AIX Storage Management Products Comparison

Document Number GG24-4495-00

June 1995

International Technical Support Organization
Austin Center

Take Note!

Before using this information and the product it supports, be sure to read the general information under “Special Notices” on page xi.

First Edition (June 1995)

This edition applies to Version 1.3 of ADSM/6000, Program Number 5765-203, Version 1.02 of UniTree, Program Number 5696-398, Version 1.02 of FSF/6000, Program Number 5696-708, and Version 4.00 of Legato Networker, Program Number 5765-316 for use with AIX Version 3.2.5 for the RISC System/6000.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

An ITSO Technical Bulletin Evaluation Form for reader's feedback appears facing Chapter 1. If the form has been removed, comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept. 632B Building 821 Internal Zip 2834
11400 Burnet Road
Austin, Texas 78758-3493

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Abstract

This document is intended to provide a functional comparison of the main storage management products currently available for AIX. The focus is not specifically on implementation, though examples and technical details are included. The intent is rather to take a new approach and examine the products within the contexts that they are used. The document is organized into sections covering the major tasks that the products are designed to perform.

The areas for comparison include backup/restore, archive/retrieve, space management, remote file system support, and migration. The products compared are ADSTAR Distributed Storage Manager/6000, UniTree/6000, File Storage Facility/6000 and Legato Networker.

The audience is intended to be IBM personnel and customers who wish to be able to position the products and determine the most appropriate solution for a particular environment. The information contained in this document is also useful as a practical guide to implementing the various functions, though where appropriate, other documents are referenced. Some knowledge of AIX V3.2.5 is assumed.

(240 pages)

Contents

Figures

Tables

Special Notices

This publication is intended to help IBM personnel and customers correctly position the main storage management products currently available for AIX and provide them with basic implementation assistance. The information in this publication is not intended as the specification of any programming interfaces that are provided by ADSM/6000, FSF/6000, UniTree/6000, Legato NetWorker, AIX V3.2.5, or the RISC System/6000. See the PUBLICATIONS section of the IBM Programming Announcement for ADSM/6000, FSF/6000, UniTree/6000, Legato NetWorker, AIX V3.2.5 and the RISC System/6000 for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM (VENDOR) products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

ADSTAR
AIX
AIX/6000
AIXwindows
IBM
InfoExplorer
OS/2
RISC System/6000
RS/6000

The following terms are trademarks of other companies:

Windows is a trademark of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

AT&T	American Telephone and Telegraph Company
Andrew File System, AFS	Transarc Corporation
DataWheel	LAGO Systems
DEC Ultrix 86/Open Desktop	Digital Equipment Corporation
HP/UX	Hewlett Packard Corporation
Macintosh	Apple Corporation
Maximum Strategy RAID	Maximum Strategy, Inc.
Microsoft Windows	Microsoft Corporqtion
Motif	Open Software Foundation Incorporated
NetWare, IPX, SPX	Novell Corporation
POSIX	Institute of Electrical and Electronic Engineers
SCO UNIX 386/Open Desktop	Santa Clara Operation Incorporated
Silicon Graphics	Silicon Graphics Incorporated
Solaris, NFS, SunOS	Sun Microsystems Incorporated
Sony, Sony 68K	Sony Corporation
UniTree	OpenVision Technologies Corporation

Other trademarks are trademarks of their respective companies.

Preface

The purpose of this document is to examine the functions that are required for storage management. All products need to be installed, customized and configured to provide the necessary services for the specific environments in which they will be used. Rather than provide step by step instructions on how to set up and use the main products currently available, the emphasis is on examining each product's usage in the major functional areas that it supports. The intent is to allow a realistic comparison to be made, based on the operational characteristics of the products, and thereby correctly position them and select the most appropriate one for the requirements.

In order to achieve this, the document is organized into chapters covering each of the main functional areas of storage management. Within each chapter, there is an introduction describing the issues and mechanisms available to support the functions. The setup and operation implementation is also covered in some detail, though where appropriate, references are made to existing documentation. The final section in each chapter will provide a comparison summary for the products.

The intended audience is to be IBM personnel and customers who wish to position the various storage management products currently available. This document will primarily assist them in product selection based upon operational requirements, but will also provide basic implementation details.

How This Document is Organized

The document is organized as follows:

- Chapter 1, “Storage Management Concepts”

  This chapter provides a general introduction to storage management. It will cover the technology involved in storing information in a distributed environment, as well as the issues involved. The techniques and mechanisms developed to manage the information will be discussed, including hierarchical storage management, space management and general file system technology.

  This is intended to provide a grounding, so that the purpose behind the design of the storage management products discussed can be properly understood. Those readers already familiar with these concepts may skip this chapter.

- Chapter 2, “Storage Management Products”

  This chapter contains high-level information on the storage management products covered in this document. The purpose of this section is to outline the features, functions and technology available within each product so that readers may gain a background in their capabilities.

  Readers can use this section to quickly gain an understanding of the range of functions provided by a particular product. Those readers not requiring an overview of the products, or who are already familiar with their basic capabilities, may skip this chapter.

- Chapter 3, “Installation and Customization”

  This chapter looks at installation and customization issues relating to storage management products. The basic steps involved in the installation of each

product are shown and the range of customizable options detailed. Specifics on these options are given in this section, where appropriate, and references made to other sections for more detail when relevant. Installation and customization of client and server elements will be covered.

The intent is to allow a comparison to be made in terms of the ease of installation and basic customization, as well as the range of options available.

- Chapter 4, “Product Administration”

  This chapter looks at the various administration tasks required. The most common procedures, such as defining and managing users and devices, are discussed and examples given. In addition, other important features, such as security, are examined as well as the support available in each product.

  General administrative tasks common to every environment are discussed in this section to allow comparisons to be made between ease of use and range of function. Basic information on implementing these tasks is included.

- Chapter 5, “Backup and Restore”

  This chapter begins by looking at the various backup possibilities available, including user/system initiated, scheduled and incremental.

  Details on the setup and management of the products are included, the object being to allow comparisons of the features available, as well as the ease of management and recovery for a range of potential problems.

- Chapter 6, “Archive and Retrieve”

  This chapter looks at archival and retrieval of information. The various facilities available are discussed and the setup and operation of the storage management products is then covered for these scenarios.

  Once again, the purpose is to allow a functional, management and ease-of-use comparison to be made.

- Chapter 7, “Space Management”

  This chapter is concerned with space management services that can be provided for client systems. The principles of space management are discussed as well as scenarios where it may be used. The steps required to set up and use space management functionality are shown for those products that support this feature.

  This chapter also demonstrates the flexibility of the support provided in terms of how a product may integrate with others, as well as highlighting ease of use, management and setup issues.

- Chapter 8, “Remote File Systems”

  This chapter looks at the requirement for storage management products to support remote file systems. Several scenarios, involving various types of remote file system, are described, and the ability of the various storage management products to support them are discussed.

  This chapter does not cover implementation details of remote file systems, nor does it reiterate the procedures for setting up the products to interact with file systems generally, unless there are specific differences for the remote file system.

- Chapter 9, “Server Hierarchical Storage Management”

This chapter looks at the process of defining storage hierarchies and the migration of data between them. The principles of hierarchical storage management and the reasons why it is used, are also covered. The steps necessary to set up those products that implement this feature are included.

- Appendix A, “Supported Platforms”

  This appendix contains a list of the storage management products covered in this document and the platforms supported by their client components.

- Appendix B, “Supported Devices”

  This appendix contains a list of the devices supported by the products covered in this document.

- Appendix C, “Storage Products Summary”

  This appendix contains a list summarizing the functions supported by each of the products covered in this document.

Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this document.

- *ADSM/6000 Installing the Server and Administrative Client*, SH26-4013
- *ADSM/6000 Administrator's Guide*, SH26-4005
- *ADSM/6000 Administrator's Reference*, SH26-4006
- *ADSM User's Guide and Reference for UNIX*, SH35-0120
- *AIX FSF/6000 Installation, Planning and User's Guide*, SC23-2587
- *UniTree Installation and Planning Guide*, SC23-2496
- *UniTree User's Guide*, SC23-2497
- *UniTree System Administrator's Guide*, SC23-2498
- *UniTree Problem Determination Guide*, SC23-2499
- *Legato NetWorker Installation and Maintenance Guide*, Phone (415) 812-6000
- *Legato NetWorker Administrative Guide*, Phone (415) 812-6000
- *Legato NetWorker User's Guide*, Phone (415) 812-6000
- *DFSMS/MVS Network File System User's Guide*, SC26-7028

International Technical Support Organization Publications

- *AIX Storage Management*, GG24-4484
- *Getting Started with ADSM/6000*, GG24-4421
- *ADSM Storage Management Services: Implementation Examples*, GG24-4034
- *ADSM Advanced Implementation Experiences*, GG24-4221
- *Using ADSM to Back Up Databases*, GG24-4335
- *Local-Area Network Backup and Recovery Using SaveUtility/2 and Legato Networker*, GG24-4180
- *Enterprise-Wide Client/Server Recovery Management*, GG24-4414

- *Managing One or More AIX Systems - Overview*, GG24-4160

A complete list of International Technical Support Organization publications, with a brief description of each, may be found in:

International Technical Support Organization Bibliography of Redbooks. GG24-3070.

To get a catalog of ITSO technical publications (known as “redbooks”), VNET users may type:

```
TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG
```

How to Order ITSO Technical Publications

IBM employees in the USA may order ITSO books and CD-ROMs using PUBORDER. Customers in the USA may order by calling 1-800-879-2755 or by faxing 1-800-284-4721. Visa and Master Cards are accepted. Outside the USA, customers should contact their local IBM office.

Customers may order hardcopy ITSO books individually or in customized sets, called GBOFs, which relate to specific functions of interest. IBM employees and customers may also order ITSO books in online format on CD-ROM collections, which contain books on a variety of products.

Acknowledgments

This project was designed and managed by:

Nick Higham
International Technical Support Organization, Austin Center

The authors of this document are:

Nick Higham
IBM UK

Marina Russo
IBM Italy

This publication is the result of a residency conducted at the International Technical Support Organization, Austin Center.

Thanks to the following people for the invaluable advice and guidance provided in the production of this document:

Cyndie Behrens
International Technical Support Organization, San Jose Center

Tim Boyce
IBM Austin

Chapter 1. Storage Management Concepts

This chapter provides a general overview of storage management. The intent is to provide a context within which the operation of the products discussed in this document can be understood.

1.1 The Storage Environment

The computing environment exists to provide users with the capability to do work. This work involves the processing of information provided from various sources, by applications, to produce output for the user. The input information can come from diverse sources, including:

- User supplied

  Information provided to the application by the user, such as budget figures, banking account information, cost information, retail figures, and scientific data.

- Peripheral supplied

  Information provided by other peripheral devices such as real-time sensing devices, bar code readers, automated teller machines, and hand held terminals.

- Application generated

  Information generated as a result of an intermediate application processing information from sources already mentioned. Examples include account balance information, sales forecasts, stock ordering information, and weather forecasting information.

