes. Name the variety -- data file size as L_{DATA}. If T_1 and T_2 are increasing linearly with L_{DATA}, then the assumption of Δ_1, Δ_2, Δ_3, Δ_4 is proved true. The following is the principle of the above method.

IF the assumption of Δ_1, Δ_2, Δ_3, Δ_4 is true, then

$$T_1=\sum_{i=1}^{4}\Delta_i+t_{MDC\ returns\ metadata}+t_{access\ data\ through\ FC\ connections}$$

$$T_2=\sum_{i=1}^{3}\Delta_i+t_{access\ data\ through\ FC\ connections}$$

In the steady transmission rate range of both FC and LAN connections, name the transmission rates of FC and LAN connections as V_{SAN} and V_{LAN} respectively. Name the fixed size of metadata as constant L_{MD}. Then we get the following:

$$T_1=\sum_{i=1}^{4}\Delta_i+L_{MD}/V_{LAN}+L_{DATA}/V_{SAN}, \text{ and } T_2=\sum_{i=1}^{3}\Delta_i+L_{DATA}/V_{SAN}.$$

Name the following constants C_1, C_2 and a:

$$C_1=\sum_{i=1}^{4}\Delta_i+L_{MD}/V_{LAN},\ C_2=\sum_{i=1}^{3}\Delta_i,\ \text{and}\ a=1/V_{SAN}$$

then T_1 and T_2 can be expressed as follows, viz., T_1 and T_2 are increasing linearly with L_{DATA} in the steady transmission rate range of both FC and LAN connections:

$$T_1=aL_{DATA}+C_1,\ \text{and}\ T_2=aL_{DATA}+C_2.$$

3.2 Access path selection analysis

For MDC, there is only one data access path – through FC connections straightly. So, under the condition that the assumption of the tiny constants $\Delta_1, \Delta_2, \Delta_3, \Delta_4$ is proved true, our attention can be focused on the effect of L_{MD} and L_{DATA} on T_1.

First of all, $V_{SAN}>>V_{LAN}$ is taken as a fact (approximately 100 times), then the factors influencing T_1 mostly are named together as T_1', and $T_1'=L_{DATA}/V_{SAN}+L_{MD}/V_{LAN}$.

In normal circumstances, $L_{DATA}>>L_{MD}$, and then

$$T_1'=L_{DATA}/V_{SAN}+L_{MD}/V_{LAN}<<L_{DATA}/V_{SAN}+L_{DATA}/V_{LAN}<<2L_{DATA}/V_{LAN}.$$

But in some unnormal circumstances, the file size is so small that $L_{DATA}<L_{MD}$, and then

$$T_1'=L_{DATA}/V_{SAN}+L_{MD}/V_{LAN}>L_{DATA}/V_{SAN}+L_{DATA}/V_{LAN}>L_{DATA}/V_{LAN}.$$

This means when $L_{DATA}<L_{MD}$, access data through FC connections from host will be more expensive than those through LAN connections. In this case, it is a good idea to change the data access path from FC to LAN. Changing the data access path of host from FC to LAN follows this way: first, one host issues a data request to the corresponding MDC through the LAN connections; second, the MDC accesses the data as it usually does; and finally, the MDC returns the data to the host through the LAN connections.

In this way, we name the total data access time interval as T_3. By imitating the deduction way we get Eq.(7), we can get Eq.(8):

$$T_3=T_2+t_{Host\ requests\ data}+t_{MDC\ returns\ data\ to\ Host\ through\ LAN} \tag{8}$$

Comparing Eq.(7) with Eq.(8) with the assumption that $t_{Host\ requests\ data}=t_{Host\ requests\ metadata}$, we can get Eq.(9):

$$T_3-T_1=t_{MDC\ returns\ data\ to\ Host\ through\ LAN}-t_{MDC\ returns\ metadata} \tag{9}$$

Ask for $T_3<T_1$, i.e. $T_3-T_1<0$, from Eq.(9), we can get the following:

$$t_{MDC\ returns\ data\ to\ Host\ through\ LAN}-t_{MDC\ returns\ metadata}<0$$

$$\Rightarrow L_{DATA}/V_{LAN}-L_{MD}/V_{LAN}<0$$

$$\Rightarrow L_{DATA}<L_{MD}$$

From this result, we can conclude that in the case of $L_{DATA}<L_{MD}$, access to data from host through LAN connections relaying by the MDC is more time saving than directly through FC connections.

