N94-33824

Value Added Data Archiving

Peter R. Berard

Battelle Pacific Northwest Laboratory
P.O. Box 999, MS K1-87
Richland, WA 99352
(509) 375-6591
(509) 375-6631 (fax)
pr_berard@pnl.gov

Abstract

Researchers in the Molecular Sciences Research Center (MSRC) of Pacific Northwest Laboratory (PNL) currently generate massive amounts of scientific data. The amount of data that will need to be managed by the turn of the century is expected to increase significantly. Automated tools that support the management, maintenance, and sharing of this data are minimal. Researchers typically manage their own data by physically moving datasets to and from long term storage devices and recording a dataset's historical information in a laboratory notebook. Even though it is not the most efficient use of resources, researchers have tolerated the process.

The solution to this problem will evolve over the next three years in three phases. PNL plans to add sophistication to existing multilevel file system (MLFS) software by integrating it with an object database management system (ODBMS). The first phase in this evolution is currently underway. A prototype system of limited scale is being used to gather information that will feed into the next two phases. This paper describes the prototype system, identifies the successes and problems/complications experienced to date, and outlines PNL's long term goals and objectives in providing a permanent solution.

Introduction

Researchers in the Molecular Sciences Research Center (MSRC) of Pacific Northwest Laboratory (PNL) spend a considerable portion of their time on the encumbering task associated with managing their scientific data. Automated tools that support the management, maintenance, and sharing of this data are minimal. Researchers typically manage their data by physically moving datasets to and from long term storage devices and recording a dataset's historical information in a laboratory notebook. While this process has been tolerated, it is not acceptable for managing the amount of data researchers will be generating in the near future.

The Environmental and Molecular Sciences Laboratory (EMSL) is currently under development at PNL for the U.S. Department of Energy (DOE). The goal of this construction project is to field a fully functional, equipped, and staffed research facility in early 1997. The EMSL will be operated by PNL as a DOE Collaborative Research Facility open to scientists and engineers from the academic community, industry, and other government laboratories for collaborative research in the molecular and environmental sciences. Major facilities within the EMSL include the Molecular Sciences Computing Facility (MSCF), a laser/surface dynamics laboratory, a high-field nuclear magnetic resonance laboratory, and a mass spectrometry laboratory. The MSCF will consist of the High Performance Computer System (a massively parallel processor), the DataBase Computer System (described below), and the Graphics and Visualization Laboratory.

With the development of EMSL, it is anticipated that by the turn of the century, data to be archived annually will be on the order of seven terabytes. The size of individual datasets will reach tens of gigabytes and the total amount of data each researcher will manage is expected to

increase significantly. Manually managing this scientific data and maintaining historical information about individual datasets will prove to be cumbersome, if not impossible. The need for high-speed, large-scale data transfer and long-term storage and retrieval of scientific data is critical to the MSCF. Large data streams will be produced by multiple computational experiments and instruments. The data archival and retrieval required to support the post-processing for these experiments and instruments is the primary driver for the high performance DataBase Computer System (DBCS).

Given these observations, researchers have two basic needs: 1) a data archiving facility that allows immediate access to any given datasets and 2) an automated means by which to maintain and access historical information about individual datasets. The solution to this problem will evolve over the next three years in three phases. As part of the MSCF, the DBCS will be used for scientific data management. PNL plans to add sophistication to existing multilevel file system (MLFS) software by integrating it with an object database management system (ODBMS) (i.e., value added data archiving). The goal is for DBCS to provide researchers with a completely automated facility in which both datasets and the associated historical information will be electronically accessible. Each phase of DBCS will be implemented on increasingly more sophisticated and powerful hardware architectures. The first phase in the evolution of DBCS is currently underway. A prototype system of limited scale is being used to gather information that will feed into the next two phases.

The sections that follow provide an overview of DBCS and describe the activities associated with each of the three phases of the DBCS development.

Database Computer System (DBCS)

DBCS will be a scientific information management "instrument." DBCS will provide data archival services over a backbone network connecting most offices, users, workstations, and servers. In addition, data archival services will be provided for very high bandwidth data transfers using a High Performance Parallel Interface (HIPPI) based high speed network. Research scientists using the EMSL will access theses services via a graphical user interface. Behind the scenes, an ODBMS will be integrated with MLFS software to provide a sophisticated data archiving package for managing scientific data (i.e., datasets) and files. DBCS will be rich in methods to store, manage, and effectively search and browse information about datasets that are part of the file system.

