A PRACTICAL METHODOLOGY FOR ARCHIVING, RETRIEVING, AND MANAGING LARGE QUANTITIES OF DATA

Carl Reinwald and Bob Vossler

Realtime Data Systems Center
Computer Sciences Corporation
301 North R Street
Lompoc, CA 93436

ABSTRACT

With the ever-increasing volume of data generated by today's telemetry processing systems, contemporary data archive technologies are proving inadequate for permanent data storage requirements. Typically after data have been acquired and recorded, specialized engineers will analyze and compare the current data with data archived during past operations. As the quantity of archived data increases, the data's accessibility becomes increasingly difficult to manage efficiently. A new archiving concept has been designed and implemented which provides efficient automated access to over three terabytes of data. The nominal retrieval time for a fifty megabyte file is less than two minutes. The system's storage media is standard VHS high-energy cassettes with a storage capacity of 5.2 gigabytes per cassette.

INTRODUCTION

The Realtime Data Systems Center (RDSC) of Computer Sciences Corporation (CSC) has developed an efficient and cost-effective mass storage archive for the Test Support Facility Integrated Flight Data Processing System (TSF IFDAPS) at the Air Force Flight Test Center (AFFTC) at Edwards AFB, CA. This paper documents RDSC's archive design methodology used to implement the TSF IFDAPS mass storage archive. RDSC's archive design utilizes a new mass storage archive capable of efficiently accessing over three terabytes of data within minutes of a request. A brief introduction to this mass storage archive precedes the archive design description.

MASS STORAGE ARCHIVE

The mass storage archive consists of Honeywell's Very Large Data Store (VLDS) and Very Large Archive (VLA) cassette jukebox. The VLDS is a rotary magnetic tape recorder designed to record and reproduce digital data on standard VHS high-energy cassettes. Using high-density recording techniques, a VLDS cassette can retain 5.2 gigabytes of user data at a cost of less than $0.0021 per megabyte. The VLDS unit is available in either a single or dual channel configuration and is capable of sustaining data transfer rates of up to two megabytes per channel. Standard host interfaces include synchronous SCSI and VME. The VLDS unit incorporates a Reed-Solomon error correction code which results in an error rate of one uncorrectable error every 12.5 gigabytes. Figure 1 shows a typical VLDS interface configuration.

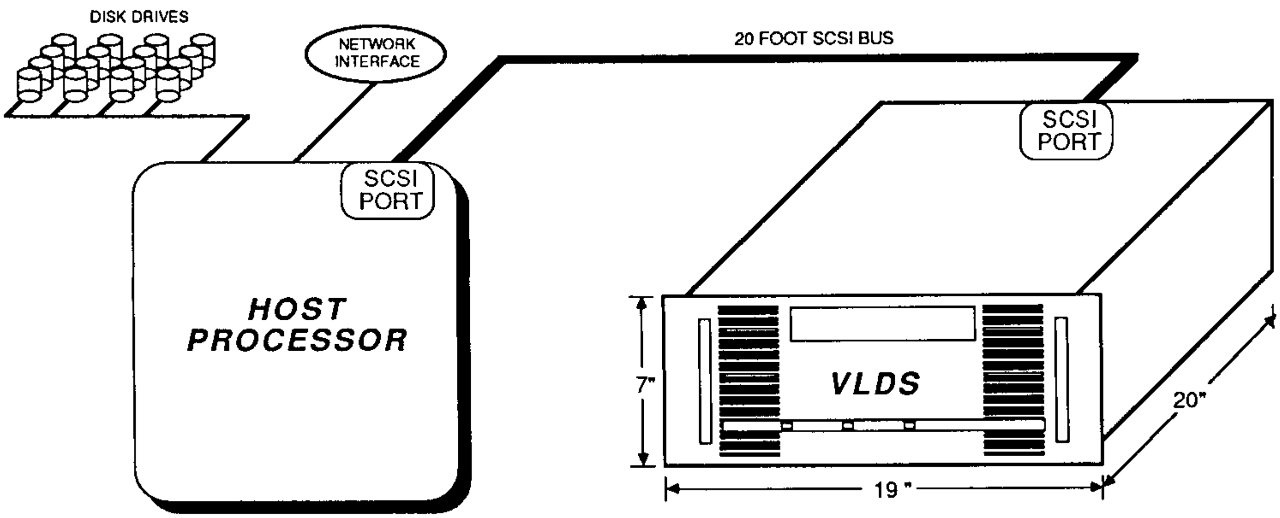

FIGURE 1. VLDS HOST PROCESSOR INTERFACE

A VLDS tape is segmented into a series of principal blocks. For a single channel VLDS unit, an entire tape contains 158,691 principal blocks where each block holds 32 kilobytes of user data. A block identity number is automatically recorded with each principal block and is referred to as a Principal Block Number (PBN). PBNs provide faster access to files stored on VLDS tape by specifying a file's location absolutely rather than relative to other files and images on the tape.