We can draw two conclusions here from the above analysis: (1) From Eqs.(7) and (8), we always get $T_2<T_1$ and $T_2<T_3$; so for the most frequent access data we should place the corresponding metadata locally, viz., destine MDC on itself. (2) As for Hosts, when $L_{DATA}<L_{MD}$, take the way of accessing data through LAN connections relaying by MDC; and when $L_{DATA}>L_{MD}$, just directly through FC connections. These can be used as a decision-making direction for data placement and access path selection in FC-SAN virtual storage environments to achieve a shorter response time.

By the way, in practice, the access path selection strategy is not often implemented at the application level but at the storage system level to gain more efficiency, and the data placement strategy is often implemented at the application level because it depends strongly on the characteristics of data objects accessed by the application. As this is not the main topic of this paper, we will describe the implementation details later in another paper.

3.2.1 Getting the value of L_{MD}

In the practical use of the conclusions in Section 3.3, the value of L_{MD} is often unknown to the storage administrators, which prevents the administrators from deciding data placement and access path selection with this decision-making direction. Therefore, we bring forward a simple method to get the approximate value of L_{MD} by using the linear model (L_{DATA}–t) obtained from the above sections.

In Section 3.2.2, we have got the result: $T_1 = aL_{DATA}+C_1$. Similarly, we can also get that:

$$T_3=bL_{DATA}+C_3, \text{ in which } b=1/V_{LAN} \text{ and } C_3 \text{ is a constant.}$$

If we take L_{DATA} as x-axis, time as y-axis, and draw the beelines of T_3 and T_1, we can prefigure the following two things: (1) the slope of T_1 is much less than that of T_3, (since $V_{SAN} >> V_{LAN}$, $a=1/V_{SAN}$ and $b=1/V_{LAN}$, so $a<<b$); (2) the y-axis intercept of T_1 is bigger than that of T_3, (as the time for returning metadata is not included in C_3 but in C_1, so $C_1>C_3$).

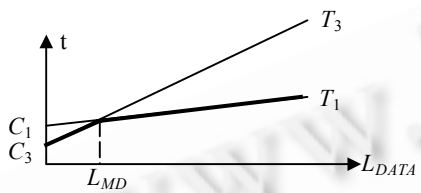

Fig.2 L_{MD} and path selection

The beeline T_3 and beeline T_1 can be obtained by testing. After that, we can get the value of L_{MD} by the way of either geometry or analytics. In Fig.2, the x-coordinate of the point of intersection of beeline T_3 and beeline T_1 is L_{MD} and the thick black zigzag line indicates the sound data access path varying with the value of L_{DATA}.

3.3 Data placement analysis

3.3.1 The concept of equivalent of virtual storage cost

In Section 3.4, the way of how to get the value of L_{MD} so that our data access path selection direction can be used in practice has been illustrated. In this way, the value of L_{MD} is got approximately from the simple linear model (L_{DATA} —t) which covers up a lot of complex details.

So, to be precise, L_{MD} implies the cost paid in the data access of FC-SAN virtual storage resources. When the cost is less expensive than what we get ($L_{MD}<L_{DATA}$), accessing to the virtual storage resources is very efficient; otherwise, it is less efficient than even sharing data on LAN environment (adding the data relay phase), which is the case of what we get is less than what we pay ($L_{DATA}<L_{MD}$). Therefore, we rename the value of L_{MD} got in this way as *equivalent of virtual storage cost*.

3.3.2 Data placement strategy

With the concept of *equivalent of virtual storage cost*, we can develop the decision-making direction in Section 3.3 to a more practical one: (1) Only when the L_{DATA} of one data file is far greater than the *equivalent of virtual storage cost*, we choose to store this data file in the FC-SAN virtual storage environment; and for the most frequently accessing data we should place the corresponding metadata locally, viz., destine MDC on itself. (2) When the L_{DATA} of one data file is less than or near to the *equivalent of virtual storage cost*, we choose to store this data file in the traditional storage methods, such as DAS or sharing data on LAN.

For the doubt that the direction above will lead data placement into a more complex and even more confusing problem, we will discuss it in Section 5.