The MLFS will provide virtually infinite file size and file system size. This is made possible by automatically moving or *migrating* files up and down a hierarchy of successively faster but lower-capacity storage devices (levels). High speed storage will be provided by a Redundant Array of Inexpensive Disks (RAID). Medium and low speed storage will be provided by "robotic removable" media devices/robots. The IEEE Storage Systems Standards Working Group is in the process of developing the IEEE Mass Storage System Reference Model [1] that will eventually result in a set of standards for mass storage system software. It is important for the long term viability of DBCS that the MLFS be based on the Reference Model. This requirement ensures that future upgrades of one or more components of the MLFS software will be feasible due to the industry standard interfaces between components.

The ODBMS component will provide persistent storage of information about datasets and files in the MLFS. This information, often referred to as *metadata*, will allow associative access to datasets and files by information other than filename. In addition, the ODBMS will provide sophisticated and extensible querying facilities, support for versioning, views, and additional security.

DBCS will be implemented in three distinct phases:

1. DBCS-0 prototype system

2. DBCS-0

3. DBCS-1

The DBCS-0 prototype is a system of limited scale and is described in detail below. DBCS-0 will be an interim system that will provide a hardware and software platform on which to develop a scientific database application, as well as tools for users and application developers. The final production system, DBCS-1, will be acquired and implemented in the third phase.

Phase I: DBCS Prototype System

The DBCS prototype system includes an ODBMS, MLFS, and minimal supporting hardware. This system is being used to gain hands-on experience and knowledge with these types of products. The prototype system's hardware includes a host computer system, SCSI disks, and an 8-mm tape robot archive. The National Storage Laboratory's version of UniTree (NSL UniTree) (hereafter referred to as UniTree) is used as the MLFS software and ObjectStore is the ODBMS.

The host computer system is an IBM RS/6000 980 POWERserver running version 3.2.3e of the AIX operating system and has 128 megabytes of memory, one 970 megabyte and two 1.37 gigabyte internal SCSI disk drives. A disk farm consisting of four Seagate SCSI-2 disk drives (connected to a SCSI-2 I/O controller) providing a total of 8 gigabyte of formatted disk space is also part of this system. However, none of the SCSI-2 disk space is being used for UniTree support. Instead, this disk space is used for the Andrew File Server (AFS) in support of other EMSL software development efforts. A total of 2 gigabytes of the internal SCSI disk drives serves as UniTree's disk cache. While this is a minimally sized disk cache, it serves the purpose for the near term prototyping efforts. A Comtec ATL-8, Model 54 8mm tape robot archive supports UniTree's long term storage needs. This robot contains two EXABYTE 8500 5gigabyte disk drives and is capable of holding 54 tape cartridges in its carousel. The host computer system is connected to other workstations through a Fiber Distributed Data Interface (FDDI) network.

An NSL UniTree license to manage 250 gigabyte of data has been purchased for the prototype system. For this initial prototype, three users have been identified for MLFS support. Many of their files are in the rangé of 50 to 250 megabytes in size, with the largest file size approaching 500 megabyte. It is expected that files produced by these users will reach 1 to 2 gigabytes within the next year.

Current Status

Due to several unfortunate events, the delivery and installation of the prototype system is behind schedule at the time of this writing. The original schedule called for delivery and installation of the system by February 1, 1993, with an additional 30 days scheduled for a series of acceptance tests. Thus, the system was to be available for general use beginning the first part of March 1993. In addition to problems encountered during system installation, the acceptance tests identified several problems that required resolution. Consequently, progress towards developing an intelligent data archiving system for DBCS has been severely impacted. In any event, the anticipated near-term work required to implement value added data archiving is described below.

Near Term Direction

It is highly desirable that researchers using DBCS have a user-friendly interface by which to access their data in UniTree. The first logical step in achieving this is to develop a layer of software that minimizes the need for users to become familiar with NSL UniTree. In the

prototype system, several scripts will be provided to shield the user from the implementation details of UniTree. These scripts will mimic standard UNIX commands (e.g., **utcp** and **utmv** for moving files to and from UniTree, **utrm** for removing files in UniTree, **utls** for listing files that are in UniTree, etc.). Where appropriate, the scripts will query the user for the file's metadata and store this information into the ObjectStore database. Lessons learned from this initial implementation will feed into future, more robust versions of DBCS.