Almost all of the input information, the applications themselves, and almost always, the output information will need to be stored somewhere within the computer system. This storage may be temporary, as in the case of some transient input data, or it may be long term, as in the case of bank account details or tax information.

Furthermore, the environments in which this work takes place can be divided into two classes:

1.1.1 Local Environments

In a local environment, the input, processing and output all take place on a single machine. This type of environment is becoming less common with the increased requirements for most business applications and users to share information. Areas where this kind of setup do still exist include standalone machines in small businesses and large, central-site computing facilities.

In both cases, with all of the businesses information in one place, protection of the information is vital.

Efficient use of the storage space available is also extremely important as there are upper limits to the amount of directly accessible fast storage on a system (see 1.3.1, “Hardware” on page 8 for a discussion of the attributes of storage devices).

Lastly, availability of the information for use by the business is equally vital as failure in the storage subsystem will prevent any work from continuing.

1.1.2 Distributed Environments

Distributed environments consist of many machines interconnected into some form of network. Machines in the network can be further subdivided into two types:

- Servers

  Server machines provide services to clients in the network. These services may include applications, data files, disk storage, or printing services, for example. Servers, therefore, generally support the applications, data, and devices that are vital to the business.

- Clients

  Clients provide local processing for users, but may access servers for other functions. Client disk storage may actually be implemented at a remote server machine for example, or a server may manage the client disk space using its own disk to vastly increase data storage capability (see 1.3.2, “Software” on page 11 for a description of space management). Clients, therefore, generally may contain some vital business data and will usually contain information vital to the local users of the machine, such as mail, calendar, diary, and other data.

Sometimes machines in a network act as both servers and clients, providing services to some machines, and using services from others.

1.2 Storage Management Requirements

Assuming that a local environment can be classified as an instance of a machine performing both client and server function, the distributed environment presents a situation with the following issues:

1. Defining storage requirements

   The overall requirements for storage must be defined in terms of the following:

   - Performance

     How fast does access to data need to be? This can also be linked to frequency of access. Data which is accessed infrequently usually does not need such rapid response as data that is used very regularly.

     These decisions govern the type of media that the data should be stored on, and to some extent, its location. High frequency, low response time data should be stored on fast disk devices; lower frequency, higher response time data may be stored on optical disks. Very infrequently accessed information can be stored on tape. In terms of location, devices that must be accessed across a network will perform less well than locally attached devices; so, if response time is critical, locally attached devices may need to be considered.

   - Availability

     How critical is immediate access to the information? This will depend, to a large extent, upon the business. Some businesses can last longer than others without access to information. A distribution operation, for example, may be unable to locate items or schedule transport without continuous access to the information, particularly if the inventory, dispatch and stock location are all automated.

These decisions govern both the type of media and the access type. Very high availability would probably require some sort of fault-tolerant RAID array. Operating system mirroring may suffice for less stringent requirements, and businesses that can survive a day or two may only require restoration from backups.

- Capacity

  Simply speaking, this is the overall volume of storage required by the operation. The ideal environment would have all information stored in fault-tolerant main memory. Unfortunately, this is slightly out of the question, due to technology and cost constraints. Cost also usually prohibits storing all information locally on fast disk devices, and server machines can run into cost, performance and technology constraints if required to support large numbers of clients solely with fast disk devices.

  The decisions here usually result in defining some sort of space management, hierarchical storage implementation or at least some form of server implementation. These concepts are covered more fully in 1.3.2, “Software” on page 11, and usually include using different media types concurrently to spread the load and reduce the cost.

- Information sharing

  There is almost always a requirement between users or departments within a business to share information, and the simplest way to institute this may often be to provide concurrent access to the same storage.

  This requirement can be fulfilled, for example, by maintaining information that needs to be shared on a server disk that can be accessed by multiple clients. This applies equally to applications and data. Other considerations should also be taken into account, such as integrity and security of the information.

2. Managing storage space

   Once the preceding requirements have been defined and implemented, the storage must be managed. This involves making the various resources available on the machines - defining tapes, optical devices, disks, and libraries. Next, the volume groups, logical volumes and file systems must be designed and created. The organization of the volume groups and logical volumes is important and can reduce the effect of failures in storage subsystems, as well as reducing recovery time. Operating system data can be kept in a separate volume group from user data for example, and different department or application set information can also be maintained separately. The resulting design will depend on the specific requirements of the environment, and a more detailed description of the design and implementation process can be found in the *AIX Storage Management* redbook.

   Once all of the file systems, logical volumes, volume groups, and physical hardware are in place, access needs to be given to any clients that will be accessing file systems remotely. This involves setting up communications and using the Network File System (NFS) to export the remote file systems and mount them from the clients for access. Various types of remote file system can be accessed from a variety of different manufacturer's machines, including NFS and the Andrew File System (AFS), for example.

   Now that access to the required applications and services is available, users can begin doing useful work for the business. This always results in the

production of data, which begins to use up space in the storage subsystems. Space management applications, hierarchical storage management applications and archival of infrequently accessed information can all assist in reducing capacity requirements. These are all discussed in more detail in 1.2.1, “Services” and 1.3.2, “Software” on page 11.

3. Protecting data

   Now that data critical to the business is being produced, provision must be made for failures in the system. As technology becomes ever more reliable, the likelihood of failures becomes less and less, but never reaches zero and certainly does not exclude accident. As such, mechanisms for recovery in the event of loss of data need to be designed and strategies for the protection of data implemented.

   This kind of planning is in addition to any enhanced availability designed into the system. Even though a RAID array can survive power supply failure, controller failure, and even continue to operate with a disk failure, this does not take accident into account. Thus, some form of backup and recovery plan must be developed and implemented. Backup and recovery is discussed further in 1.2.1, “Services.”

4. Providing device access

   In order for devices to be used for storing information, whether from backups, archives, or just everyday usage, there has to be some form of interface to the device. Many disk and tape devices are directly supported by the operating system, but there are some devices, particularly libraries and OEM tape, optical jukebox, and disk devices, that are not. In these cases, the products may require some higher-level software product that incorporates the necessary lower-level interfaces that allow them to be used.

   This kind of consideration and planning needs to be effected in advance so that the storage system design can actually work. It may require the ordering and integration of additional products, such as those described in this document.

This completes an overview of the storage environment and requirements. The rest of this section will look in more detail at the services that have been mentioned and which form the main function of the products that are the subject of this document. A look at device characteristics will follow.

1.2.1 Services

The products discussed in this document are designed to assist with the process of storage management. From the preceding overview, it can be seen that once a computer environment has been set up, there are a number of ongoing tasks that must be implemented in order to maintain it. While administration, monitoring and resolving problems are major tasks, the most essential tasks from the overall business perspective are:

- Backup and recovery
- Archive and retrieve
- Space management

and these will now be examined in more detail.

1.2.1.1 Backup/Recovery

As has been mentioned, providing additional copies of vital data is essential to protect against failure or accident. This process is known as backup and involves the planning and implementation of a strategy to guard, as fully as possible, against catastrophe. Not every possible contingency can be covered, and furthermore, to ensure complete recovery, a new copy would need to be made of every piece of data the instant that any change occurred. This would obviously be impractical; therefore the plan needs to be a compromise.

There are three main kinds of backup:

- Full backups

  This is where all data in the system is copied to backup media, thereby ensuring complete recovery to this state in the event of failure. This normally takes some time to complete. Exactly how long depends on the total volume of information, usually at least several hours.

  Accidental erasures or small losses of data can still be recovered without needing to restore the entire system, but normally less easily.

- Incremental backups

  This is where a full backup is taken at a predetermined interval, say once a month. Every week, a further *incremental* backup of data that has changed since the last backup is taken. This normally involves far less data than the full backup. The exact periods depend again upon the value of the data being created. Incremental backups should be taken once a day and full backups once a week if loss of more than a days worth of data means collapse of the business.

  Recovery involves reloading the last full backup and then applying the incremental backups in sequence up to the failure point.

  Accidental erasure of files can be recovered, but may require reloading the original backup and then, in the worst case, applying all of the incrementals.

- Progressive backups

  This is similar to an incremental backup. The difference being that where incremental backups copy all changes since the last backup, incremental or full, each progressive backup copies all changed data since the last full backup. This means that progressive backups are quicker to recover with, but require more storage and take a little more time.

The decisions about when to do it and what sort of backup to use (usually a mix of the possibilities) are generally business related. The actual implementation can be automated, but when the environment is distributed this becomes enormously complicated. This is where higher-level tools, such as some of those discussed in this document, come in. They provide the ability to centrally manage automated or user-initiated full, incremental or selective backups in a networked environment. Multiple copies of data can be held, and a variety of devices can be used to store the data, including intermediate optical storage and many kinds of tape library or standalone device.

1.2.1.2 Archive/Retrieve

Archival of data involves moving the information to less expensive storage. The purpose of archiving is to free up space on fast, expensive disk storage used by information that will be rarely, if ever, accessed again but which must be maintained accessible, just in case. The information is usually copied to inexpensive mass storage, such as tape, from where it can be retrieved if ever needed again. With the advent of cheap, high-capacity optical storage, infrequently accessed information, such as images, is increasingly being archived to this medium. Retrieval is quicker from optical storage, though both optical and tape archived data will have longer access time.

A second usage of archiving is for preserving snapshots of information, a *version* of a file perhaps. This can allow regression back to earlier versions in the event of errors as well as preserving earlier versions for support reasons.

With multiple users and many machines in a distributed environment, managing this can become a complex task. Once again, higher-level tools can provide a centralized, automated mechanism for enabling this function. Multiple device types can be supported as the destination of archive data, and archive can be integrated in some cases with backup; a copy of archived information automatically taken, for example.

1.2.1.3 Space Management

The third essential task that must be carried out is some form of space management. It is inevitable that the amount of space required by user applications and data will grow to meet, and then exceed, the amount of space available. Adding more and more expensive disk storage is one solution to this problem, but space management can offer a far less expensive alternative.

Space management comprises monitoring information in user storage, and, based on a set of predefined criteria, moving certain data from this storage space to some form of server storage pool. A pointer of some sort is left in the user storage space; so, to all intents and purposes, the information is still there, though the space it occupied is now available for other data. Criteria include things like the age of the file, last access, size, and user storage utilization thresholds.

With difficulty, this task could be done manually, but in a complex distributed environment, this becomes increasingly hard to manage. Once again, automatic mechanisms for implementing this function are available, and some of the products discussed in this document provide centrally managed services to accomplish this task.

1.2.2 Devices

All of the mechanisms and tasks discussed thus far are related to the management of storage. The actual implementation of the storage itself is also vitally important, and this section looks at the main criteria that differentiate the various storage devices available.

The following section will examine the technologies themselves. A complete examination of all of these points is available in the *AIX Storage Management* redbook.

