The Comparison of Data Replication in Distributed Systems

Iman Zangeneh, Mostafa Moradi,Ali Mokhtarbaf

Abstract**—**The necessity of ever-increasing use of distributed data in computer networks is obvious for all. One technique that is performed on the distributed data for increasing of efficiency and reliablity is data rplication. In this paper, after introducing this technique and its advantages, we will examine some dynamic data replication. We will examine their characteristies for some overus scenario and the we will propose some suggestion for their improvement.

Keywords**—**data replication, data hiding, consistency, dynamic data replication strategy

I. Introduction

GRID networks and distributed systems are payed special attention for accessibility to the processing and data resource and also for conecting the relations. For correct accessing to the data resource and also for protecting of correctness of these resource, some policies and techniques must be performed. One of these techniques is creating of multiplyed versions from distributed data. Performing of this method and its analysis could be useful in distributed systems efficacy for different reasons, because it decrease width band consumption, and decreasing of delayed accessibility and causes other advantages. The other point is that, with due attention to the increased volume of distributed data and necessity of usage of theme by remote users, in most cases, creating the similar versions of data is the only way. In the next chapter, we will describe and define the data replication and the we will show their difference with data hiding .In the third chapter , in addition to short introducing of some existance techniques for replication of data , we will compared these techniques in the last chapter, we will conclude and we will suggest some suggestion for future.

II. Data Replication

In distributed system, file transportation from remote sever to the machine user, causes the bandwidth consumption in all of the route between user and server. Maybe the requested file is big, and this increases the delaying of accessibility. For solving this problem usually the data replication technique is used. Also replication helps to the equality of the yielding and increases reliability via creating some copies from same data. In addition replication could increase the tolerance of distributed system error, and increases scaling of system.

III. Replication Strategy

Now, in short we introduce some important strategies for the case that users and resources are organized in tree hierarchical order topology.

For selection, usage and performing of these strategies, apart from the distributed system three general problems must be considered [3]:

How is the computer users relation topology and resources, in distributed system?

What are the functions that on the data station are the data resource? These functions are divided in two general: reading from station and writing on the station. Sure, in most of the presented strategies, we assumed that only reading action is performed and complicated case that includes writing is not considered. Here, we follow this assumption for easiness.

The other important problem is the evaluating criterion, and selecting different strategies and ranking these criterions. The response time average is used as a basic criteria for assessment of replication different techniques. The other important criteria is the used bandwidth for data transporting. The other important subject is that the strategy must minimized the costs of accessibility to data such as protecting the false version, updating the versions, replication and maintenance costs of the storing place (memory), that it uses by false versions.

A. Dynamic and Static Replication

In theory, replication is divided in two parts of static replication and dynamic replication [5]. In most of the large distributed system, the volume of the data is high and it is about PetaByte. Therefor such system need to the dynamic replication strategies, which there, eliminating and management of the copy versions are performed automatically. Therefore in the next, we will introduce the dynamic strategies.

Now in short we introduce some important strategies for the case that users and resources are organized in tree hierarchy order topology [1, 2].

1) Without Replication or Hiding

This strategy in reality is then base method for comparison of the other strategies. In this technique replication does not occurred. All data collection are in the root of hierarchy. In this way we could perform the colloction of the access design and we could measure the average of the time response and width of the consumed band as a assessment criteria.

2) The Best User

In this way, every user node, protects detailed antecedent of its file. This record shows the numbers of requests for each file, and the nodes which each request is coming from it. Therefore this strategy is work in such way that in obvious time distance, every node controls that are the numbers of requests for every its file are exceed of threshold or not.If it is so, the best user is selected for this file. The best user is the one that create the most of request. Therefore this node creates a copy version of this file in the best user. Therefore, all of the file that, request for them exceeds of threshold are replication

Iman Zangeneh is with Islamic Azad University of BaghmalekBranch,Iran,I_zangeneh@yahoo.com
Mostafa Moradi is with Islamic AzadUniversityof BaghmalekBranch,Iran,mos_moradi@yahoo.com
Ali mokhtarbaf is with Islamic Azad University of BaghmalekBranch,Iran,ali_mkf@yahoo.com

in other place.