Perhaps the ultimate mass storage solution is to provide users with what appears to be a virtual file system or file "space" in which mass storage services are performed automatically. One such implementation can be found at the Pittsburgh Supercomputing Center, where AFS has been successfully extended to provide mass storage support and multiple copies of data to users [2, 3]. In the prototype system, this concept will be tested by attempting to implement a file space in which a user's UniTree files are accessed as local files. In reality, this may be as simple as using scripts or C programs that automatically shuttle files to and from a specific directory under the user's home account (using anonymous FTP). Alternatively, it may be possible to use existing tools such as Alex [4]. In Alex, anonymous FTP sessions are disguised as a pseudo-file system that allows users to access FTP sites as a local file system. Since Alex is freeware, it is unlikely that any implementation using this product will provide a long term solution for DBCS.

Problems Encountered and Lessons Learned

Buying an off-the-shelf integrated mass storage system that will meet the needs of the EMSL is simply not possible. Integrating hardware and software to support mass storage needs is a challenging task and requires considerable resources and talent. Like many systems' integration efforts, the time required to implement such a system can easily and grossly be underestimated due to unforeseen difficulties. Integrating this seemingly simplistic mass storage system for the DBCS prototype proved to be such a challenge. PNL is fortunate to be in a situation where time has been appropriated for experimentation. While installing the prototype system resulted in a rather significant schedule slip that was not budgeted for, the lessons learned will prove to be a valuable asset in the next two phases of DBCS and will be carried forward into future procurements. Likewise, it is intended that these lessons be disseminated to others who are undertaking similar efforts.

Perhaps the most significant negative impact was due to the fact that no attempt was made to integrate this system prior to delivery to PNL. The prototype system was perceived as a simple implementation compared to other systems in use today. The fact that the prototype system's hardware is virtually identical to that used by the NSL UniTree development team indicated that the system could be integrated on site. However, certain acceptance tests on the prototype system identified problems that required an update from version 3.2.2 of the AIX O/S to version 3.2.3e, as well as several Program Temporary Fixes. This resulted in distinct differences in the versions of the AIX O/S used by PNL and NSL, and caused severe complications in integrating NSL UniTree. In addition, PNL's tests identified several unknown bugs in UniTree.

The extent of the bugs detected indicate that more extensive testing is required prior to releasing future versions of the software. While PNL does not perceive NSL UniTree to be a mature product at this time, the advanced features it offers for mass storage systems, including third party transfer and support for multiple dynamic hierarchies, certainly warrant it as a worthy candidate for investigation. In any case, the efforts spent on integrating NSL UniTree into this prototype system have not been wasted. In order to avoid these types of problems in future procurements for DBCS, the integrator who is awarded the contract will be required to submit a detailed integration plan, integrate the system prior to delivery, and demonstrate the system by successfully running a predefined set of acceptance tests.

Phase 2: Database Computer System - Level 0

The DBCS prototype system is implementing one hierarchy of storage devices (SCSI disks and an 8-mm tape robot archive). Phase 2 in the evolution of the DBCS, DBCS-0, will expand the capabilities of the prototype system by adding a second hierarchy of more powerful devices (RAID disks and a fast tape robot archive). NSL UniTree will support these multiple dynamic hierarchies of storage devices.

The files and datasets to be stored in DBCS-0 will range in size from less than 1 megabyte to multiple gigabytes. DBCS-0 will efficiently support data archiving of this vast range of file sizes by providing multiple hierarchies of storage devices. The smaller files will be assigned to the hierarchy containing the slow devices (i.e., SCSI disks and 8mm tape robot archive) and the larger files will be assigned to the hierarchy containing the faster devices (i.e., RAID disks and the fast tape robot archive). Determining the optimal match of file size-to-hierarchy will be achieved by experimentation. A RAID Level 3 configuration will result in maximized data throughput and efficient management of large files. It is possible to manage small files on the RAID disks under this configuration by utilizing a log-structured file system [5, 6], that groups the small files into a segmented log. The advantages and disadvantages of using this approach in DBCS-0 have not yet been identified, but will be investigated.