1.2.2.1 Performance

The performance of a device in the storage environment can be defined as a measure of the throughput of information to and from the device. The faster information can be written to and read from the device, the higher the performance. Throughput itself is usually defined for two cases:

1. Instantaneous or burst

   This is the data rate or throughput that the device is capable of at peak transfer of small quantities of data over short intervals. It is usually higher than the sustained data rate.

2. Sustained

   This is the average throughput of the device over a longer period of time. It is usually lower than the instantaneous transfer rate because other factors, such as bus contention and various latencies, must be taken into account.

High-performance devices are normally used for storage of data that will be used interactively or requires rapid response time. Lower-performance devices tend to be used for archive-type operations, with devices used for backup requiring the least rapid response time. Obviously, the faster any device is the better, but other criteria such as capacity and cost govern the final choice of product for a specific requirement. Disk, for example, is high performance, reasonably high capacity, but cost per byte is also very high compared to other storage devices; so it would not really be suitable for backup.

1.2.2.2 Availability

The availability of a device defines the extent of the device's ability to continue functioning in the event of component failure. This is usually achieved by providing redundant backups of vital components that can take over function from the primary components if they should fail. The level of redundancy is directly related to the availability of the device; one with complete redundancy, that is to say backup components for all major parts, is known as a fault tolerant device. Availability is also achieved through technology, such as RAID, where data is mirrored, or written with parity, for example, allowing the original information to be reconstructed should a single disk fail. This again relies on redundancy in the disk array.

As with performance, availability is a trade-off against price. The higher the level of redundancy and sophistication, the greater the price. Vital online information is probably best protected with highly available disk, such as a RAID array, whereas backup protection tends to take the form of multiple copies of the information.

1.2.2.3 Capacity

The capacity of a device basically defines the amount of information that it is capable of storing. This is mainly dependent upon the technology. The latest disk devices have capacities up to 4GB in a single unit; optical disks have capacities around 1GB per disk, and tape cartridges can have capacities of up to 10GB, with compression. Again, other factors must be taken in to account, such as cost per byte, performance and availability. In addition, tape and optical libraries can massively expand available capacity, though the other factors need to be considered.

1.2.2.4 Function

The functionality of a device in the storage environment defines the operational characteristics of the medium and includes reading, writing, seeking, sequential access and random access. Applications will have different requirements in these areas with regard to their data access; some may only need to read data, some read and write, some may need random access to database information, and some may only need to write sequential output data. Different devices support different combinations of these functions. For example, disk devices can usually support sequential and random read and write, while some optical devices are read only, and tape devices do not support random access.

1.3 Storage System Technology

This section will complete the storage management overview by looking at the actual hardware and software products that implement the functions described so far.

1.3.1 Hardware

The following topics will briefly examine the current hardware technologies available in terms of the properties discussed earlier in this chapter. More information on hardware can be found in the *AIX Storage Management* redbook.

1.3.1.1 Disk Devices

Disk devices use one or more platters coated with a magnetic material that are attached to a central spindle and rotated past read and write heads. The read and write heads (sometimes combined in a head) are attached to a mechanical arm called an actuator that moves them radially with respect to the disk surfaces. In this way, data is written as a series of concentric tracks. The time taken to read data is a combination of the time taken for the head to be positioned over the correct track (seek time), the time taken for the correct part of track to rotate under the head (rotational latency), and the time taken read the data from the disk surface (data rate from the disk). The time taken to read data is a combination of the seek time, rotational latency and data rate to the disk. Some disks also incorporate additional performance-related algorithms, such as elevator seeking, where a number of requests will be sorted by track order so that they can be satisfied on a single seek pass. The combination of these things define the performance of the disk.

Banding defines a mechanism whereby the disk surfaces are split into a number of concentric regions known as bands. Those bands further away from the center of the disk will have larger area, and disks using this technology increase the bit rate in each successive outward band. The result is that the bands further out from the center have a similar bit density to those closer in; therefore more data can be stored. The capacity of a disk depends upon the areal (bit) density, and this depends upon techniques like banding and the technology used to construct the head and magnetic surface. The number of platters in the drive increases the physical size of the drive as well as the capacity.

Availability, in terms of individual disk drives, depends largely upon the quality of construction, and hence the Mean Time Between Failures (MTBF) for the drive. A higher MTBF means a more reliable drive. Availability can be enhanced by using the drive in an array or subsystem which can provide redundant power supplies, cooling fans, hot pluggability for the drives, and maybe some form of RAID.

Disks are the highest-performance storage devices outside of main memory. High capacity can be achieved, but at relatively high cost. Availability can be assured with some form of array or subsystem.

1.3.1.2 Optical Devices

Optical devices fall into a number of different categories, depending mainly upon the technology used:

1. CD-ROM

   In Compact Disk-Read Only Memory optical devices, the information is molded into the media when the disk is manufactured as a sequence of pits in the surface. The existence or absence of a pit determines the binary state one or zero. The disk is spun at high speed inside the device, and a low intensity laser is used to read back the information from the disk surface. The laser synchronizes with the start of the track and is then moved radially in and out to access different parts in much the same way as magnetic disk.

   Performance and capacity are reasonably low for a single optical disk. The advantage of CD-ROM is in the length of time that data can be stored on the media, and this makes it useful as an archive medium. CD-ROM is also used for information distribution.

2. Rewritable

   With rewritable optical disks, magneto-optical technology is used to store the information. The media surface is coated with concentric tracks of magnetic material and a combination of an electromagnet and a laser is used to read, write and erase information. The magnetic state of the track can be changed by heating up the material with a high-intensity laser and then applying a magnetic field with the electromagnet. Before a rewritable disk is used, it is prepared by setting the state of all the magnetic domains in the tracks representing bits to the same polarity. Recording is then effected by changing the state of domains that are to represent ones using the magneto-optical process described earlier. Different polarity domains polarize low-intensity laser light differently; so the information is read back using a low-power laser and passing the reflected light through a polarized filter. Erasure is essentially the same process as preparation.

   Performance is superior to CD-ROM, comparable to uncompressed tape data rates. Capacity is higher but nowhere near tape or disk. Again, the data life of the media is extremely long which makes rewritable optical media excellent for archive use. The advent of optical libraries allowing greatly increased storage also makes rewritable optical media useful for storing interactive data with lower access requirements.

3. WORM

   With WORM or Write Once Read Many technology, again there are a number of different implementations:

   - Ablative

     This technology uses a high-power laser to physically burn pits into the media surface much like those in CD-ROM technology. The absence or presence of a pit indicates the binary state, and the information is read back by measuring the reflected intensity from a low-powered laser.

   - CCW

With CCW or Continuous Composite Write-once, the same magneto-optical technology used in rewritable optical devices is employed. In order to provide WORM behavior, erasure of the information is simply prevented through firmware.

- Phase Change

  This mechanism uses the same principles as ablative WORM; though in this case, a high-powered laser alters the physical properties of the material used to form the media surface, causing it to adopt a crystalline structure. The altered form reflects low-powered laser light at a different intensity, allowing the information to be read back.

- Dye-Polymer

  Also similar to ablative and phase-change technologies, this mechanism uses the high-powered laser to alter the chemical properties of an organic dye that is coated onto the media surface. When exposed to the high-powered beam, the dye absorbs energy and becomes darker. A low-powered laser is then used to read back information as differences in reflected intensity.

WORM technologies have similar operating characteristics to rewritable optical media.

4. Multifunction

   Finally, multifunction devices incorporate the capabilities of both magneto-optical and WORM devices to allow both technologies to be used in the same drive.

   As a result, these drives exhibit similar operating characteristics to WORM and magneto-optical devices.

1.3.1.3 Tape Devices

There are two basic technologies involved in the manufacture of tape devices though both use the same principles for reading and writing data to the media. However it is packaged, and whatever materials are used for its construction, tape consists of a long strip of material ranging from 4mm wide to half an inch. The strip is coated on one side with a magnetic material, similar to that used in disk manufacture. The strip of tape is then passed in front of read/write heads, also similar to those used in disk devices, and information written and read by altering or sensing the polarity of the magnetic domains on the media surface.

The differences in the two technologies are in the ways in which the information is written to the tape surface, the mechanisms used for transporting the tape past the head and the read/write heads themselves.

1. Helical scan

   With helical scan technology, the tape surface is wound around a cylindrical head that is inclined at an angle to the direction of tape travel. A complex tape transport mechanism extracts the tape from the cartridge and winds it partially around the cylinder. The cylinder contains one or two sets of read/write heads and is spun at high speed while the tape is moved past the spinning head. Due to the angle of the cylinder and the spacing of the heads inside it, tracks are written onto the tape at an oblique angle.

   This mechanism makes very efficient use of the tape capacity and has a good data rate for continuous operation. This is at the cost of start/stop performance

and high mechanical complexity. In addition, since the cylinder is in contact with the tape surface, the tape wears out more rapidly.

2. Longitudinal Recording

   With longitudinal recording, the tape is moved past a stationary read/write head causing tracks to be written longitudinally down the length of the tape. Some designs provide for multiple read/write elements in the head, allowing several tracks to be simultaneously written in parallel. Other implementations allow the head to be stepped perpendicular to the direction of tape travel, allowing multiple tracks to be interleaved, giving much higher capacity. The combination of these two approaches is known as serpentine track interleaving.

   Capacity is not as high as helical scan, but performance is very good. The process is non-destructive as the tape surface does not need to be in such sustained contact with the head; media life is consequently longer. Simpler transport mechanisms can also be employed which results in higher overall reliability.

Both technologies make use of data compression, giving even higher cartridge capacities. Performance ranges from roughly equivalent to optical to that approaching disk devices for the fastest tapes. However, access to information remains sequential.

1.3.2 Software

The following topics will look at the software products available that assist in implementing the storage management procedures discussed earlier in this chapter. This overview includes a look at the elements of the operating system that manage storage as well as higher-level tools.

1.3.2.1 Operating System

The operating system of a computer exists to provide an environment in which the hardware resources of a system can be usefully used to perform work on behalf of users. In order to do this, the operating system is comprised of many elements that manage the various parts of the computer system and provide services that enable applications and users to do work. This section will focus on those particular elements that relate to storage management, namely:

1. The logical volume manager
2. File systems
3. User and application services

More detailed information on the operating system and those elements that pertain to storage management can be found in the *AIX Storage Management* redbook.

Logical Volume Manager: The logical volume manager (LVM) defines a higher-level interface, transparent to applications and users, that allows the division, allocation and management of storage space. This interface is implemented as a set of operating system commands, device drivers and tools that collectively comprise the LVM.

The LVM itself defines a number of entities that form the basis for managing secondary storage space within the system:

- Physical volumes

This entity is the basic element of a storage subsystem. It is a physical magnetic or optical disk.

- Volume groups

  Physical volumes (PVs) are collected together into an entity known as a volume group (VG). A PV can only be in one VG at a time. A VG therefore defines a large pool of storage space, and many system functions, such as some backup commands, can operate on VGs.

- Physical partitions

  The storage space in a VG is divided into a number of equally sized chunks known as physical partitions (PPs). A PP is the smallest piece of allocatable storage in a VG.