3) Spring Replication

In this strategy, when the request of root for a file exceeds of threshold, the copy version is created in the next level, but it is on the route toward the best user; Therefore this new place for this copy version is the father of best user. In such way, when the request for the second level of hierarchy of data exceeds of favorite threshold, then the replication is performed in the next lower level, and in this order a requested favorite file maybe replicated automatically.

We must mention that here requested designs could present three positional characteristics for data files [2]:

1-Temporary position: The files that recently are begin accessible, possibly will be accessible again.

2- Geographic location (user location): The file that recently are begin accessible by a user, possibly by a user near it is access for him.

3- Spacial location (file location): The files which are near to the files that recently are accessible possibly will be access again.

4) Simple Hiding

If a user requests a file, a copy of that file will be copied on that place.

5) Hiding plus to Spring Replication

In this way the strategy number 3 and strategy number 4 are combined. The user save the files in the region place. Server, alternatively recognizes favorite files and scatters them into lower hierarchy.

6) Rapid Development

In this way a copy version of file is saved in all of its route toward the user. That is when the user request a file, one copy is saved in every layer. This matter causes the rapid development of data.

IV. THE COMPARISON OF REPLICATION STRATEGIES

A. The comparison of replication dynamic strategies

For comparison of the different introduced strategy we need to define the accessibility designs. We used these designs from three degrees [1].

P-Random: There is no pre assumption position.

P1: The data are in the low rate of temporary location.

P2: The data are in the low rate of the geographic and temporary location.

The results, that will be indicated following are for the five strategies of without replication, simple hiding, best users, hiding plus spring and rapid development. We don't discuss absolute spring because we can see its results in the hiding plus spring strategy. These results are attained for three above accessibility design degree on the basis of simulation for thousand requests. When the P-Random model is used, all of the strategies except best user and spring show the significant improvement in comparison without replication. It does not seem that best user works well for random accessibility, because the average of responses is four times more than the time in without replication. Also the best user almost equal to the base method of without replication consumes the width band.

In the time of using of P1 accessibility model , all of the strategies except best user, both in the term of accessibility delay both in width band consumption could improve the accessibility process, the term of P2 model, only the best user strategy does not show any saving. Even for bandwidth, saving by using of the best user is insignificant. Ten percent of saving is in comparison with without replication. Although the fourthy percents improvement in accessibility delay is significant in comparison with base method, therefore the best user always works worse that simple hiding.

In fig. 1 and fig. 2, the results of spring plus hiding, rapid development and simple hiding are shown. The two first strategy with simple hiding which is the comparisons standard are compared. Thus figure 1 shows saving results that obtain from rapid and spring method [1].

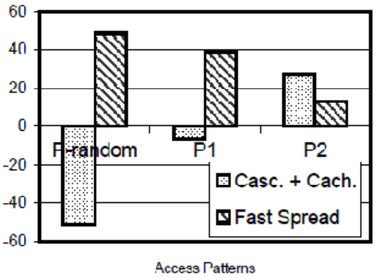

Fig. 1 Saving percent in response time for three access pattern based on simple hiding

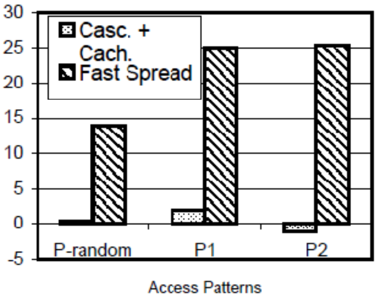

Fig. 2 Saving percent in bandwith for three access pattern based on simple hiding

As it in the fig. 1 and 2, the spring method does not work well when the accessibility models do not include any position, for random data, the response time is better , when simple hiding is used instead of spring method. The rapid development method also is better than simple hiding and does not work for random data.

There is almost a fifty percents eduction in response time in term of rapid development.