4 Experimental Results

4.1 Test environment

This test environment is the core of the massive storage subsystem in the digital library prototype[9].

4.1.1 Topology and components

As shown in Fig.3, the test environment is constituted of three servers, one FC hub, one Ethernet hub and a FC disk box. Of the three servers, two are assigned as hosts and one is assigned as MDC. The connections between the Ethernet hub and the servers are all LAN connections; and the connections between the FC hub and the servers, and between the FC disk box and FC hub are all FC connections. The FC disk box is used for expanding the number of FC disk arrays in the environment.

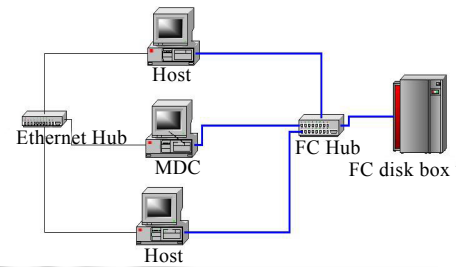

Fig.3 Test environment

4.1.2 Software, hardware and data resources

The operating system is Windows 2000 Professional, and the primary software is Tivoli SANergy version 2.2; the primary hardware includes one 8-port FC switch, three IBM FastT200 storage servers, and one IBM FastT200 storage expanding box, four FC hard disks (10Krpm, totally 135G).

The data resources include several types of files: texts, pictures, three-dimensional animations, videos and some meaningless machine-generated files. However, in our test, we pay attention to the sizes of the files, but not the types of the files. So, for the continuity of our test result curves, we will use some meaningless machine-generated files, which will not affect the correctness of our results.

4.2 Test results

The following results are got either by Tester in Tivoli SANergy version 2.2 or by Performance Monitor in Windows 2000 Professional. The record size is 1,000KB.

4.2.1 Verificaiton of the assumption of the tiny constants Δ_1, Δ_2, Δ_3, Δ_4

In Section 3.2.2, all the discussions are in the steady transmission rate range of both FC and LAN connections. So when the values of T_1 and T_2 are tested varying with L_{DATA}, the transmission rates should be monitored at the same time. We take the steady transmission rate of LAN connection for granted as only some light load of metadata takes place. Therefore, only FC connection should be monitored.

In Fig.4, the steady transmission rate range is from 231MB to 755MB, and in this range, T_2 is increasing linearly with L_{DATA} (shown as file size in the figure). In Fig.5, the steady transmission rate range is from 20MB to 80MB, and in this range, T_1 is increasing linearly with L_{DATA} too. Thus according to the method in Section 3.2.1, the assumption of the tiny constants Δ_1, Δ_2, Δ_3, Δ_4 is true. In other words, the items $t_{resolves\ metadata}$, $t_{issues\ real\ data\ request}$, $t_{MDC\ searches\ metadata}$, and $t_{Host\ requests\ metadata}$ can all be treated as tiny constants compared to the items $t_{MDC\ returns\ metadata}$ and $t_{access\ data\ through\ FC\ connections}$.

In addition, we also test a group of result for T_3. As shown in Fig.6, the steady transmission rate range is from 20MB to 80MB, and in this range, T_3 is increasing linearly with L_{DATA} too. With Figs.5 and 6, we can be sure that the linear models in Section 3.4: $T_1 = aL_{DATA} C_1$, $T_3 = bL_{DATA} + C_3$, are practical and reasonable to be used for getting the value of L_{MD}.

4.2.2 Getting the value of L_{MD}

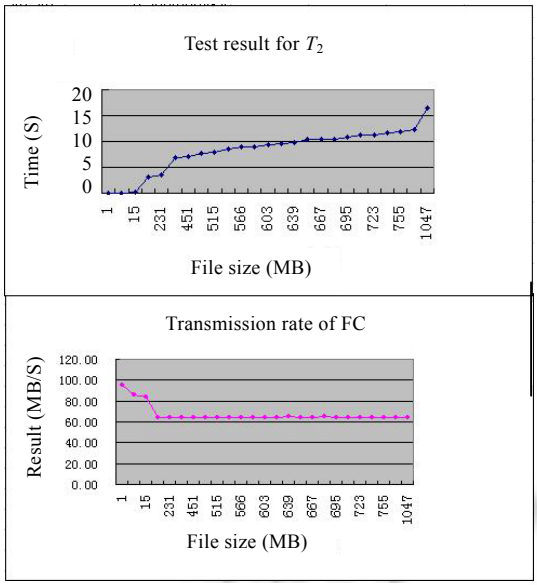

Fig.4 Read from MDC (T_2)