- Logical partitions

  In order to implement storage management functions, a higher level of abstraction known as a logical partition (LP) is defined. LPs map to PPs, but the mapping does not have to be 1:1 or indeed, even sequential. An LP can map to up to three PPs, and a sequence of LPs can point to PPs scattered anywhere in a VG.

- Logical volumes

  Generally the first level of application-addressable storage space, a logical volume (LV) consists of a sequence of LPs and thereby defines an area of disk storage. Multiple copies of this area can exist by defining multiple mappings of the LPs to PPs in the VG. For example, if each LV maps to two PPs, then two identical copies exist of the information in the LV. This is known as mirroring.

  Applications can write directly to LVs as a randomly accessible, arbitrarily sized storage space without having to be concerned with which disks the data is on or whether mirroring is being implemented for availability purposes. Most applications use a higher level of abstraction known as a file system (described in the next section), but some applications, such as databases, or higher-level storage management products use an LV directly because they wish to format the space in their own way.

File Systems: In order to allow for organization of data and provide enhanced storage management function, the operating system defines a further level of abstraction know as a file system. A file system is basically just that, an environment for the organization and management of files.

There are many different kinds though most UNIX file systems are hierarchical in nature. This means the file system provides the ability to organize the LV into a hierarchical file tree. The top level of this tree is called the root directory, and files can be stored in this directory. Branches of this tree are known as subdirectories and can be created in the root directory. Each subdirectory can have further files and/or subdirectories within it, thereby providing an organizational mechanism. Operations can be performed on collections of files within a subdirectory, on the hierarchy down from that subdirectory or on an entire file system. Navigation of the file system is accomplished by traversing the directory structure or moving from subdirectory to subdirectory up and down in the hierarchy.

In order to use a file system, it must be mounted. Mounting means locating the root directory of the file system at a mount point or access point so that it can be traversed. When a system is first powered up, it will have a default file system

structure containing the operating system and any licensed programs loaded. Thus there will always be a main root directory. All other file systems will be mounted into this file system in order to access them, generating a single hierarchy. Mount points are subdirectories that do not have to be empty though once a new file system is mounted over a subdirectory, the original subdirectory can no longer be traversed.

The converse of mounting is unmounting, and access to mounted file systems can be discontinued by unmounting or removing the access point. In the case of a mount point that contained information originally, this means the information is accessible again.

As was mentioned earlier, there are many kinds of file systems, and the main ones that will be mentioned in this document include:

- Journaled File System

  The main file system implemented under AIX is known as the journaled file system (JFS). The JFS defines a hierarchical file system, as previously described, but with the addition of journaling of all transactions that occur within the file system. This allows for enhanced recovery in the event of any failures during normal operations. The journal contains a log of all transactions that have completed since the last consistent state of the file system, as well as those in progress. If there should be a power failure, for example, on recovery the JFS can be returned to a consistent state by applying all of the completed transactions in the JFS log.

- Network File System

  Using the Network File System (NFS), it is possible to mount file systems in a machine A, that physically exist on another machine B, and access them as if they were local to machine A. With appropriate configuration, any file system on machine B can be exported, that is to say made available to NFS, and then mounted on any other machine with NFS in a network.

  NFS servers have been written for many types of operating systems, and as a result, file management structures, minidisks from VM for example, can be mounted and accessed from an AIX system.

  It is important to note that many operating system commands rely on the availability of a standard UNIX file system structure. Thus, it is sometimes the case that although a foreign file system can be mounted and accessed, some operations will not work.

  The issue of security is also important, and access to remote data should be managed carefully.

- Migratable File System

  Space management has been discussed earlier in this chapter, and one mechanism that has been defined to assist in the implementation is the migratable file system (MFS). A MFS is effectively mounted over an existing file system and intercepts standard file management commands in order to provide management functions. In this way, it is able to locate data in other places while making it appear to the local users and applications that the data is resident in the local file system. As an example, as the local file system approaches full, the MFS can copy large local files to another system, maybe using NFS, but intercept local calls to list the files and add a pointer to the migrated file. In this way, the migrated file appears to reside locally, but its

space has been freed up for use by other data. If the file is then accessed, say with an open command, the MFS intercepts this call and can copy the file transparently back, perhaps moving other files to make room.

There are a number of different MFSs, and they use different mechanisms though the principle is basically as has been described.

- Andrew File System

  Similar to the purpose behind the NFS, the Andrew File System (AFS), provides a mechanism for accessing remote files. The implementation is different, however. The AFS defines a distributed file system where individual file systems on AFS servers are integrated to provide a large virtual file system that can be mounted on client machines. The file system is seen as one integrated whole, but is usually spread across many server machines in a network.

  AFS also requires code on the client and server machines to implement the functionality.

- Other file systems

  As has already been alluded to, there are many other types of file systems implemented on many other types of operating systems. This is really only of consequence when the data in those file systems requires managing by some local process. The type of access will usually be NFS or via some local higher-level tool that provides an alternative access method. Some of the types of file systems that will be discussed later in this document include:

  – VM minidisks

    Under VM, storage is provided in the form of minidisks which users access and use for data and application storage. Minidisks are the basic component of storage and there is no further subdivision. In terms of NFS access, a minidisk is the exportable unit which can be mounted into an AIX file system.

    Further information on accessing VM minidisks in relation to the products discussed in this document can be found in Chapter 8, “Remote File Systems” on page 191.

  – MVS partitioned data sets

    Under MVS, Partitioned Data Sets (PDSs) can also be mounted using NFS. The PDS defines a two level hierarchy which allows a single level of subdirectories from the AIX point of view.

    More information on MVS PDSs and their relationship to the products discussed in this document can be found in Chapter 8, “Remote File Systems” on page 191.

  – MVS Hierarchical File System

    Also under MVS, there is the OpenEdition Hierarchical File System (OE HFS). This defines a POSIX-compliant hierarchical file system that is compatible with other POSIX-compliant hierarchical file systems. The organization of the OE HFS is the same as that described earlier in this section for general file systems. The OE HFS can be mounted and accessed from an AIX file system.

More information on the OE HFS and its relationship to the products discussed in this document can be found in Chapter 8, “Remote File Systems” on page 191.

User Services: There are a number of services provided by the operating system that allow implementation of storage management procedures by users or system administrators:

- Backup and restore

  The operating system provides several commands that allow information in the storage subsystems to be backed up by users or system administrators. These commands include the following capabilities:

  – Entire system backup/restore
  – Volume group backup/restore
  – File system backup/restore
  – Directory backup/restore
  – Individual file backup/restore
  – Remote system backups
  – Incremental backups

  These operations can be implemented from the command line, and some of them have System Management Interface Tool (SMIT) interfaces.

- Archive and retrieval

  The operating system also provides commands that allow information to be archived and retrieved. Archive can be to many types of devices, including disk, optical, tape, and diskette. Archive can also be by entire file system, subdirectory tree or even individual file.

- Storage space management

  Operating system commands such as `find` allow system administrators to search file systems and purge old information or automatically archive files that have been unused for some time.

In addition, using facilities such as the `cron` scheduler, it is possible to automate many of these functions.

More information on operating system support for these functions can be found in the *AIX Storage Management* redbook.

1.3.2.2 Higher-Level Tools

Supplementary to the commands provided by the operating system, and in many cases making use of its features, are higher-level storage management tools. These applications are designed to augment the capabilities provided by the operating system, integrate functions and provide a generally easy-to-use mechanism for storage management.

Among the enhancements provided are the following:

- Hierarchical storage management
- Space management

- Backup and archive services
- Ease of use

Hierarchical Storage Management: Most information that is created is initially stored on fast magnetic disk for interactive access. As time goes by, the amount of information accumulates and access to older information may become more infrequent. Gradually, the available disk space becomes used up, and rather than maintain relatively unused information taking up costly room on disk, another alternative is hierarchical storage management (HSM). With HSM, storage hierarchies are defined with each level being a pool of storage space. A typical hierarchy may consist of a fast disk pool, a medium speed optical pool and a sequential access tape pool. The levels in the hierarchy are linked, and when a higher level pool usage reaches a certain threshold, information from it (based on certain criteria, such as usage) is automatically migrated to the next level down (in this case optical). Should the optical pool approach its usage threshold, information in it can be migrated down to the next level, in this case a tape pool.

In this way, storage space is efficiently used, and the information most commonly required remains on the fastest available media. Should a request occur for a migrated file, it can be automatically migrated back up the hierarchy to the top for access.

Access to intermediate levels for archive or backup services can usually also be supported.

The number of levels and their composition is arbitrary and normally based on the requirements of the environment.

Products providing hierarchical storage are discussed in Chapter 9, “Server Hierarchical Storage Management” on page 209.

Space Management: It is usually impractical to set up hierarchical storage on every machine in a network, both from a cost and administrative point of view. An alternative to managing the expensive disk space for other machines in the network is to implement some form of space management for those machines. Due to the fact that this service effectively creates a hierarchy involving a client disk (top level) and server disk (lower levels), it is sometimes called Client Hierarchical Storage Management. Indeed the ADSM Version 2 space management product is called HSM. In order to avoid confusion, it is a good idea to think of Hierarchical Storage Management, as defined in the previous section, as Server Hierarchical Storage Management.

Space management involves providing a mechanism to monitor disk space utilization on behalf of client machines and the use of some method to ensure continued space availability. This is usually implemented with some form of MFS on the client system that transparently manages the local disk. Based on criteria, such as file age, file size or disk utilization thresholds, information is moved from the client disk to a server machine. From the client point of view, all information continues to appear to reside locally though many files may in fact be physically resident at the server. If required, those remotely copied files can be transparently returned. In this way, small client disks can be made to seem much larger.

Space management mechanisms can interact with hierarchical storage at the server and use pools in the hierarchy to store client data.

Space management typically also includes facilities to cause important files to be marked ineligible for migration as well as commands to force migration of files.

Products providing space management are discussed in Chapter 7, “Space Management” on page 173.

Backup and Archive Services: Although the operating system provides backup and archive capabilities, and indeed, allows remote operations and a degree of automation, much configuration and shell script programming would be required to create an effective system. Higher-level tools, usually designed to operate in a distributed environment, can provide these services in a more structured and easy-to-use form.

Backups can be scheduled centrally to happen at specific times or regular intervals and at various levels for clients. Multiple archive copies can be requested to allow versioning. Backup and archive services can usually also be user-requested as required.

Some products provide the capability to compress the transferred backup and archive data to improve network performance.

Products providing backup and archive services are discussed in Chapter 5, “Backup and Restore” on page 135 and Chapter 6, “Archive and Retrieve” on page 165.

Ease of Use: One of the main purposes of higher-level tools is to provide an easy-to-use mechanism for the management of storage subsystems. As environments become more distributed, the task of management becomes ever more complex, and methods to integrate and automate the required tasks become more important.

In addition, some tools provide graphical user interfaces. These interfaces greatly simplify the tasks from both the user and administrative points of view.