A VLDS unit functions similarly to a standard home Video Cassette Recorder (VCR). After inserting a tape one may record data, reproduce data, fast forward or fast reverse the tape the same way a standard VCR does. End-to-end search of the tape takes approximately 90 seconds. The VLDS unit is a Write Once Read Many (WORM) recording device which inhibits the writing of data over a previously recorded tape section. However, if a tape has not been used to capacity, the VLDS does allow additional information to be stored on the tape starting from the last principal block that was recorded. Bulk erasing of the entire tape permits a tape to be reused once the information on that tape is no longer required.

One major design consideration when utilizing the VLDS results from the necessity to synchronize the VLDS unit's servo mechanism to the tape's control track pulses. When attempting to read or write a VLDS, the VLDS must first rewind the tape approximately 100 PBN from the actual starting PBN. The VLDS then moves the tape forward, reads the control track pulses, and synchronizes the VLDS unit's servo mechanism to the tape's control track. This latency period (approximately two seconds) creates a noticeable decrease in the overall data transfer rate and therefore starting and stopping the tape while saving and restoring files should be kept to a minimum. As a result, saving and restoring small files from VLDS tape can be a time consuming process. Therefore RDSC's archive design methodology focuses on archiving and restoring multi-megabyte size files.

The VLDS unit may be implemented as a single, standalone archive unit or incorporated into the VLA. The VLA is an automatic VLDS cassette handler that holds 600 VLDS cassettes for a total on-line storage capacity of over three terabytes (equivalent to 20,000 reels of 6250 BPI, GCR nine track tapes). The VLA robotic assembly consists of two 300-cassette carousel storage drums, the VLA cassette handler mechanism, and a bar-code reader capable of reading bar-code identification labels placed on the cassette. Up to six VLDS units can be mounted in the VLA. Access to any VLDS cassette is possible in less than 8 seconds and the VLA foot print is slightly larger than an office desk. The VLA interfaces to the host computer through a standard RS-232C port. Figure 2 is a graphic representation of the VLA.

