16.5 A SECURE AND LOW COST NETWORKEDCONTINUATION OF OPERATIONS STANDBY SYSTEM FOR THE OAR SCIENTIFIC MANAGEMENT SYSTEM

Eugene F. Burger[1]*, Jeremy Warren[2]

[1]JISAO, University of Washington, Seattle, WA 98195
[2]NOAA/OAR, 1315 East-West Highway, Silver Spring, MD 20910

Finding solutions for ensuring continuation of operations (COOP) for enterprise applications is a challenge for anyone who manages organization-wide applications. The Department of Commerce requires all NOAA operational systems to have a continuation of operations plan in place and that necessitates us to consider standby alternatives for a component of the OAR Science Management System.

The first line of defense to ensure system redundancy is usually some backup regime that would include regular incremental and full system and data backups. Typically copies of these backups are kept off -site. There is, however, the chance that a catastrophic event, like a natural disaster, could disable the application in question, and also completely disrupt the transport and network infrastructure where this application is hosted. Even if rebuilding the application at the same location is possible, network connectivity and access to offsite backup tapes could be problematic.

A full standby system would be the obvious solution to provide complete redundancy to an organization wide system and while this is the most desirable implementation, there were reasons why such a system would not be feasible for FDMS. This paper will discuss a low cost alternative we evaluated and implemented for an OAR wide application.

1. Background

The Financial Data Management System (FDMS) is a component of the NOAA OAR Science management system. This application provides financial and science project related budgeting information to administrative office at all of NOAA Research's office across the country.

The application is built on MS Access and the data for each office is separated into its own database. We are in the process of redesigning this application to serve data from a unified database. User access to the FDMS desktop application is through a Citrix Metaframe client that allows secure and seamless access to the FDMS application. Data are downloaded daily into the FDMS Oracle and SQL Server databases from the CAMS Data Warehouse, the NOAA system of record. After data transforms have been applied to make these data compatible with the FDMS application, the data are transferred to the different FDMS instances. The FDMS application is hosted on four servers, all located at the NOAA campus in Seattle.

Redundancy for hardware failures on the FDMS servers is provided by backup power supplies and RAID 5 disk arrays. Daily backups of each server's system state and all data provide some level of insurance in case of a full server failure.

The FDMS application, although important to OAR management, is not a mission critical application. This gave us more flexibility in considering standby alternatives for FDMS. The objective of this investigation was to explore options for a low cost standby system for en enterprise level system, like the FDMS application.

2. Possible solutions

The first and obvious option is to rely on the recovery of system state and data from backup tapes, either copies kept on or off-site. This reliable option, already in place with FDMS, can not be depended on as the only standby system for the reasons given above.

Another possible solution would be to completely duplicate the FDMS hardware and software infrastructure at another location. This would mean a doubling of the number of servers and software licenses we now need for FDMS. Such a duplicate system would then continuously be updated with application updates and application data. Because of this application infrastructure duplication, costs for FDMS would nearly double. Additional network costs for standby system network connectivity will be incurred, that will further increase the cost of ownership of this system. Furthermore, it is completely feasible that this system might never be used before it has to be replaced. Because of the cost implications and the other reasons given, it was felt that an operational full standby system would not be a viable solution for FDMS.

3. The FDMS solution

The third solution we considered is to replicate all FDMS data and system information to a remote storage device. This information would then be used as the data source to build a new FDMS implementation.

The device we would replicate to would be a low cost, high storage volume, Network Attached Storage (NAS) device. NAS devices provide cheap and high volume RAID 5 redundancy storage

space. Should we have to revert to these data and application information at the time of a disaster, personnel would then rebuild the FDMS systems from the data and system information on the NAS box. Replacement data and application servers would have to be purchased to rebuild the FDMS application on.

For this test we chose a NAS box that uses the Windows operating system because the majority of FDMS servers are Windows servers. Using a Windows operating system on the NAS device would allow us to serve data directly from the NAS device without having to copy the data to the new server. While a NAS box is cheaper than a regular server with the same data volume, another low cost alternative would be to use a retired server that has sufficient storage space.

To facilitate the replication of the data and system state information, we considered a number of applications: PeerSync from Peer software, Availl Replication from Availl and RepliWeb's R1 and RDS products. Although there are many more replication applications available, we evaluated only these products because of our replication software requirements. We drew up a list shortlist of replication software requirements that included differential mirroring, multiple channel mirroring, data encryption and the ability to mirror data over the internet.

The capability to mirror data over the internet, and encrypt the data during transmission was imperative since we do not have access over dedicated data lines to the destination server. For this reason the replication would be over the Internet. Some of the data in question do contain sensitive information that necessitates data transport encryption. Obviously it was important that this encryption also apply to the authentication information that would be sent to the replication destination.

Only the R1 application had a bandwidth tuning feature. With this feature a job could be modified to use a pre-set percentage of available bandwidth certain times of the day. A job can, for example, be configured to use only 5% of available bandwidth during work hours when the user work load on the server is higher, but as much bandwidth as needed during low user workload hours.

Differential mirroring allows for only those data blocks of large files that have changed since the last replication, to be transferred. This feature significantly shortens the time it takes to replicate typically large files, like database files. Differential mirroring cannot be applied to certain file types (database backup files) and to speed up replication for these files the ability to do multiple-channel replication would be needed. Multiple channel replication speeds up the process by using more than one replication process to replicate a single file. Using multiple-channel replication does require more CPU cycles. This is an important consideration on servers with a heavy workload with a priority towards user application performance.