Chapter 2. Storage Management Products

This chapter describes each of the products covered in this document at a relatively high level. The purpose of these descriptions is to allow the reader to gain a background-level understanding of the complete capabilities of each product.

2.1 ADSTAR Distributed Storage Manager/6000

ADSM provides client and server components that allow for network-based backup/restore, archive/retrieve and space management services. Server and client components are supported on a wide range of platforms (see Appendix A, “Supported Platforms” on page 223 for more information). In addition, various remote-mounted file systems are supported (see Chapter 8, “Remote File Systems” on page 191 for further details).

2.1.1 User Client

The user client component is provided from either a command line interface or a GUI (shown in Figure 1). It executes at the client system and allows the user to initiate backup/restore and archive/retrieve operations locally. The command line interface can be used in environments where the GUI is not available.

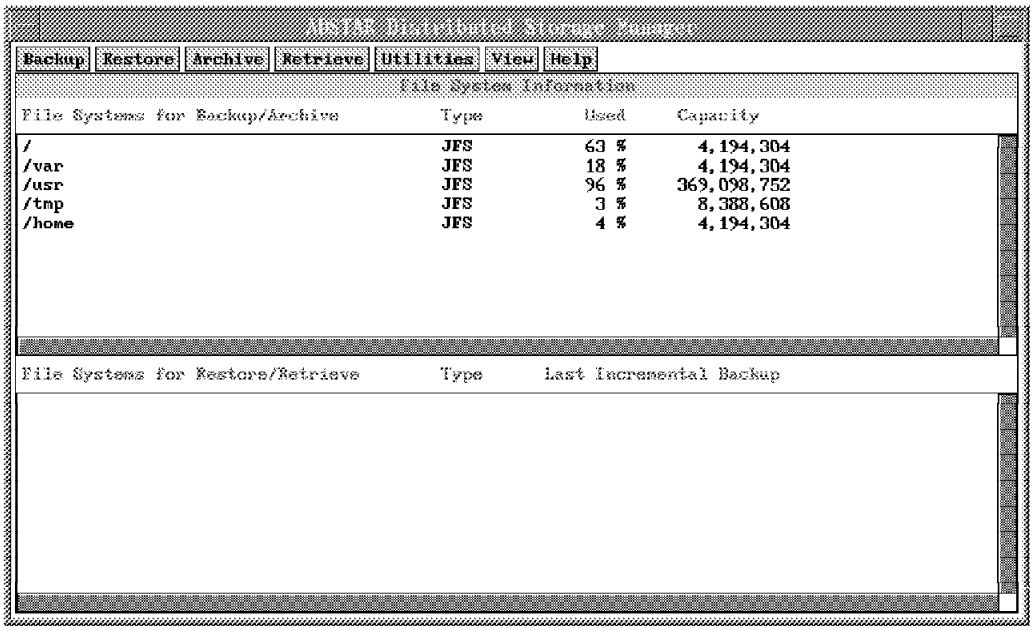

Figure 1. User Client Backup/Restore Graphical User Interface

The following functions can be initiated from the GUI:

- Backups

  These can be performed incrementally, by directory tree or on an individual file-specification basis.

- Restores

These can be requested by directory tree, by file specification or by subdirectory path.

- Archives

  These can be performed by file specification or by directory tree.

- Retrieves

  These can be performed by file specification or by directory tree.

- Utility functions

  These include allowing the user to change passwords, set authorizations and display information about current options and policies in effect.

- View options

  These allow the user to alter the way in which information is presented by the GUI.

Compression can be performed at the client prior to transmission of data across the network to the server in order to maximize network performance and minimize ADSM server storage requirements.

With the announcement of ADSM/6000 V2.1, space management capability has been included with a product called ADSM Hierarchical Storage Management. This gives the client the ability to create file systems that will be managed by the ADSM server. The management capability includes the automatic migration of ordinary files (not directories or special files) from this file system to the server filespace based upon age or size criteria. A GUI is provided to manage the functions available at the client (see Figure 2 on page 21), as well as making it possible to manually migrate files to and from the server.

The following functions can be initiated from the GUI:

- Selective Migration

  This allows files to be manually migrated to the server.

- Selective Recall

  This allows files to be manually returned from the server to reside physically at the client.

- Space Manager

  This provides options to control the behavior of the space management service. Space management function can be added or removed from a file system, or the criteria by which files within a managed file system are migrated can be adjusted.

Figure 2. User Client Space Management Graphical User Interface

2.1.2 Server

The ADSM server component provides a command line interface for managing the product though there is also an administrative client component that can be used to manage the server from any supported machine in the network (see 2.1.3, “Administrative Client” on page 23). An administrative GUI is currently available on AIX, OS/2, HP, and SUN platforms; a command line interface is also available on these and other supported platforms.

The server manages local storage devices to provide space for client backup, archive and space management. Local storage is defined by class and can be disk, optical or tape. Devices of these classes are then defined as storage pools into which client backups and archives take place. Thus, clients can backup or archive directly to a tape pool, optical storage pool or faster disk pool. Pools can be linked into a hierarchy with an arbitrary number of levels of any mix of tape, optical or disk products. Furthermore as many hierarchies as are required can be defined. Data is migrated from the upper levels of the hierarchy downwards as higher levels become full. Typically, faster disk devices occupy the higher levels of a hierarchy with correspondingly slower devices at intermediate levels, culminating in tape devices or libraries. This mechanism ensures the most efficient use of available storage space.

The server supports the concept of storage policies. Policies define how data is to be managed for clients by ADSM and can be defined for total business requirements, department requirements and individual workstation requirements, for example. A large degree of flexibility is possible. Clients with similar requirements are grouped by domain. Within each domain, various policy sets can be defined

with only one being active at any time. The policy set contains management classes that define the way in which the ADSM/6000 server will manage backup, archive and space management services for the clients. These definitions are collected into copy groups, one for each type of service. A copy group for backup, for example, will define among other things, how many copies should be maintained for each backup version and how long copies should be kept.

A central scheduling service is available that allows backups, archives, restores, retrieves, macros, client operating system commands and ADSM server commands to be defined to occur automatically for clients and servers. Both prompted and polled initiation is supported, meaning the services can be initiated from the server or the client.

Multiple file system types are supported, including the Journaled File System, Network File System and Andrew File System. Multiple clients can access the server simultaneously, and multiple devices can be used at the same time.

The ADSM/6000 server also maintains a database in order to keep track of the location of all the data being managed. This database can be mirrored for additional availability. A transaction log is also kept to assist in ensuring the integrity of the data. The meta-data from the database can be exported between ADSM/6000 servers on different platforms, allowing workload balancing and the movement of definitions between servers.

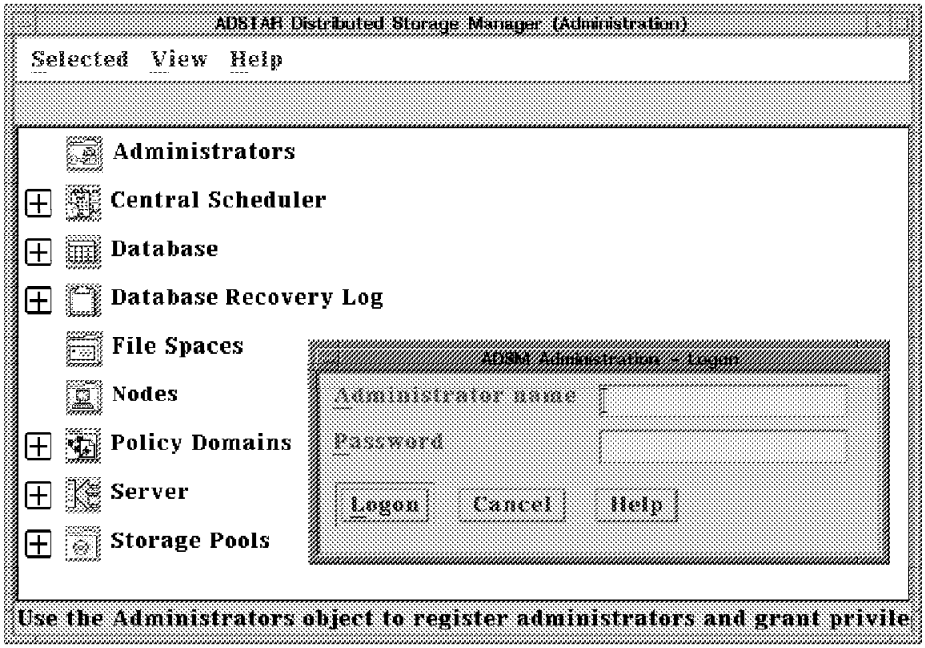

Figure 3. Administration Client Graphical User Interface

2.1.3 Administrative Client

The administrative client is also available as both a command line and a graphical user interface. This client can be invoked from any supported client machine in the network, though the GUI is only supported on HP, Sun, OS/2 and AIX platforms, and provides the following functionality:

- Administrators

  This option allows creation, management and deletion of administrator user profiles. The scope of individual responsibilities can also be managed.

- Central Scheduler

  Administrators with this privilege can create, manage and delete schedules for clients and servers in the network.

- Database

  This selection allows the ADSM/6000 server database activity to be monitored.

- Database Recovery Log

  This selection allows the ADSM/6000 server recovery log to be monitored and the mode of operation to be controlled.

- Filespaces

  This option allows information relating to the filespaces created for client node data to be displayed. Client filespaces can also be deleted or renamed.

- Nodes

  This option allows information relating to the client nodes themselves to be displayed and manipulated. Client capabilities and allowed operations can be managed as well as new clients defined and existing clients deleted.

- Policy Domains

  This option allows the creation, management and deletion of information relating to policies. Policy domains, policy sets, management classes, and copy groups can all be manipulated.

- Server

  This options provides for control of the server to those administrators with system privileges. The server can be halted, enabled and disabled. Audit and license functions can be controlled and parameters governing server operation manipulated. In addition, the activity log can be viewed, server processes in operation can be monitored, server/client sessions in progress can be observed and controlled, and information relating to the usage of sequential media, such as tape, browsed.

- Storage Pools

  This option allows the creation, manipulation and deletion of storage pools. Hierarchies can be defined and status and utilization information can be monitored.

2.1.4 Application Programming Interface (API)

An API that gives applications access to ADSM/6000 services has also been made available. Applications such as databases or other space management tools can use the API to directly store and retrieve information from the server. An example of an application that uses this function is FSF/6000 which can make use of the ADSM/6000 server for migration of client files.

2.2 File Storage Facility/6000

FSF is supported under AIX and Sun Solaris V2.2, both products providing the same functionality. FSF offers client machines space management capability. This is provided in the form of a managed cache on a disk at the client machine. Files stored in this cache can be automatically migrated to server disk space based upon certain usage criteria, thereby giving the appearance of much greater disk space at the client than actually exists.

Access to server disk is either via NFS or using the ADSM API. Thus, most platforms supporting NFS can act as a disk server for FSF as well as any ADSM server platform. In addition, FSF will interoperate with UniTree/6000.