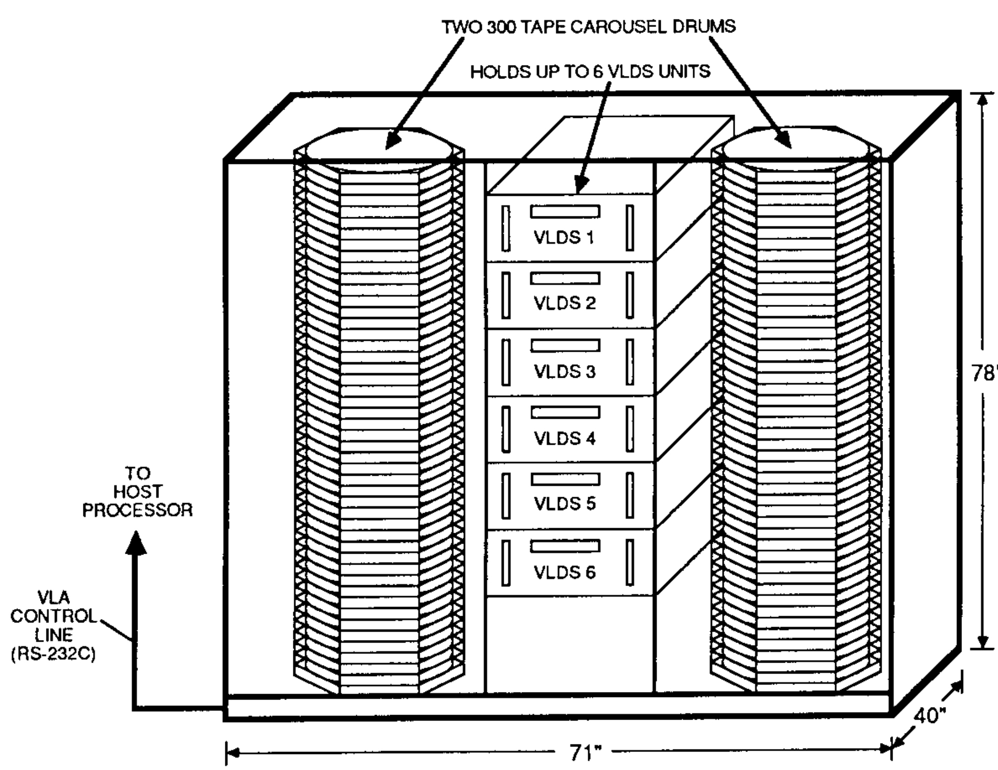

FIGURE 2. THE VERY LARGE ARCHIVE (VLA)

VLDS/OPTICAL DRIVE COMPARISON

In the past, digital tape and optical disks have been the two primary mass storage media. For a large archive, digital tape proves to be too unmanageable. Therefore the primary storage media for a mass storage archive has been the optical disk. RDSC identified three significant advantages the VLDS mass storage device offered over optical disk drives. The primary advantage is data transfer rate. A typical optical disk drive has a nominal data transfer rate of 300 kilobytes per second. A single channel VLDS unit can maintain a data transfer rate over 600 percent faster than optical disk drives (two megabytes per second) and a dual channel VLDS unit is over 1200 percent faster. However, the VLDS unit does not provide a read-after-write capability and therefore direct comparison of the data transfer rates are misleading for data archiving. To save a file, the VLDS software records the file, rewinds the tape to the start of the file, and verifies the data recorded by reading the error correction code. This reduces the effective archive data transfer rate to approximately 750 kilobytes per second - still over 200 percent faster than optical disk drives. For all VLDS tape file restorations, the VLDS data transfer rate is the full two megabytes per second.

A second advantage the VLDS unit has over an optical disk drive is cost per megabyte of storage. An optical disk costs approximately $0.025 per megabyte where a VLDS tape costs approximately $0.0021 per megabyte. This cost difference represents a savings of almost $25,000 dollars per terabyte of data storage.

The third VLDS advantage concerns the storage media’s reusability. Optical disks are true WORM media; once written, the optical disk is a Read Only media. The VLDS tape is also a WORM media with the exception that the tape can be bulk erased and used again once the original data is no longer required.

ARCHIVE DESIGN METHODOLOGY

Having selected the mass storage device, RDSC developed the following archive design methodology for the TSF IFDAPS mass storage archive implementation.

A multi-processor system will derive the maximum benefit inherent to implementing the VLA. Up to six VLDS units may be mounted in the VLA and therefore up to six processors may have direct access to a VLDS unit in the VLA. One processor is typically designated as the file server to which the VLA and one or two VLDS units will be connected. The file server’s disk drives provide a staging area for restoring files from VLDS tape and are hereafter referred to as staging disks. Specialized storage archive tasks would control access to the VLA and VLDS units. The VLA’s robotics would be controlled by the file server.

In this multi-processor system, all processors are connected to a common network. Network tasks are implemented on each processors to provide remote file access and network file transfer capabilities. As mentioned above, specialized storage archive tasks control access to the VLA and VLDS units. These tasks provide both local and remote access to the VLA and VLDS unit(s) interfaced to the file server.

To manage the large quantity of data which the VLA is capable of supporting, two databases are recommended. Both databases would reside on the file server. The first database identifies all files on each VLDS tape in the VLA and will be referred to as the Archive Database. The two primary items contained in each Archive Database entry are the VLDS tape identification label and the file's pathname. The tape identification label identifies which VLDS tape the file is stored on. The file's pathname indicates where the file is located if the file is on one of the file server's staging disks. Maintaining the pathname in each Archive Database entry provides two benefits. First, a user or application program can reference one common place to determine where any file is located on the file server. Second, the pathname information prevents the restoration of a file which already resides on a staging disk. For a telemetry data (TM data) archive system, each Archive Database entry could also contain the operation that the file is associated with, the time slice contained within this file, and any other file specific information.

The second database manages the VLDS tapes in the VLA and will be referred to as the VLA Database. A VLA Database entry will exist for each of the 600 VLA tape bins. Each entry will contain the VLDS tape identification label, the type of files saved on the VLDS tape (TM data files, system files, etc.), and the space still available on the tape. The entry also reflects if no VLDS tape is present in the corresponding VLA bin.