Each hierarchy will consist of two levels of successively faster but lower-capacity storage devices (Figure 1). The "first level" within each hierarchy will consist of disk drives that provide relatively high-speed storage. The "second level" within each hierarchy will consist of one or more tape robot archive machines which provides relatively medium/low speed storage. The "first hierarchy" (i.e., Hierarchy 1) will consist of RAID disks and one or more high capacity tape robots for long term storage of datasets and files. The "second hierarchy" (i.e., Hierarchy 2) will consist of SCSI disk storage and an 8-mm tape robot, both of which are directly connected to the host computer system. Among other things, Hierarchy 2 will be used for intermediate storage of datasets and files. UniTree, the ODBMS, and the 8-mm tape robot, which are part of the prototype system, will be transferred to the DBCS-0 archive computer system.

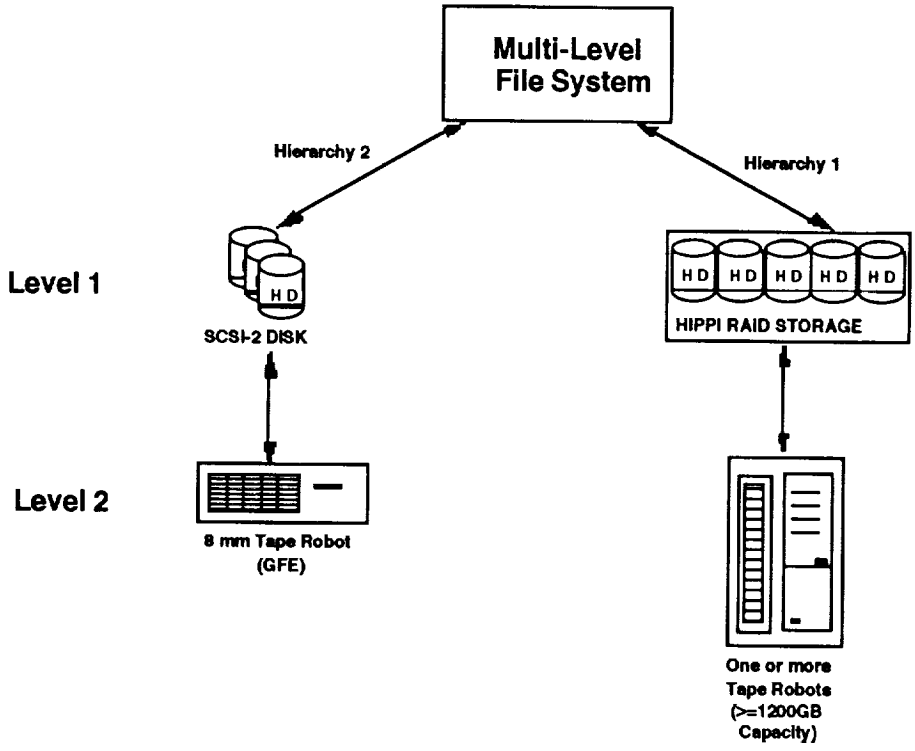

Figure 1: DBCS-0 Levels and Hierarchies

DBCS-0 will support a variety of client platforms, including the High Performance Computer System (HPCS) (a massively parallel processor), High Performance Graphics stations, workstations, etc. (Figure 2). Those clients requiring high speed data transfer (e.g., HPCS, High Performance Graphics stations) will be integrated with DBCS-0 through a HIPPI based high speed network. A slower speed backbone network (i.e., FDDI/Ethernet) will be provided for client platforms that only require medium speed data transfer. The DBCS-0 computer system will be devoted to managing the datasets and files on these client platforms and will be reconfigurable as required (both software and hardware). Third party transfer, as implemented by NSL UniTree, will be utilized for data transfers from HIPPI connected clients. This implementation of third party transfer passes control packets to the host computer system over the slower speed network and passes the data packets over the HIPPI network. This should prove to be a very efficient way in which to move data between clients and DBCS-0.