The client cache appears as a normal file system to the user. The cache space is managed, based on the following criteria:

- File age

  Files that have been in the cache for a defined length of time unused will be copied to the server disk and only a pointer left to them in the cache. It will still appear to the user that the file resides locally, but the disk space it would have used is free for other files. Should the user require the file, to view, execute, or edit, it is transparently copied back to the cache for the operation. Before the transparent retrieval, these files are cached with no local copy (remote).

- File modification

  Files that have been modified will be copied up to the server though the file will also remain locally until age or cache utilization causes the local copy to be removed. These files are said to be cached with a local copy.

- Cache utilization

  If the cache should reach a defined space utilization level, then any eligible cached local files, or files unused for long enough, will be cached remotely, thus freeing space.

- Manual pruning

  Users can elect to have files cached remotely on a manual basis. If the file is accessed subsequently, it will be transparently retrieved as normal.

- Manual pinning

  Users may have files that are not used often enough to avoid being pruned, but when accessed, require good initial response time. In these cases, the files can be pinned. Pinned files are never pruned by FSF; a copy always remains locally.

Additional advantages gained from the use of FSF include the consequent ability to manage files centrally, particularly when using ADSM as a server. The centrally

located files can be backed up easily, and in the case of ADSM, use made of hierarchical storage management to further increase efficiency and availability.

Administration and control of FSF is via SMIT only.

2.3 UniTree/6000

UniTree provides hierarchical storage management facilities for client NFS and FTP users in a distributed environment. A two-level storage hierarchy is defined on the UniTree/6000 server consisting of an upper level of disk cache and a lower level of tape or optical storage. This hierarchy is seen as a normal file system by clients, the actual location of information in the hierarchy being transparent. Based upon criteria such as age and disk cache utilization, files are automatically migrated by UniTree to the lower level.

The UniTree/6000 file system can be accessed by clients either using NFS or via FTP:

1. NFS

   The UniTree file system is mounted at the client machine using standard NFS services. User access to the mounted file system is as for any file system. All files stored in the UniTree file system will actually be stored in the hierarchy at the server.

2. FTP

   Clients use standard FTP commands to access the UniTree file system. Files can be sent and retrieved from the UniTree file system in this way. A number of additional commands are also provided within the enhanced FTP server at the UniTree server that allow clients to perform file status and management operations.

In both cases, no additional code is required at client machines which use standard AIX FTP and NFS services.

The UniTree server can manage a variety of tape and optical storage devices and libraries as the second level of storage. Up to 16 copies of each piece of user information can be maintained, each on a separate storage volume within the lower level of the hierarchy. All information in the hierarchy is tracked by UniTree on disk. Mirrored copies of this information can be automatically kept and can be backed up by UniTree to the second level of storage. The first level disk cache is not backed up, but as long as the information is periodically migrated to the second level, restoration is possible in the event of disk failure.

The hierarchical storage system of UniTree primarily provides for management of disk space on behalf of clients as well as a mechanism for archiving information. Information is tracked and managed on a file basis; directory structure is not preserved by UniTree in the event of disk cache failure.

Customization and administration of UniTree are via SMIT at the server machine. As has been mentioned, the user interfaces are via the standard NFS and FTP services at the client machines.

2.4 Legato NetWorker

Legato NetWorker provides backup and restore facilities to clients in a distributed environment. Highly flexible manual or automatic services are available to a diverse range of client platforms. The Legato NetWorker server supports a wide range of backup devices, including tape and optical libraries.

The Legato NetWorker architecture is client/server based, with the following components:

- Server

  The Legato NetWorker server allows backups to be centrally scheduled or performed on client demand. Multiple requests can be processed concurrently to multiple backup devices, if required. The server informs users and administrators of backup events and maintains a catalog of history information for each client.

  The server supports a range of backup devices, including optical and tape libraries for use as backup devices, and media from any device on a particular system using Legato NetWorker is interchangeable with the same device on any other platform supported which is also using Legato NetWorker.

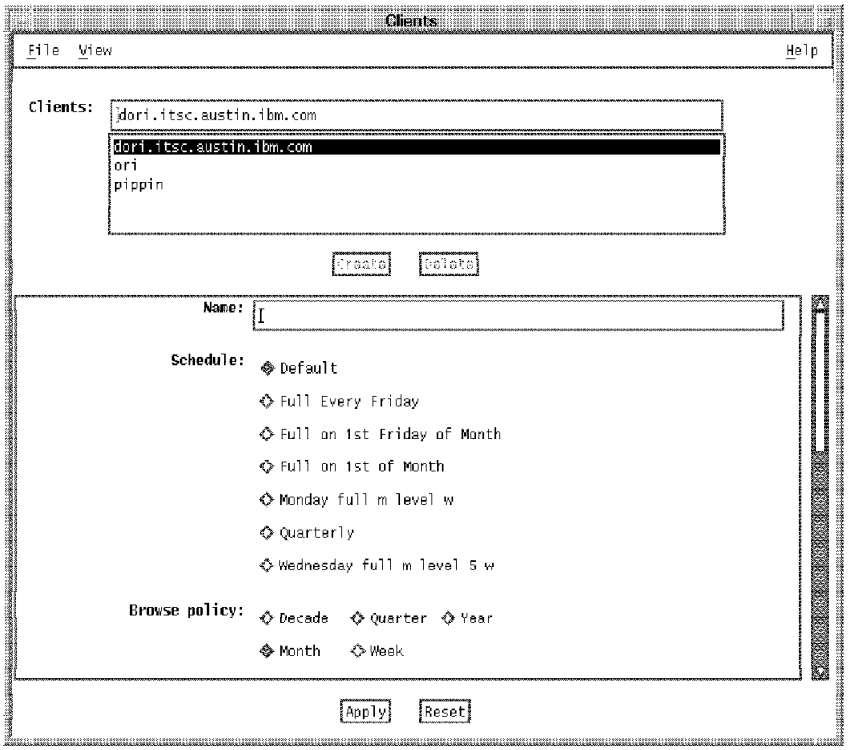

Figure 4. Client Backup/Restore Graphical User Interface

- Client

  Using the Legato NetWorker client code, users can initiate backups of files, directories or file systems to the server. The client history maintained by the server is available for browsing and selection of specific file, directory or file system versions for restore.

Clients interact with Legato NetWorker via a Motif-based graphical user interface (see Figure 4).

- Administrator

  The Legato NetWorker server can be configured, administered and monitored from the server or any client in the network. From the Motif-based graphical user interface (see Figure 5), schedules can be defined, changed and deleted, new clients can be added and backup strategies defined.

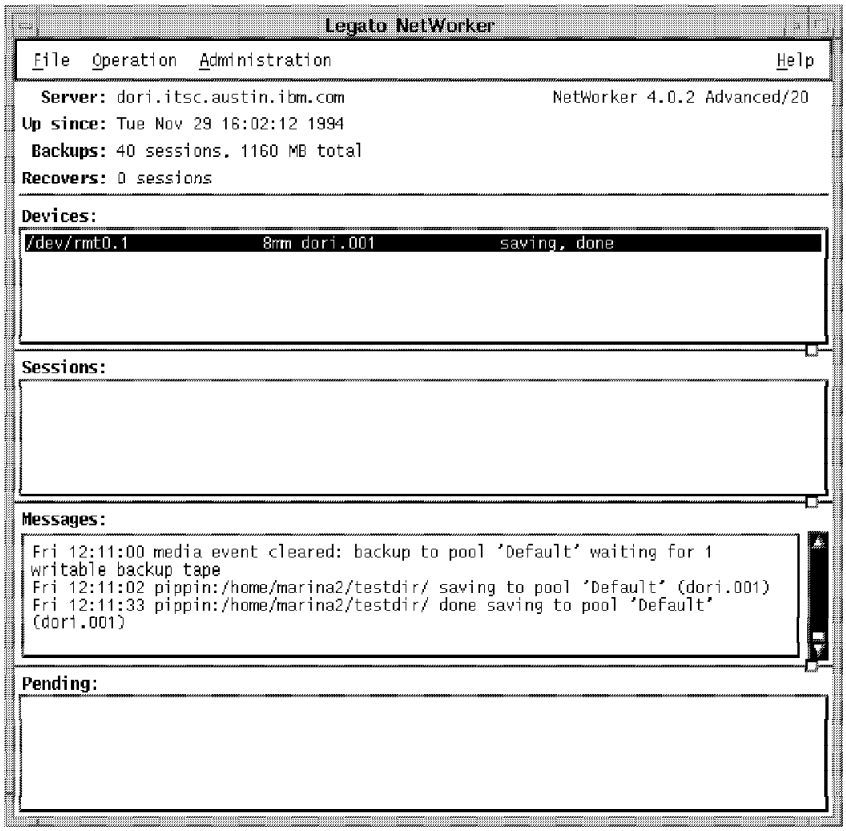

Figure 5. Administrator Graphical User Interface

Backup and restore operations can occur while clients are using their data without risk of corruption. Client and server functions are interoperable across dissimilar hardware platforms.

Chapter 3. Installation and Customization

This chapter describes the main tasks required to install and customize the storage management products. This includes the basic steps needed in order to get the products up and running as well as a complete list of the customization options available for each product. More detailed information on specific areas is available from references to other sections of this document or to different publications.

3.1 ADSTAR Distributed Storage Manager/6000

The basic steps that need to be performed in order to install the ADSM product can be summarized as follows:

- Server setup
  - SMIT installation
  - Customization of the options file
  - Definition of the volumes
  - Loading of the kernel extension
  - Setup of database and recovery log
  - Server activation
  - Further optional customization
- Client setup
  - SMIT installation
  - Customization of the options files
  - Setup of the Administrative Client
  - Setup of the Backup/Archive Client
  - Further optional customization

3.1.1 Server Setup

For a detailed description of the installation process, refer to the document *ADSM/6000 Installing the Server and Administrative Client*.

3.1.1.1 Installing the ADSM Server

To install the AIX/6000 server, ensure you have at least 25MB of disk space. You must also have root authority to perform the installation.

Run the System Management Interface Tool (`smit`) to install the server.

If a previous ADSM version is installed on your system, you need to reinstall the server. Some versions of SMIT do not support the overwrite option; in this case you have to use the `installp` command from the command line using the `-F` option to force the overwriting of the existing version.

When the installation is complete, exit from SMIT; all of the ADSM server files will have been installed in the directory /usr/lpp/adsmserv/bin.

3.1.1.2 Updating the Server Option File

The server options file contains all of the settings for the initialization of the ADSM server:

- Communications options
- Client/server options
- Database and Recovery Log options
- Miscellaneous (date, time, and so on)

ADSM provides a sample file named `dsmserv.opt.smp`; you must copy this file and customize it according to your requirements:

```
cd /usr/lpp/adsmserv/bin
cp dsmserv.opt.smp dsmserv.opt
```

The first option that must be specified is the communication method. The AIX/6000 server supports the following communication methods:

TCP/IP	Is the default, supported by all of the clients
NETBIOS	Supported by OS/2, DOS, Microsoft Windows clients
IPX/SPX	Supported by OS/2, DOS, Microsoft Windows and NetWare clients
SNALU6.2	Supported by RS/6000, OS/2 and NetWare clients

The other options can be changed at a later time; the server must be stopped and started after any change in order for it to reread the configuration.