RDSC envisions a typical utilization of the archive described above as follows. A user on a processor other than the file server, requests access to a specific file. Through a network communication task, the request is transferred to the file server. One of the specialized storage archive tasks checks the Archive Database to determine if the file is already on a staging disk. If the file is on a staging disk, the pathname is returned to the user. If the file is not on a staging disk, another storage archive task uses the VLDS tape id obtained from the Archive Database entry and checks if the tape is contained in the VLA. If the VLDS tape is in the VLA that tape is mounted into one of the file server's VLDS units. The file is then automatically restored to one of the staging disks. Upon completion of the file restoration, the user has access to the requested file.

The final element of the archive design is the initial archiving of files to a VLDS tape. If VLDS units were only interfaced to the file server, any file requiring archiving to VLDS tape must first be transferred to the file server if it is not residing on one of the staging disks. Assuming the data files are relatively large, the transferring of files to the file server would contribute significantly to network congestion. To alleviate this network congestion, a VLDS unit may be connected directly to each processor which generates files requiring archiving to VLDS tape. As mentioned previously, up to six VLDS units may be mounted in the VLA. Besides the file server, up to five other processors may have direct access to a dedicated VLDS unit mounted in the VLA. Since the file server controls the VLA's robotics, these five other processors would send requests across the network to load and unload tapes to and from their corresponding VLDS unit. When a file is archived directly to VLDS tape from one of these processors, the file archive utility would send a request to the file server to add a new entry to the Archive Database. The file would then be available for immediate restoration if a user on another processor requests access to that specific file.

RDSC utilized the archive design methodology described above to integrate a VLDS/VLA mass storage archive into the TSF IFDAPS system. RDSC's mass storage archive provides the TSF IFDAPS analysts with quick and efficient access to all archived data within minutes of their request. The remaining sections of this paper document RDSC's actual implementation of this archive design methodology.

TSF IFDAPS OVERVIEW

Figure 3 is an overview of the TSF IFDAPS system which RDSC designed, integrated, and delivered to the AFFTC at EAFB. Three identical real-time Flight Monitoring Subsystems (FMS) acquire and process the telemetry data (TM data). Each FMS contains a History Processor (HIS) which records the acquired TM data. The Storage Archival Subsystem (SAS) functions as a file server, providing access to the archive TM data. The Storage Archive consists of a temporary storage area (staging disks) and an archival mass storage device. All subsystems are connected through Network Systems Corporation's HYPERchannel. All subsystems have access to a history processor's current TM data as well as the archive TM data on the SAS through the HYPERcharmel. Figure 4 graphically depicts the interconnection between the SAS and the HIS. Figure 4 shows only those hardware and software elements which directly relate to the archiving and retrieval of TM data files. Both the HIS and SAS processors are Gould Concept 32 mainframes. For a detailed description of the TSF IFDAPS system, referred to reference number one.

The Air Force plans to acquire and record up to 7.4 gigabytes of TM data from each flight test operation. This recorded TM data must be rapidly archived and removed from the history processor to facilitate preparation for the next operation. The SAS must provide access to all archived TM data files. A year's worth of archive TM data must be available within minutes of an analyst's request without manual intervention. Additionally, the archive TM data files must be stored on removable media to simplify the transferring of TM data to remote sites for additional analysis and post-flight processing.