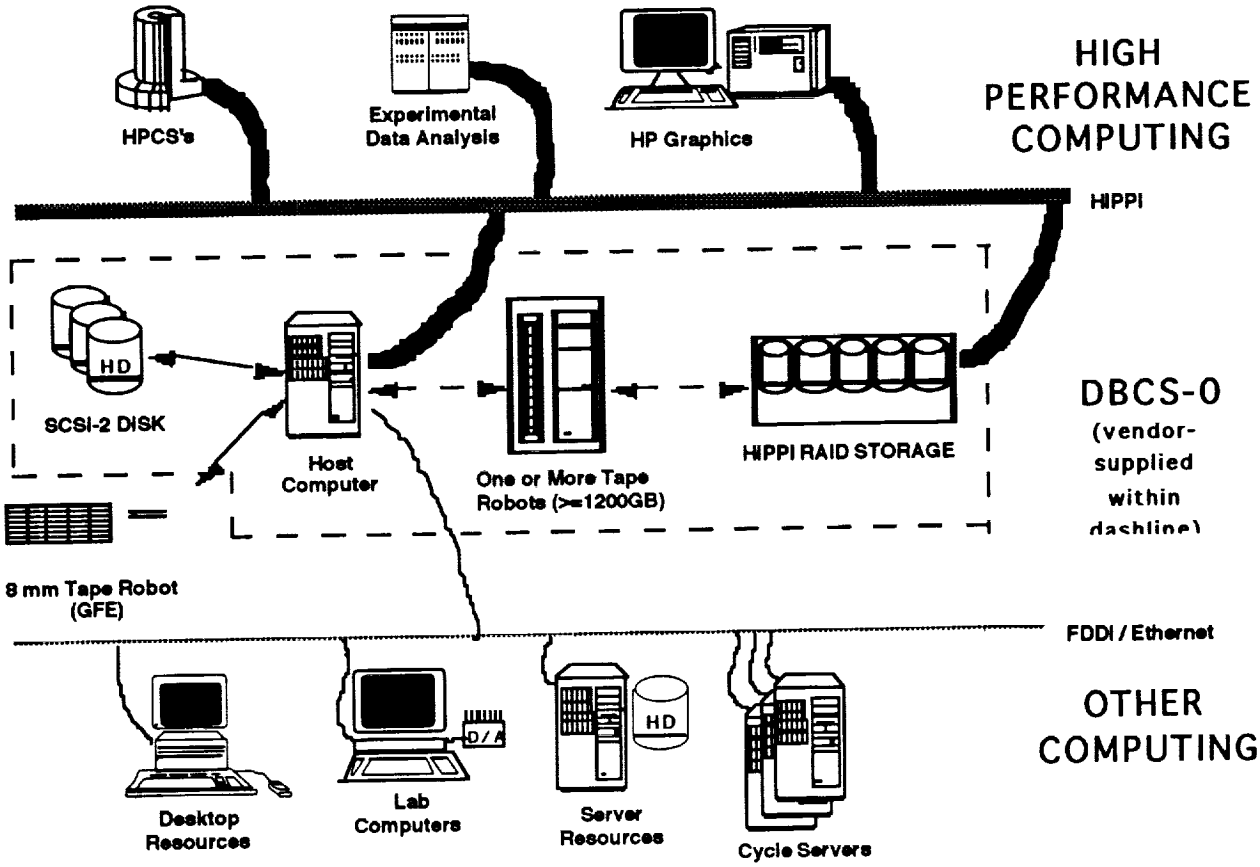

Figure 2: DBCS-0 Connectivity

One of the greatest challenges anticipated in implementing DBCS-0 will be supporting a massively parallel processor (MPP) such as HPCS. The problems associated with supporting the mass storage needs of an MPP are well documented in [7]. NSL UniTree in its present form will not be able to support an MPP because it only supports one logical stream of data. A new breed of MLFS software that is capable of supporting scalable, parallel storage systems is required. While there are no such products readily available today, any efforts in developing a reasonable solution will be closely tracked.

Current Status

A Request For Proposal for DBCS-0 has been prepared and distributed to companies interested in bidding on providing a solution. Hardware procured for DBCS-0 will include a host computer system with 256 megabytes of memory and 8 gigabytes of SCSI-2 disk space, a HIPPI connected RAID with at least 30 gigabytes of usable disk space and a minimum sustained data transfer rate of 40 megabyte/second, and at least 1200 gigabytes of storage on one or more fast tape robot archives. The fast tape robot archive(s) will have at least four tape drives, each capable a sustained data transfer rate of 2 megabyte/second. The NSL UniTree license procured for the prototype system will be upgraded to support the desired amount of storage and the ObjectStore license will be transferred to the new host computer system. The integrated DBCS-0 system is expected to be delivered in October 1993 and acceptance tests will begin thereafter.