Continuous file replication ensures a file is copied to the replication server as soon as it is changed on the source server as apposed to time scheduled replication, where the file transfer is time scheduled.

The capability to run on multiple operating system was another requirement since the FDMS application uses a combination of Windows and Linux operating systems for its servers.

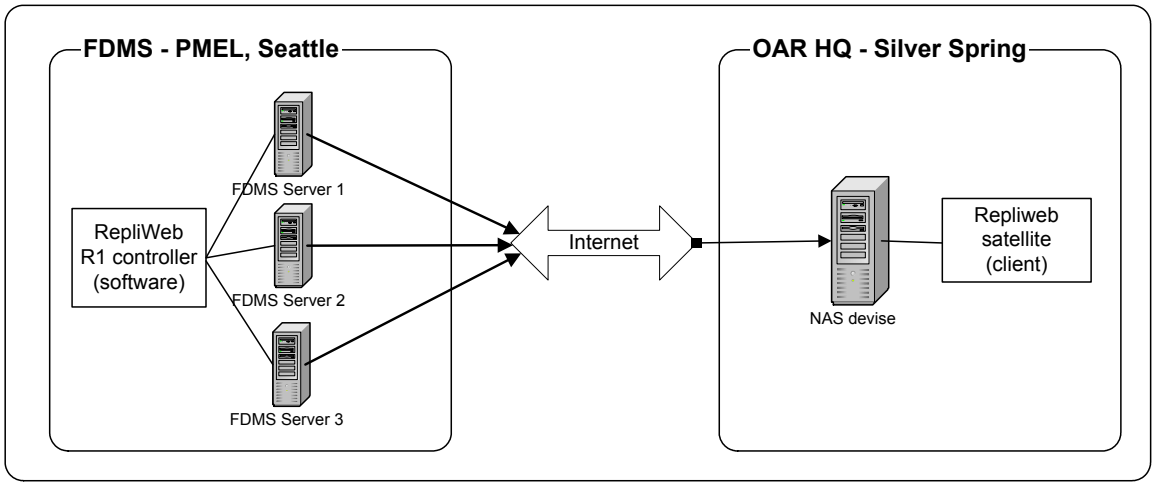

Figure1. The FDMS standby-system topology.

Considering these requirements, we decided on the R1 replication software from RepliWeb.

We have been using this replication infrastructure for three months and although we have never had to revert to this standby system, we have appreciated the additional benefits to this standby provides. The roll back ability that has allowed us to quickly retrieve damaged data files. Another advantage is the capability to compress the data on the destination server, which allows us to extend the roll-back period (the length of times for previous copies of files are kept).

Our test procedures were to first ensure the replication software complied with our minimum requirements. We then tested the viability of replicating the data files from all the database servers' application we use with the FDMS system. File replication was tested over our local network, and later over the internet. We found it a shortcoming that open files or locked files could not be replicated. Although there is a third party application that would enable this functionality with the software we use, we instead chose to use time scheduled replications of backups of the data files.

By looking at the replication network traffic between a test server and the destination server we confirmed the data were encrypted. Another test was to see what the transfer time for very large files were. By configuring a replication job to use multiple replication processed for file groups that would contain very large files, we were able to get respectable replication times. The time to replicate one 12 GB file took 118 minutes. Finally, the NAS box was shipped to OAR Headquarters where we are now using it to replicate FDMS application information and data to.

Although all the software and data servers used with FDMS are of the shelf software, the configuration information for this software is vitally important should the whole server ever have to rebuild. Because of this, comprehensive system and application documentation is a vitally important component to this standby system.

5. Recovery scenario

Rebuilding the FDMS application from the data and application information on the NAS box would first require colleagues at the OAR Silverspring headquarters to purchase replacement servers for FDMS. They would then reinstall the all applications and database servers required by the FDMS application, and restore user authentication information from the data on the NAS box. They have the option of serving the data directly from the NAS box, or to copy the data to the new data server.

This downside to this implementation is that it does not provide an immediate system fallback capability, and that rebuilding the replacement system is time consuming. Fallback tests we did during the evaluation period showed that we were able to rebuild a partial system and start serving data within a day.

6. Conclusion

This type of standby file-system is a viable option for a non mission critical operational system, such as the FDMS application. In the short time that this system has been operation we have found it to be reliable, cost effective, and have appreciated the additional data redundancy capability it has provided.

A drawback to this system is that it is not an immediate standby system and there will be a time lag while a new system is built from the application and system data contained on the standby storage device.

This implementation relies heavily on reliable internet connectivity to maintain a full system mirror on the storage device. Should network connectivity be interrupted to the FDMS servers, continuous data replication would also be interrupted. This was not a major concern since FDMS also relies on the internet for user connectivity. Users would thus not be making changes to their data during network downtime.

Acknowledgement

This publication is funded by the Joint Institute for the Study of the Atmosphere and Ocean (JISAO) under NOAA Cooperative Agreement No. NA17RJ1232, Contribution #1100. PMEL contribution No. 2765. The views expressed herein are those of the authors and not necessarily those of NOAA or any of its sub agencies.