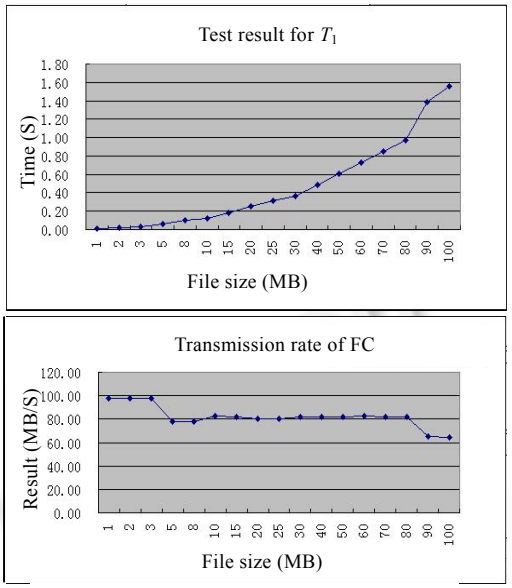

Fig.5 From host (T_1)

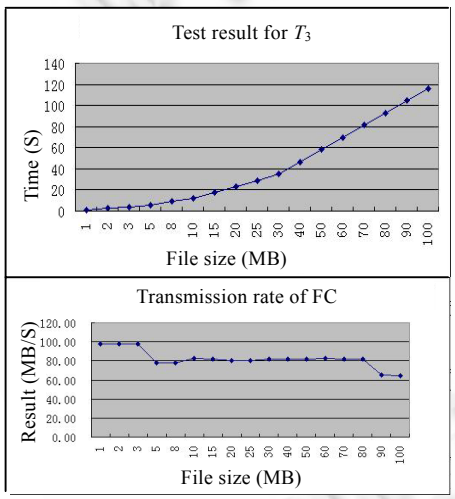

Fig.6 From host indirectly (T_3)

Following the method in Section 3.4, the lines of T_3 and T_1 are drawn according to the real testing result. But as shown in Fig.7, the slope of T_3 is much greater than that of T_1. Multiplying T_1 by 100, we get Fig.8, in which T_1X100 and T_3 are very similar to each other. This implies that taking $V_{SAN}>>V_{LAN}$ as a fact is practical and reasonable too. But in this case it is hard to pick out the point of intersection by the way of geometry. So, an analytics way should be used to get the value of L_{MD}.

The four steps of the analytic way are as follows: (1) focus on the steady transmission rate range near to the origin point; (2) in this range, imitate beelines for T_3 and T_1; (3) solve the analytic expressions T_3' and T_1' for the beelines; (4) work out the intersection of T_3' and T_1' by solving the linear equations, and the x-coordinate is the value of L_{MD}.

Following the s5teps above, we get the value of L_{MD}: 5.5KB.