Future Direction

Data that is currently generated in the MSRC is manually managed and maintained by researchers. Pertinent historical information about this data is usually recorded in a laboratory notebook. When disk space shortage mandates, data is manually archived to tape for long term storage via standard backup procedures. These tapes are then physically maintained by the researcher. The degree of reliability in this process is directly dependent on the researcher's ability to maintain and coordinate the tapes and notebook. Over time, inventory control of data in long term storage becomes a time consuming task.

It is expected that researchers will initially be reluctant to relinquish control of their data and its associated historical information (i.e., metadata) to an automated facility such as DBCS-0. Researchers feel secure in having the ability to control the physical media and laboratory notebook at all times. Developing a similar level of trust in an automated system must be achieved in order for DBCS-0 to be successful. Extensive, but fair acceptance tests for DBCS-0 will ensure a high degree of reliability in all hardware and software components. Lessons learned in testing the DBCS prototype system will be used in developing the scripts and procedures for the DBCS-0 acceptance tests. Likewise, all procedures and software developed for DBCS-0 will undergo a rigid set of predefined tests that will ensure all files/datasets and metadata are maintained in a highly reliable manner. Recovery from failures must be handled gracefully.

Like any other system, DBCS-0 will require a certain level of administration. The individual(s) responsible for administering DBCS-0 must be intimately familiar with each hardware and software component. Administration policies will be written to ensure that users' data is maintained in the most reliable manner. Researchers will be polled for their concerns and this input will be incorporated into the administration policies. Regular and routine maintenance of the MLFS and ODBMS will be performed to ensure recovery in the event of system failures. Once developed, the policies will be automated to the extent possible.

The administration policies for DBCS-0 will also account for management of media used for long term storage. To reduce the amount of human intervention in administrating DBCS-0, it is necessary to provide a maximum amount of storage capacity within the robot archive(s) at all times. While redundant copies of datasets/files will be allowed, only the media containing the primary copy of files/datasets will typically reside in the robot for an extended period of time. Media containing secondary copies of files/datasets will periodically be removed from the robot along with the necessary backup of the MLFS software's databases. The number of times each individual media is used will be automatically tracked in DBCS-0. When any media has outlived its expected lifetime, all of the files and datasets it houses will be transferred to new media and the old media will be destroyed. All information supporting the media management policies will be stored in the ODBMS and will be readily available for the DBCS-0 administrator.

Future implementations of DBCS-0 will provide researchers with a Graphical User Interface (GUI) for managing and manipulating files. This interface will allow users to move files and datasets to and from the mass storage system, enter the file/dataset's metadata, and search and browse the metadata for any publicly accessible files/datasets. Researchers will be able to access files and datasets within DBCS-0 by querying the ODBMS for attributes of the data (e.g., all datasets on a "class" of molecule). These features will save a researcher considerable time compared to the present day practice of keeping notes in one's laboratory notebook and will allow researchers to readily exchange information, thereby increasing their overall productivity and effectiveness. While some types of metadata will be entered by the user via the DBCS-0 GUI, it is important that the user not be required to enter any metadata that can be collected automatically (e.g. user's name, date, etc.).

Restricted access to files/datasets and metadata will be supported. Some metadata maintained by researchers are private notes and are not intended to be shared with others. Also, some datasets/files produced or collected may be immature or meant to be used only on an interim basis. This type of information will be protected from unauthorized access by allowing each researcher to determine what data are to be shared with others. Access to a researcher's data may be granted on a user and/or group basis. In order to support this requirement, the underlying MLFS software must support restricted access to data which has been archived.

A toolkit will be provided that will allow developers of scientific applications to manage and manipulate files and datasets from within their code. Access to the advanced features and local extensions of the MLFS software will be provided within this toolkit. Likewise, information in the ODBMS will be accessible through this toolkit. This *back door* into DBCS will provide application developers the means to search the ODBMS for the required information and stage and migrate files/datasets within the mass storage system.