3.1.1.3 Defining Volumes for the Server

The server can use either AIX/6000 logical volumes or files for its recovery log, database and storage pool volumes.

SMIT can be used to set up logical volumes for ADSM. Please notice that the sizes of these volumes cannot be changed once they have been defined to the server because ADSM uses the initial size allocation to calculate data placement for later retrieval.

The space needed by the server must be preallocated and this can be achieved in two ways. A GUI interface provided by the EZADSM utilities can be used or the `dsmfmt` command can be used as follows:

```
dsmfmt -byte -type filename size
```

where:

byte	Specifies the size in bytes: `-k` for KB, `-m` for MB (default)
type	Specifies the type of the file to be formatted: `-db` for database, `-log` for recovery log and `-data` for storage pools
filename	The name of the file to be formatted
size	The number of bytes to allocate

An example of formatting the files `dsmlog1` and `dsmlog2` to use as recovery logs follows:

```
dsmfmt -m -log dsmlog1 9 dsmlog2 9
```

It is not necessary to issue the `dsmfmt` command for logical volumes.

3.1.1.4 Loading the Kernel Extension

Before starting the server, the kernel extension must be loaded. The current directory must be the one in which the ADSM files are located. The command to issue is:

```
loadpkx -f pkmonx
```

After any system boot, this command must be reissued: a good solution would be to insert it into the /etc/rc.tcpip file:

```
loadpkx -f /usr/lpp/adsmserv/bin/pkmonx
```

3.1.1.5 Setting Up the Database and Recovery Log

During the SMIT installation, an initial database and a recovery log are automatically defined; these may be used or discarded as required.

The first time the server is run, it must be started in installation mode in order to internally format the recovery log and the database.

The `dsminst` shell script can be run, which defaults to the standard names, or the `dsmserv install` command can be directly issued with the correct names of the volumes to be used:

```
dsmserv install 2 dsmlog1 dsmlog2 1 /dev/rdsmvol1
```

3.1.1.6 Activating the Server

On an AIX/6000 system, start the server with the following command:

```
/usr/lpp/adsmserv/bin/dsmserv
```

When the server starts, it initiates a server console session that is used to operate and administer the server until administrative clients are registered.

The server may be run in the background, using the command:

```
nohup dsmserv quiet &
```

Notice that in this case there is no console to control the server, and an Administrative Client will need to be run.

At startup, the server displays the following information:

```
ANR0900I Processing options file dsmserv.opt.

ADSTAR Distributed Storage Manager for AIX-RS/6000
Version 1, Release 3, Level 0.0/0.6

Licensed Materials - Property of IBM

5765-203 (C) Copyright IBM Corporation 1990, 1994. All rights reserved.
U.S. Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corporation.

ANR7800I DSMSERV generated at 13:47:10 on Sep 29 1994.
ANR7801I Subsystem (master) PID is 8939.
ANR0990I ADSM server restart-recovery in progress.
ANR0200I Recovery log assigned capacity is 12 megabytes.
ANR0201I Database assigned capacity is 4 megabytes.
ANR0306I Recovery log volume mount in progress.
ANR0353I Recovery log analysis pass in progress.
ANR0354I Recovery log redo pass in progress.
ANR0355I Recovery log undo pass in progress.
ANR0352I Transaction recovery complete.
ANR1305I Disk volume /dev/rdsmmed varied online.
ANR1305I Disk volume /dev/rdsmslow varied online.
ANR1305I Disk volume /dev/rdsmfast varied online.
ANR1305I Disk volume /dev/rdsmopt varied online.
ANR2100I Activity log process has started.
ANR0811I Inventory client file expiration started as process 1.
ANR2803I License manager started.
ANR8200I TCP/IP driver ready for connection with clients on port 1500.
ANR0812I Inventory file expiration process 1 completed: deleted 0 backup files
and 0 archive files.
ANR0993I ADSM server initialization complete.
ANR2560I Schedule manager started.
adsm>
ANR9620I An EVALUATION LICENSE for 1 client connection(s) will expire on
03/16/2029.
ANR2813I Server is licensed for a capacity of 25 gigabytes and 1 clients.
```

The ADSM license is based on the number of registered clients and the total amount of storage allowed; when ADSM is installed, only 25GB of storage and one client can be registered for use.

Refer to Chapter 4, “Product Administration” on page 69 for more details on how to request and implement the license.

The last actions to perform before installing the clients are:

1. Name the server
2. Register an administrator

At the ADSM command line prompt, enter:

```
servername=your_server_name

register admin admin_id <password>
grant authority admin_id class=system
```

3.1.1.7 Customizing Additional Server Options

This step is optional. It is reported here only for completeness and to show all of the capabilities and flexibility of the ADSM server.

You can list the complete options set for the server by issuing the command:

```
query OPTion
```

The report is as follows:

```
Server Option     Option Setting       Server Option     Option Setting
---------------- ------------------- ---------------- ---------------
CommTimeOut       60                  IdleTimeOut       15
BufPoolSize       512                 LogPoolSize       128
DateFormat        1 (mm/dd/yyyy)      TimeFormat        1 (hh:mm:ss)
NumberFormat      1 (1,000,000)       MessageFormat     1
Language          AMENG               MaxSessions       25
ExpInterval       1                   MirrorRead DB     Normal
MirrorRead LOG    Normal              MirrorWrite DB    Sequential
MirrorWrite LOG   Parallel
TcpPort           1500                IPXSocket         8522
NetbiosBufferSize 16384               NetbioSessions    25
LuName            DSMSERV1            TPNProfilename
IPXBufferSize     4096                CommMethod        TCPIP
Message Interval  1

adsm>
```

Figure 6. ADSM Server Options

The following list provides explanations for the options. The parts of the options in capitals are the minimum amount of the option that actually needs to be entered in the options file.

- Client/server options:

  COMMTimeout The number of seconds server waits for a client message during a database update operation

  EXPINterval The interval in hours between automatic inventory expiration runs. If the value is set to 0, files are removed only with the `EXPIRE INVENTORY` command

  IDLETimeout The number of minutes server waits for a client to initiate a communication

  MAXSessions The maximum number of simultaneous client sessions

  MSGINTerval The interval in minutes in which the operator is prompted by the server for a mount operation

- Database and Recovery Log options:

BUFPoolsize	The size in kilobytes of the database buffer pool
DUMPloaddb	Not used during normal server initialization. For more details on database dump/load function, refer to 4.1.4, “System Availability” on page 85
LOGPoolsize	The size in kilobytes of the recovery log buffer pool
MIRRORRead	Specify how the server reads mirrored volumes
MIRRORWrite	Specify how the server writes mirrored volumes. For more details on mirroring, please refer to 4.1.4.3, “Mirroring” on page 89

- Communication options:

COMMMethod	The selected communication method(s)
TCPPort	The port address for TCP/IP communication
LANAdapter	The network adapter number for NetBIOS communication
NETbiosname	The name to be used for communications in the network
NETBIOSBuffersize	The size in kilobytes of the NetBIOS communication buffer
IPXSocket.	The socket number on which the SPX driver is to wait
IPXBuffersize	The size in kilobytes of the IPX/SPX communication buffer
LUName	The logical unit name for SNA LU 6.2 communication
TPNProfilename	The transaction program profile name for SNA LU 6.2 communication

- Miscellaneous options:

DATEformat	The format by which dates are displayed
TIMEformat	The format by which time is displayed
NUMberformat	The format by which numbers are displayed
Language	The national language in use

3.1.2 Client Setup

The ADSM server has been installed, so the installation of the ADSM clients can now be described: namely, the Administrative Client and the Backup/Archive Client. Setup of the space management client is described in 7.1, “ADSTAR Distributed Storage Manager/6000” on page 173.

As already stated in the previous chapters, the ADSM client programs can run on different platforms, but for the purpose of this publication, we will limit our description to AIX/6000 clients.

3.1.2.1 Installing the ADSM Client Program

The root user installs the ADSM client files using the System Management Interface Tool (SMIT). The installation requires at least 3.1MB of disk space, both for Administrative and Backup Client programs.

If reinstallation of the client programs is required, follow the procedure described in 3.1.1.1, “Installing the ADSM Server” on page 29.

At the end of the installation, exit from SMIT; the ADSM Administrative Client files are installed in the directory /usr/lpp/adsm/bin, while the Backup/Archive Client files are in /usr/lpp/adsm/sm/bin. The installation procedure creates symbolic links to the executable files.

3.1.2.2 Updating the ADSM Client Options Files

ADSM provides a sample file, `dsm.sys.smp`, that contains the minimum options required to get started; this file must be copied and customized according to requirements:

```
cd /usr/lpp/adsm/sm/bin
cp dsm.sys.smp dsm.sys
```

The next figure shows the contents of the client sample system options file:

```
*SErvername  server_a
   *COMMethod          TCPip
   *TCPPort            1500
   *TCPServeraddress   node.domain.company.COM
```

Figure 7. ADSM Client Sample System Options File

In this file, there can be multiple options sets, one for each ADSM server to which communication is required from the workstation.

By default, the client contacts the first server identified in the system option file. A different default server can be specified by editing the client user option file:

```
cd /usr/lpp/adsm/sm/bin
cp dsm.opt.smp dsm.opt
```

The next figure shows the contents of the client sample user options file:

```
*SErvername    A server name defined in the dsm.sys file
```

Figure 8. ADSM Client Sample User Options File

3.1.2.3 ADSM Administrative Client

On AIX/6000, there are two different administrative interfaces available:

1. The administrative command line interface, which is started by entering the command `/usr/lpp/adsm/bin/dsmadmc`
2. The administrative graphical user interface, which is started by entering the command `/usr/lpp/adsm/bin/dsmadm`

Refer to Chapter 4, “Product Administration” on page 69 for a detailed description of ADSM administrator tasks.

```
ADSTAR Distributed Storage Manager
Command Line Administrative Interface - Version 1, Release 3, Level 0.0
(C) Copyright IBM Corporation, 1990, 1994, All Rights Reserved

Enter your user id: yyyyy
Enter your password: xxxxxx
ANS5100I Session established with server ADSM: AIX-RS/6000

adsm>
```

Figure 9. Administrative Client Command Line Interface Display

Refer to Figure 3 on page 22 for the Administrative Client GUI display.

3.1.2.4 ADSM Backup/Archive Client

If AIXwindows is running on the workstation, the ADSM graphical user interface (GUI) can be started and used to perform most of the Backup/Archive Client tasks.

To start the interface, one of the following two commands can be run:

```
dsm
dsm &
```

depending whether the client is to be run in the foreground or in the background. The client GUI interface is shown in Figure 1 on page 19.