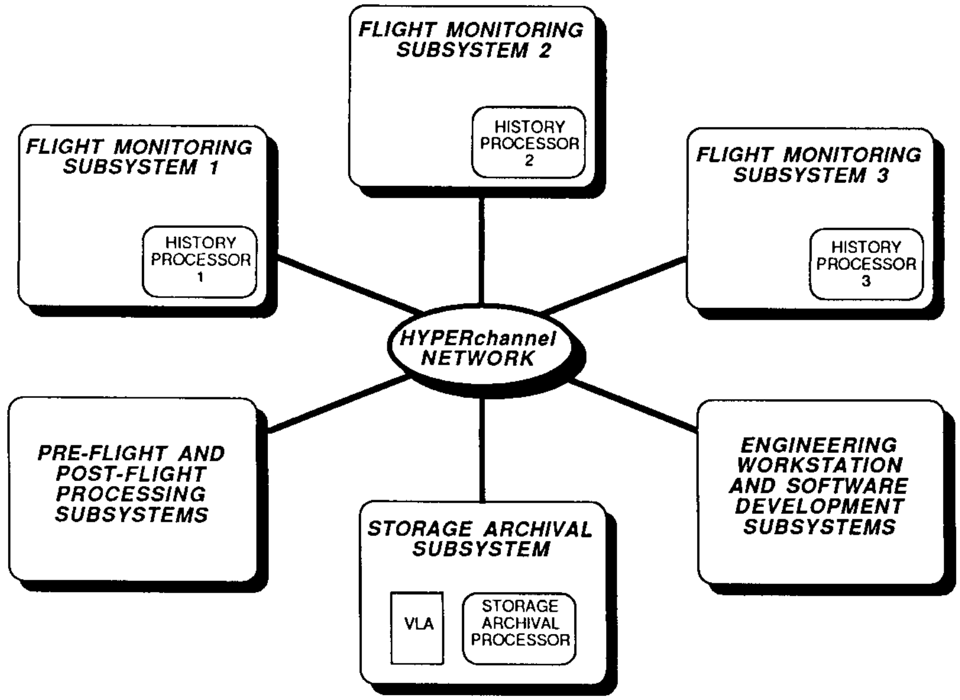

FIGURE 3. TSF IFDAPS SYSTEM OVERVIEW

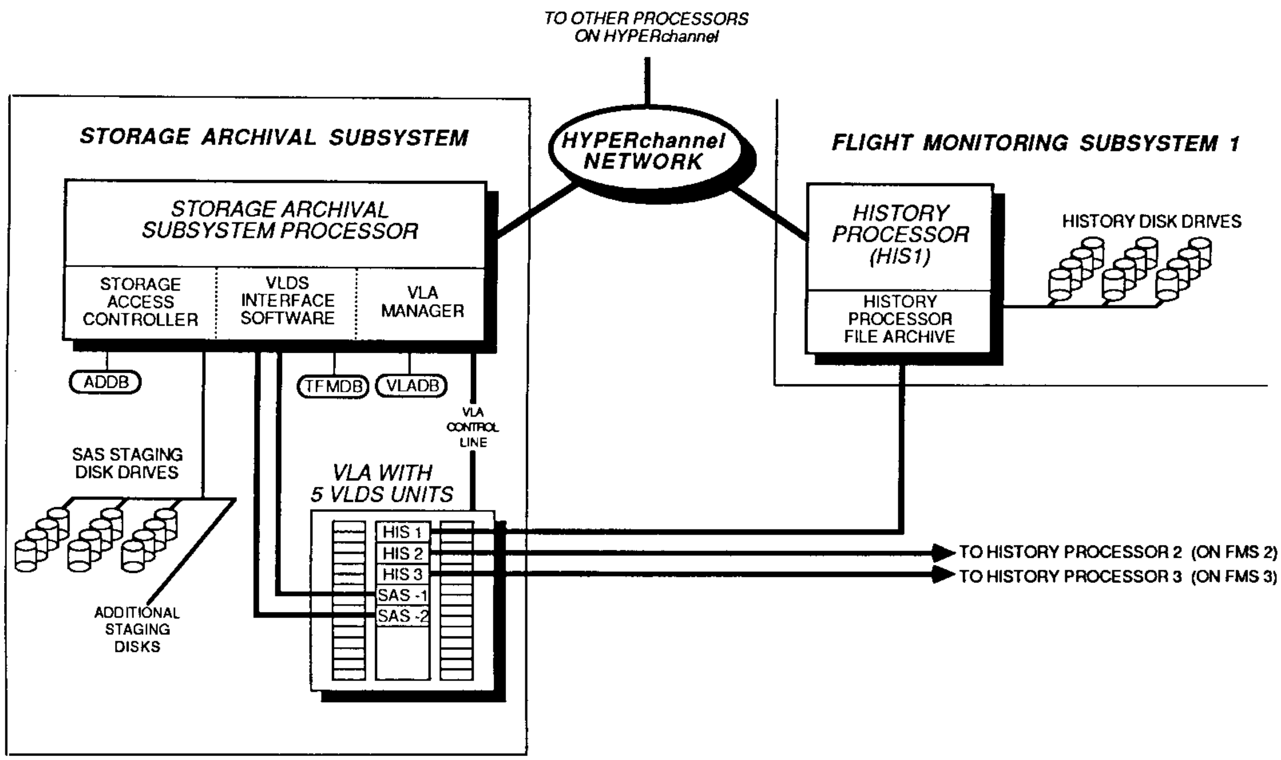

FIGURE 4. SAS AND HIS INTERCONNECTION OVERVIEW

RDSC selected five single-channel VLDS units and the VLA as the mass storage device. Each VLDS unit was set to a two megabytes per second data transfer rate and each was equipped with a synchronous SCSI board. As figure 4 indicates, two VLDS units are connected to the SAS processor and one VLDS unit is connected to each HIS processor. All five VLDS units are mounted in the VLA

HOST COMPUTER INTERFACE

Gould Concept 32 mainframes do not have a standard synchronous SCSI board but Gould does provide a High Speed Data Interface (HSD) which can achieve data rates of up to 3.2 megabytes per second. RAC Information Systems of Great Neck, NY developed an HSD-to-SCSI controller board which plugs directly into the HSD and provides a synchronous SCSI interface. One initial drawback of the SCSI interface was the 20 foot limitation of the SCSI bus. Since four different processors were going to be connected to VLDS units mounted in the VLA, it was imperative that the VLDS interface length be extended. To accommodate this specification, the HSD/SCSI controller board was designed for remote installation. The HSD/SCSI board can be removed from the HSD, installed in a remote chassis, and connected back to the HSD via standard HSD I/O cables. This capability extends the VLDS's interface range from the SCSI limit of 20 feet to 100 feet. A custom HSD/SCSI controller board chassis was developed by RDSC to house the five HSD/SCSI boards and provide the necessary power.

To provide the software interface between the VLDS units and the Gould mainframes, RDSC utilizes two of RAC's software packages. RDSC uses the Archiver Software package to save and restore standard operating system files to and from VLDS tape on the SAS. The Archiver maintains a disk database which facilitates access to any file on a VLDS tape by maintaining the starting PBN and file size for every file stored on VLDS tape. Since the TSF IFDAPS history data is recorded in a non-standard file format, RDSC developed a custom VLDS file archive utility which utilized RAC's Fortran Callable Subroutines package. RDSC specifically developed this file archive utility to write data in the same format as the Archiver program.