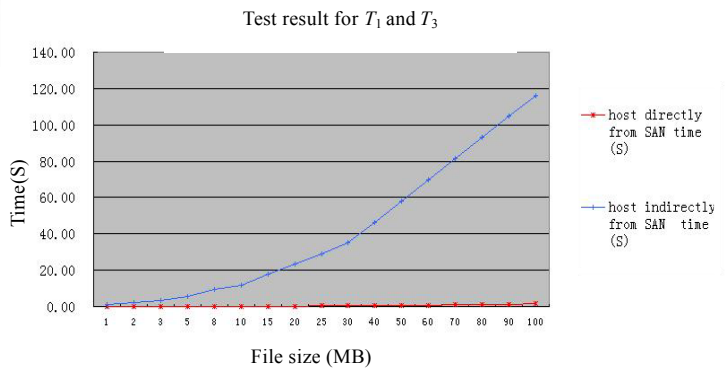

Fig.7 Comparison of T_1 and T_3

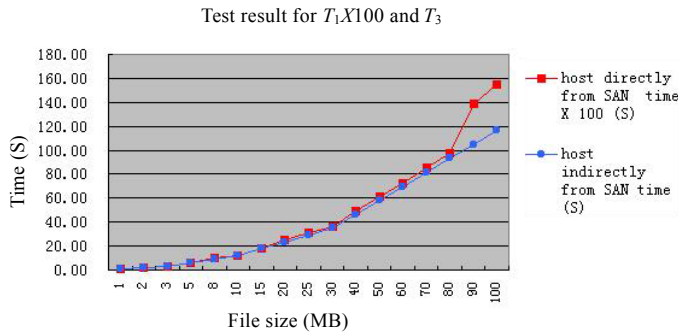

Fig.8 Comparison of T_1X100 and T_3

Mathematic 4.1 is a good tool for this work. As the value of L_{MD} here means equivalent of virtual storage cost, the value may vary in different FC-SAN virtual storage environments, but the value can be got in the same method as described above.

4.2.3 Discussion on the "Part Linearity" phenomenon

In our extensive tests, we find that sometimes T_1 is folded to several line segments but not a straight line as what we suppose (Fig.9). We call this phenomenon "part linearity". The reason for this is not clear now, but the intuitional thought is that "part linearity" may cause more than one intersection points for T_3 and T_1. Figure 10 shows this case.

But in one linear segment, there could be at most only one intersection because there is at most one intersection for two beelines in one plane. This still agrees with our linear model for L_{MD} in Section 3.4.

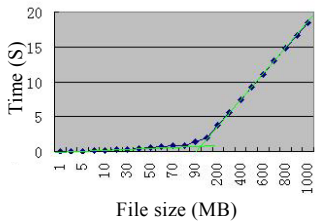

Fig.9 "Part Linearity" phenomenon

Deep consideration indicates that T_3 and T_1 will never have more than one intersection unless the performance of the transmission rate of FC connections is degraded severely (V_{SAN}< V_{LAN}). The thick black line segment in Fig.10 shows the degrading. As the slope of T_1 is $1/V_{SAN}$ and the slope of T_3 is $1/V_{LAN}$, so only when $V_{SAN} < V_{LAN}$, $1/V_{SAN} > 1/V_{LAN}$, i.e. the slope of T_1 is bigger than that of T_3. But in normal situations, $V_{SAN} > V_{LAN}$, so the slope of T_1 is normally smaller than that of T_3. This means the y-coordinate of T_1 will not increase faster than that of T_3 in normal situations. Therefore, T_1 will not catch up with T_3 to form the intersection point after their first intersection. Figure 11 shows the normal ubiety of T_1 and T_3 in the case of "part linearity". The thick black zigzag line indicates the sound data access path varying with the value of L_{DATA}, which is consistent with the direction of data access path selection in Section 3.3.

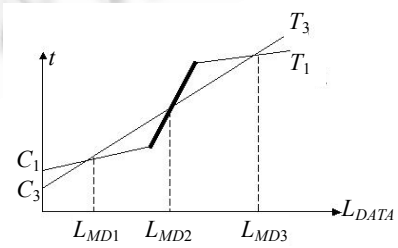

Fig.10 Multiple intersections

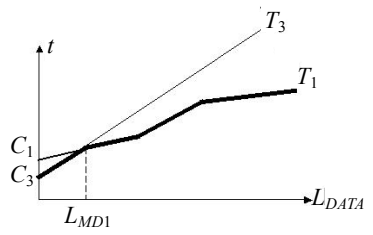

Fig.11 Normal situation

5 Discussion and Future Work

In Section 3.5, the concept equivalent of virtual storage cost is brought forward, and the direction for data placement is developed according to this concept. The main idea of this direction is that if the size of file is bigger than *equivalent of virtual storage cost*, storing the file in FC-SAN virtual storage environment is worthy of the virtual storage cost; otherwise, it is not worthy of the cost, and the file should be stored in the traditional storage environment. Someone will doubt that the direction may lead data placement into a more complex and even more confusing problem, but our belief is that each storage method or environment has its own distinguishing features and is suitable for its own range of storage tasks. Our work is to find out the law that which method is suitable for which task. But things are not stopping here. For the complex applications such as digital libraries, the storage task is a mixture, from cold tiny text files to hot huge video files. In this case, our solution should be an orderly reasonable combination of kinds of storage methods, and each storage method do its best in the whole solution, so that the combination serves the application more efficiently than any single storage method. Therefore, our future work is to work out the most efficient combinations for different applications, and Digital Library is one of the most representative ones. The combination solutions of FC-SAN and DAS/NAS will bring some extra management work but not a lot: for the administrators of storage systems, knowledge of more kinds of storage devices is required; and for the administrators of servers, by utilizing the virtual storage technology, storage spaces from different storage devices appear as the logical volumes, and the operations on these volumes are absolutely the same. The only one difference between these volumes is that the performances are varying. To make each storage method do its best in the whole solution, the simplest way is to authorize different application the proper right on proper logical volumes, which forces the usage strategy.