B. The comparison of replication in distributed system and in the data station

The strategies that examined up to now, are used in distributed system. Replication technique could be used in the data station specially. Although the presented strategies in this two discussion in term of concept are similar, but in term of the most of the aspect are similar, but in term of the most of the most of the aspect such as model, assumptions, mechanisms, validation, and performing are different [3]. Replication in the distributed systems, because of error tolerance and in the data station for efficiency is important. In this two field, techniques and mechanisms are similar, but comparison of used protocol in two field is difficult work. Because of some special delicates, the mechanism are very similar, but in function have many difference. For this, the using of the results of the one field in the other field is complicated work.

V. Discussion And Conclusion

Among dynamic replication that is discussed in this paper, the best user strategy had a worst function among other strategies. in most of the function additional costs that this strategy produces is more that its benefit and its advantages and it works worse that without replication base. What the experimental results show, the best strategy there is not for all the scenario. Although the rapid development method both in term of response time both in term saving in bandwidth works better than simple hiding, but its additional cost is high, because all of the saving space completely used by rapid development.

Although the results for the above methods, are very promising, but the results of the scenarios and work environments are artificial. To obtain more accurate results, we need to realistic scenarios.

A fundamental point that the topic should be regarded as replication, adaptation, and the next thing to keep them out Data correctly. One of the main problems is consistent with keeping copies. The unofficial, means that when a copy is updated, you must also ensure that other copies are updated. Otherwise, copies will not be the same. If replication to improve reliability and efficiency will help, what can I prevent it? Unfortunately, the proliferation of data should costs be paid. This problem is amplified by the presence of multiple copies, may be compatibility problems. Whenever a copy is modified, it will be copied from other copies. To ensure consistency in results, these reforms should be applied to all copies. When and how to reform, the cost of replication will take place.

To understand the problem, improve time to consider access to web pages. If certain measures, if not, you may receive a web page from remote server can take several seconds to complete. To improve performance, web browsers locally received a copy of the web page will save (put in the cache). If the application requests the same page again, the browser automatically returns the local copy. Access time seen by the user, it is interesting. But if the user wants to always have the latest version of the page, it may be unfortunate. The problem is that if this change in the distance, the reform does not apply to the copy in the cache. Consequently, it will be worn copies.

A solution to the problem, return the old copy to the user that they will not be in the cache. But this solution, if the copies are not near the user, access time is increased. Another way is to allow to server to updating the copy in cache or invalid it. But this requires that all server memory cache to store and send messages to them. This reduces the overall system performance.

Compatibility, is only half the story. Should also consider implementing adaptation have. Usually two or a few points to consider. First of all, managed to locate copies | Contribute Copy'll not only take into consideration, but also defines how content providers are distributed in [4].

The second point, keeping the copies consistent. In most cases, applications need to adapt to the strong. The informal, means that when making the copies should be distributed immediately. Different methods for implementing strong consistency can be defined as work that is suggested.

References

[1] Kavitha Ranganathan and Ian Foster, 2002.“Identifying Dynamic Replication Strategies for a High-Performance Data Grid”, Department of Computer Science, The University of Chicago, Chicago, America.

[2] Houda Lamehamedi, Boleslaw Szymanski, and Zujun Shentu, 2002.“Data Replication Strategies in Grid Environments”, IEEE computer society Press, Los Alamitos, CA, pp, 378-383.

[3] M. Wiesmann, F. Pedone, A. Schiper, B. Kemme, G. Alonso, 2000. “Understanding Replication in Databases and Distributed Systems”, Swiss Federal Institute of Technology (ETHZ), Institute of Information Systems, ETH Zentrum, CH-8092 Zürich, Swiss.

[4] Bettina Kemme and Gustavo Alons, 2000. ”Dont be lazy: Postgres-R, Anew way to implement Database Replication”, Information and communication System Group, ETH Zurich, Switzerland.

[5] Heinz Stockinger, Asad Samar, Bill Allcock, Ian Foster, Koen Holtman, Brian Tierney, 2001.“File and Object Replication in Data Grids”, Proc. 10th Intl. Symp. on High Performance Distributed Computing