STORAGE ARCHIVE DATABASE

TSF IFDAPS telemetry data files may be as large as 123 megabytes. RDSC set this file size limit for ease of handling the TM data files under the host computer's operating system. If every VLDS tape in the VLA were filled to capacity with 123 megabyte TM data files, there would be more than 25,000 files contained in the VLA. Since files are typically less than 123 megabytes, the VLA could contain even more files. The Archiver program limits a VLDS tape's capacity to 1100 files and therefore the VLA's capacity is 660,000 files.

For the TSF IFDAPS implementation, RDSC incorporated several databases into the storage archive. Archival and retrieval software utilize these databases to provide automated access to TM data or other files stored in the VLA. First there is a VLA database (VLADB) which contains all the information needed to manage the bins in the VLA. Second there is the Archiver program's database (TFMDB) which contains all the information needed to locate any file on any VLDS tape and to restore that file to a staging disk. A third database, the Archive Data Database (ADDB), provides retrieval of TM data by operation name and time slice keys. Each ADDB entry contains the operation name, the time slice (file start and stop time and day), the file's path name, and the tape identification label for the VLDS tape on which the file has been archived.

HISTORY PROCESSOR FILE ARCHIVING

For TSF IFDAPS the majority of archived TM data files originate from a history processor. If VLDS units were only connected to the SAS, any file generated on the history processor would have to be transferred across the HYPERchannel to the SAS before the file could be archived to VLDS tape. To alleviate this HYPERchannel congestion, a dedicated VLDS unit is connected directly to each history processor. This configuration permits archiving of any history processor file directly to VLDS tape.

To sustain recording rates as high as 2.4 megabytes per second for an entire operation, the history processor interleaves the TM data on to three history disks simultaneously. A special History Recording task was developed by RDSC to perform this data interleaving and support recall of the TM data. The entire amount of recorded data for an operation is treated as if it were one large file spanning up to twelve disk drives (7.4 gigabytes).