In fact, the decision-making method of data placement and data access path selection brought forward in this paper is only the first step. This method is static and basic. For some applications such as video service in digital libraries, this static method cannot solve the problem efficiently when there are too many users demanding several hot video clips. In this case, we can exploit the potential of FC-SAN by placing data and replicating data dynamically. As the data placement is dynamic, the data access path selection will be dynamic too. We believe that this is a meaningful and promising research area for us.

6 Conclusions

The massive storage system is the key component for large-scale digital libraries. Data placement and access path selection is the first thing to considerate in any storage environment deployment. This paper focuses on finding the direction of data placement and access path selection for a better performance in the FC-SAN virtual storage environment, which is still an open problem.

In this paper, a linear time-consuming model about data access is put forward based on the analysis of the virtual storage principle in FC-SAN environment. A decision-making method of data placement and access path selection is given based on the model. Then the concept of equivalent of virtual storage cost is brought forward, and the meaning together with the practical usage of this concept is discussed, which helps to develop the decision-making method: (1) Only when the size of one data file is far greater than the *equivalent of virtual storage cost*, we choose to store this data file in the FC-SAN virtual storage environment; and for the most frequent access data we should place the corresponding metadata locally, viz., destine MDC on itself. (2) When the size of one data file is less than or near to the equivalent of virtual storage cost, we choose to store this data file in the traditional storage methods, such as DAS or sharing data on LAN. (3) As for hosts, when $L_{DATA}<L_{MD}$, take the way of accessing data through LAN connections relaying by MDC; and when $L_{DATA}>L_{MD}$, just directly through FC

connections.

After that, all the assumptions and methods of the former theoretical analysis are verified by the experiments in a FC-SAN virtual storage environment, the core of the massive storage system in our digital library project. In the experiment phase, a practical four-step analytic way for L_{MD} is brought forward, and the "part linearity" phenomenon is discussed, which all help to apply our decision-making method into practice. This method is a simple and practical direction for data placement and access path selection in a massive storage system of coexisting FC-SAN and DAS to achieve a better performance.

References:

[1] Clark T. Design Storage Area Network: A Practical Reference for Implementing Fibre Channell SANs. 2nd ed., Massachusetts: Addison Wesley Longman, Inc, 2000. 1~6.

[2] Phillips B. Have storage area networks come of age?. IEEE Computer, 1998,31(7):10~12.

[3] Storage Networking Industry Association. http://www.snia.org/

[4] Kilmatin P. Success stories in high availability using VERITAS software eBay. 2001. http://eval.veritas.com/downloads/sus/eBay.pdf

[5] IBM. Old ominion University moves up to a smart SAN solution from IBM and Vicom. 1999. http://www.storage.ibm.com/ibmsan/press/dominion/dominion.pdf

[6] IBM. CP&L automates remote data backups with an integrated IBM storage solution. 2000. http://www.storage.ibm.com/ibmsan/magstar.pdf

[7] IBM Tivoli. Frequently Asked Questions Tivoli® SANergy™. 2001. http://www.tivoli.com/products/documents/faqs/sanergy_faq.html.

[8] IBM Tivoli. Tivoli SANergy Administrator's Guide Version 2 Release 2. 2nd ed., 2000.

[9] Xing CX, Zhou LZ, *et al*. Developing Tsinghua University architecture digital library for Chinese architecture study and university education. In: Lim EP, *et al*., eds. Proc. of the 5th Int'l Conf. on Asian Digital Libraries. New York, Berlin: Springer-Verlag, 2002. 206~217.