Supporting the features described above requires that the MLFS software and the ODBMS be integrated. A considerable amount of analysis and experimentation is required before attempting this integration. Much work has been done in an attempt to provide extended capabilities and intelligence to existing MLFS software. As an example of one such effort, Isaac describes a prototype in which the UniTree is integrated with a DBMS [8]. A standard Structured Query Language interface is provided for accessing data in the file system. Data are automatically staged from the file system to the DBMS. As Isaac points out, utilizing the Bitfile Identifier found in UniTree's Name Server database for referencing datasets warrants further investigation. This use of a Bitfile Identifier will likely prove an efficient way in which to access files/datasets within DBCS-0. Also, expanding Isaac's concept to include maintaining metadata about each dataset in the DBMS is consistent with the requirements for DBCS and worth investigation. PNL plans to leverage efforts such as these when integrating the MLFS software with the ODBMS.

Phase 3: Database Computer System - Level 1

The third and final phase will yield a production mass storage system (DBCS-1) for the EMSL. The planned November 1995 delivery of DBCS-1 will result in a fully operational system in March 1996. Development and enhancement of DBCS-1 is planned to continue after the system is put into operation. Currently, it is not assumed that DBCS-1 will simply be an extension of DBCS-0. While some components of DBCS-0 may be reused in DBCS-1, emerging technology may dictate that DBCS-1 be an entirely new system. It is imperative that evolving technology in both hardware and software be tracked closely. The specifications for DBCS-1 will be prepared based on information collected during the first two phases, input solicited from EMSL users, recent trends and developments in the mass storage community, etc. It is important that the communication channels with others undertaking similar activities, as well as vendors of mass storage systems/products, be open and active at all times. Future papers will describe this phase in more detail.

Conclusions

Intelligent data archiving services will be provided to researchers in the EMSL in 1996 in the DBCS-1 system. MLFS software will be integrated with an ODBMS to provide researchers with a convenient and efficient way in which to manage their files/datasets and electronically maintain its associated historical information. Emerging technology and information gathered in two previous phases of development will drive the specifications for DBCS-1.

The first phase is currently under way. A DBCS prototype system of limited scale is being used to gain hands-on experience and knowledge of NSL UniTree and ObjectStore software. The second phase, DBCS-0, will expand the capabilities of the prototype system to include more powerful storage devices in multiple dynamic hierarchies later in 1993. DBCS-0 will be faced with the challenge of supporting a variety of clients, including a massively parallel processor.

Acknowledgments

The work described in this paper is based on the efforts of several people at PNL. I wish to acknowledge my colleagues for their input, support, and previous work in this area.

Pacific Northwest Laboratory is operated for the U.S. Department of Energy by Battelle Memorial Institute under contract DE-AC06-76RLO 1830.

The National Storage Laboratory is a collaborative effort of various industry partners and DOE Laboratories at Lawrence Livermore National Laboratory (LLNL). LLNL is operated for the U.S. Department of Energy under contract W-7405-Eng-48.

Bibliography

1. S. Coleman and S. Miller, *Mass Storage System Reference Model Version 4*, IEEE Technical Committee on Mass Storage Systems and Technology, May 1990.

2. D. Nydick, K. Benninger, B. Bosley, J. Ellis, J. Goldick, C. Kirby, M. Levine, C. Maher, and M. Mathis, "An AFS-based Mass Storage System at the Pittsburgh Supercomputer Center," *Digest of Papers, Eleventh IEEE Symposium on Mass Storage Systems*, IEEE Computer Society Press, October 1991.

3. J. Goldick, K. Benninger, W. Brown, C. Kirby, C. Maher, D. Nydick, B. Zumach, "An AFS-Based Supercomputing Environment," *Proceedings, Twelfth IEEE Symposium on Mass Storage Systems*, IEEE Computer Society Press, April 1993.

4. V. Gate, "Alex - a Global Filesystem," *Proceedings of the Usenix File System Workshop*, 1992.

5. M. Rosenblum, J. Ousterhout, "The Design and Implementation of a Log-Structured File System," *Proceedings of the 13th ACM Symposium on Operating Systems Principles*, February 1992.

6. M. Seltzer, K. Bostic, M. McKusick, C. Staelin, "An Implementation of a Log-Structured File System for UNIX," *Proceedings of the 1993 Winter Usenix*, January 1993.

7. S. Coleman, R. Watson, R. Coyne, H. Hulen, "The Emerging Storage Management Paradigm," *Proceedings, Twelfth IEEE Symposium on Mass Storage Systems*, IEEE Computer Society Press, April 1993.

8. D. Isaac, "Hierarchical Storage Management for Relational Databases," *Proceedings, Twelfth IEEE Symposium on Mass Storage Systems*, IEEE Computer Society Press, April 1993.