Obviously, archiving data in this format would be irretrievable on another processor without a special task and enough disk space to restore the entire 7.4 gigabytes. To facilitate archiving and management of the TM data from a history processor, the data is segmented into a series of history slices. Each history slice is approximate 123 megabytes and as many as 64 history slices may be generated from a single operation.

Given the special nature of the FMS history file, RDSC developed a special purpose history file archive program. This custom archive program was designed to transfer the history file data from disk to VLDS tape at the throughput rate of the VLDS unit (2.0 megabytes per second). Achieving this data transfer rate insures that the entire history slice file can be transferred to VLDS tape without stopping the VLDS unit. Four megabytes of memory buffers are needed to double buffer the writing of data to the VLDS tape.

This program interfaces through the HYPERchannel with programs running on the SAS in order to request a VLDS tape be mounted in the history processor's VLDS unit before starting an archive. Upon completion, the history processor file archive program requests that the tape be returned to the VLA. The program also updates the ADDB through the HYPERchannel.

STORAGE ARCHIVE ACCESS

Typically, analysts request data by specifying a time slice from an operation. The ADDB's primary function is to provide the correlation between a time slice from an operation and a corresponding file. RDSC developed a special Access Controller program to provide access to TM data via the ADDB. One of its primary functions is to utilize the ADDB to translate a user defined time slice within an operation into a file name which contains the requested data. Upon finding a file which contains the requested time slice that file is restored to disk if it was not already there. The ADDB entry contains the VLDS tape identification which is used to located the tape in the VLA. With the file name and the VLDS tape id label, the file's tape location is extracted from the TFMDB and the file is automatically restored to a staging disk. The nominal file restoration time is approximately two minutes.

VLA MANAGEMENT

RDSC also developed a VLA Manager program which provides both a manual and an automated interface to the VLA. All VLA functions may be executed manually including moving VLDS tapes from a VLA bin to a VLDS unit, returning a tape from a VLDS unit to a VLA bin, moving tapes from one bin to another bin, etc. The VLA Manager maintains a VLA database (VLADB). A VLA Database entry exists for each of the 600 VLA bins. Each entry contains the VLDS tape identification label, the type of files saved on the VLDS tape (TM data files, system files, etc.), and the space still available on the tape. The entry will be blank if no VLDS tape is present in the corresponding VLA bin.

The VLA's two carousel drums contain 300 tapes each. Each drum has six sides and each side contains 50 tapes (see figure 2). A door on the side of the VLA provides access to one drum side at a time. Therefore of the 600 tapes in the VLA, 50 tapes can be loaded and unloaded at any one time. The VLA Manager was designed to always use the same drum side for adding and removing tapes. Using the VLA's robotic arm, the VLA Manager can move tapes between these 50 import/export bins and the other 550 bins. This design limits a user's access to only 50 bins. The VLA Manager has complete control over the remaining 550 bins.

When an automated file restore request is processed, the VLA Manager program is requested to mount the appropriate VLDS tape into a VLDS unit. This program determines which VLA bin the VLDS tape is in via the VLADB. The VLA robotics moves the cassette mechanism to the specified bin. The built-in bar code reader reads the tape's bar code label to ensure that the requested tape is in the specified bin. The VLA robotics then moves the tape into the specified VLDS unit.

When archiving TM data from the history processor, the appropriate VLDS tape is determined by the VLA manager. If files for the particular operation have already been archived to a VLDS tape, that VLDS tape is automatically selected to insure that all files from an operation are archived to the same VLDS tape if possible. If no files have been archived from a particular operation, a blank VLDS tape is selected for archiving the operation's file. The VLA Manager utilizes the VLADB to determine when a tape is too full to be used for archiving.

CONCLUSION

Contemporary data archive technologies are proving inadequate for permanent data storage requirements. By integrating the VLA, VLDS units, and an efficient archive management system RDSC has created an extraordinary mass storage archive. The VLDS capacity and data transfer rate exceed the performance of any data storage system with removable media. The VLA provides multiple processors with direct access to several terabytes of data. The archive management system presented in this paper provides an efficient methodology for tracking large numbers of data files. The RDSC implementation of the TSF IFDAPS mass storage archive demonstrates the practicality and efficiency of utilizing VLDS units and the VLA. Data that once took hours or days to access can now be reached within minutes.

REFERENCES

1. Quesenberry, Dave and Reed, Gary, "Telemetry Processing - Realtime to Postflight", CSC-RDSC, ITC Annual Proceedings, 1988.