ADSM also provides a command line interface that can be used as an alternative to the GUI. A client command session can be started in either batch or interactive mode:

```
$ /usr/lpp/adsm/sm/bin/dsmc

ADSTAR Distributed Storage Manager
Command Line Backup Client Interface - Version 1, Release 3, Level 0.0
(C) Copyright IBM Corporation, 1990, 1994, All Rights Reserved

dsmc>
```

Figure 10. ADSM Backup Client Interactive Command Session

Issue `quit` to exit from an interactive session.

The batch session is useful when only a few simple commands need to be issued, as in the following example:

```
# dsmc set password

ADSTAR Distributed Storage Manager
Command Line Backup Client Interface - Version 1, Release 3, Level 0.0
(C) Copyright IBM Corporation, 1990, 1994, All Rights Reserved

Please enter password for node "ORI":
Please enter a new password:
Reenter new password for verification:
Password updated

#
```

Figure 11. ADSM Backup Client Batch Command Session

3.1.2.5 Customizing Additional Client Options

The following options can be set for the ADSM clients:

- Administrative Client System Options:

DATEformat	The format in which dates are displayed
LANGuage	The national language to use
NUMberformat	The format in which numbers are displayed
TIMEformat	The format in which time is displayed
CPIcbuffersize	The size in kilobytes of the buffer used for SNA LU 6.2 communications
CPICMOdename	The mode name for SNA LU 6.2 communications with the server
PARtnerluname	The logical LUName used by the SNA LU 6.2 transaction program
SYMbolicdestination	The name used for an SNA LU 6.2 connection to the server
TPname	The symbolic name for a transaction program in an SNA network
IPXBuffersize	The size in kilobytes for the IPX/SPX communications buffer
IPXServeraddress	The hexadecimal address for IPX communication. It can be obtained by running the utility `getipxad` from the server
IPXSOcket	The socket number on which the server SPX communication driver is to wait for requests
NETbiosname	The NetBIOS name for the workstation
NETBIOSServername	The name for the ADSM server using NetBIOS communications
TCPBuffsize	The size in kilobytes of the internal TCP communications buffer
TCPPort	The TCP/IP port address of the server
TCPServeraddress	The Internet address for TC/PIP communications
TCPWindowsize	The size in kilobytes of the TCP/IP sliding window for the client

- Backup/Archive Client System Options:

Since the administrative options are a subset of the Backup/Archive Client options, this section will only cover those options that have not yet been described.

CHANGingretries	The number of retries for backing up or archiving a file in use
COMPression	Whether to compress files before sending them to the server. It is strongly recommended to specify `compression on` in the dsm.sys options file (the default is `no`).
DIRMC	The management class to be used for directories
GRoups	The groups of users allowed to use ADSM services
INCLEXCL	The path and file name of the include/exclude file
MAILPROG	The program and userid to which ADSM should send newly generated passwords
MAXCMDRetries	The maximum number of times the client scheduler attempts to process a failing scheduled program
PASSWORDAccess	Whether users are prompted for password if a password is required
QUERYSchedperiod	The hours the client scheduler waits between attempts to contact the server
RETRYPeriod	The minutes the client scheduler waits between attempts to process a failing command
SCHEDLogname	The name of the file storing schedule log information
SCHEDMode	The schedule mode to be used: POLLING or PROMPTED
USERS	The users on your workstation allowed to request ADSM services
VIRTUALMountpoint	Defines a virtual mount point for a file system

The system options defined at the workstation can be viewed simply by running the `dsm` command and clicking on Utilities. The system display is as follows:

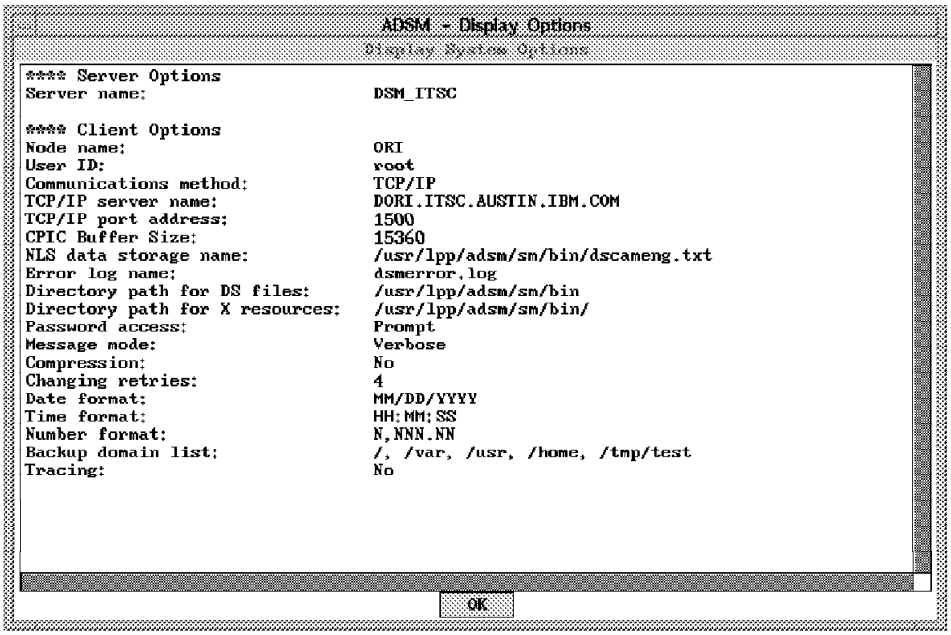

Figure 12. ADSM Backup/Archive Client System Options GUI Display

- Backup/Archive Client User Options:

  The miscellaneous options for the Administrative Client can also be set in the client user options file. A list of the specific client user options follows:

DOMAIN	The file system included in the default domain for incremental backup
NODEName	The name of another node for which ADSM services are required
QUIET	Prevents the appearance of processing information on the screen
REPLace	Whether to automatic overwrite an existing file when restoring or retrieving
SCROLLLines	The number of lines to display at one time on the screen
SCROLLPrompt	Whether to scroll to the end of the list or stop after displaying the number of lines specified in the previous option
SUBDIR	Whether to recursively descend directories for selective backup, restore and query backup
TAPEPrompt	Whether ADSM should prompt for a required tape to be mounted or wait
VERBOSE	Causes processing information to be displayed on the screen; the alternative is QUIET.

3.2 File Storage Facility/6000

The basic steps required to get the FSF product up and running are as follows:

- Planning for product prerequisites
- Product installation
  - Unmount of previous FSF directories
  - SMIT installation
  - Updating the .profile
- Product customization
  - Setup of NFS or ADSM access
  - FSF file system configuration
  - Mounting the FSF file system and starting FSF
  - Setting the access permissions
  - Further customization and installation verification

3.2.1 Product Requirements

Unlike other client storage management products, FSF/6000 does not require special server software to enable its functionality as it may operate with any standard NFS server, with or without a storage management product, such as UniTree. FSF may also cooperate with an ADSM server, providing it with additional automatic client storage management services.

You can install the product on any RS/6000 diskfull workstation running the AIX 3.2 operating system or later releases, and on the IBM N40 notebook. FSF requires a minimum of 10MB of internal disk storage.

The supported LAN adapters are:

- IBM Token-Ring High Performance Network Adapter

- Ethernet High Performance LAN Adapter or integrated Ethernet adapter for applicable machines
- Fiber Distributed Data Interface Adapter and FDDI Dual Ring Upgrade kit

3.2.2 Installation

For a detailed description of the installation process and of the most common installations problems, refer to *AIX FSF/6000 Installation, Planning and User's Guide*.

3.2.2.1 Unmount of Previous FSF Directories

This step only needs to be performed if updating from a previous release of FSF. Before installing the product, the existing FSF file system should be unmounted. To do this, as root, do the following:

1. Be sure that nobody is currently using the cache directory
2. Kill any running process started from the cache
3. Enter `smit fsf`
4. Select the **Unmount an FSF File System** option
5. Answer **yes** to the related question

3.2.2.2 SMIT Installation

Before loading FSF files, be sure that there is at least 5MB free in the /usr directory. Root authority is required to perform the installation.

Run the `smit install` command to install the FSF product.

When the installation is complete, exit from SMIT; all the FSF files are installed in the directory /usr/lpp/fsf.

3.2.2.3 Updating the User Profile

The FSF commands reside in the directories: /usr/lpp/fsf/utilities and /usr/lpp/fsf/bin. The .profile of any user planning to run FSF commands needs to be edited and the PATH variable updated to include the two FSF pathnames.

3.2.3 Customization

Once the FSF files have been successfully installed into the system, the logical volumes that FSF will use to manage files need to be created. These volumes are:

1. The server store, an existing NFS file system or an ADSM filespace located on a remote server
2. The local cache, a client JFS file system
3. The FSF manager logical volume
4. The FSF log logical volume

3.2.3.1 Setup of NFS or ADSM Server

Before starting the configuration of the client, the appropriate file system must be prepared on the remote server. This task can be performed as follows:

NFS server

1. Logon as root on the server machine
2. Define the file system that will become the server storage pool for the FSF client
3. Change the permission bits in order to allow read/write access to anybody
4. Run the SMIT NFS program in order to export the file system, setting to `no` NFS root access. Remember, do not create files in the FSF client file system cache while logged on as root

ADSM server

1. Ask the ADSM administrator to register the client node with a password on the ADSM server
2. Set up the client option files for ADSM at the client machine. FSF installation creates two sample files in the directory /usr/lpp/adsmapi. The file dsm.sys.smpapi must be copied to dsm.sys and the file dsm.opt.smpapi to dsm.opt in the same directory. Customize them according to the installation requirements. For more information about the customization of these files, refer to 3.1.2, “Client Setup” on page 34
3. Edit the file /usr/lpp/fsf/etc/adsm.passwd, adding one line with the following information:
   - The name of the server as specified in file dsm.sys
   - The name of the client node
   - The ADSM password of the client node
4. Register the ADSM filespace from the client, using the command `dcsd_dsm_fsreg` with the following arguments:
   - The name of the client node
   - The ADSM password of the client node
   - The session type (or communication method) specified in the file dsm.sys (which, for AIX, is usually `tcpip`)
   - The name of the filespace that is to be registered (the FSF client cache directory)
   - The file type of the filespace (JFS)

   The command is in the directory /usr/lpp/fsf/bin. A sample invocation of this command follows:

   ```
   dcsd_dsm_fsreg CLIENT_NAME CLIENT_PWD SESS_TYPE FILESP_NAME FILESP_TYPE
   ```

5. Configure the backup copygroup in the policy domain assigned to the FSF client machine.

   FSF backs up, rather than archives its files, to the ADSM server. If no configuration is performed at the ADSM server, then the FSF client will inherit the default configuration which is the STANDARD policy domain. In this domain, it will have the STANDARD policy set, the STANDARD management class and the Backup copygroup. From a configuration point of view, the copygroup is the most important as it defines the number of copies maintained for backed up files as well as the retention period. These parameters are on page two of copy control in the Backup copygroup properties, if using the

administrator GUI. If using the console or administrator command line interface, use the `UPDATE COPYGROUP` command. If unchanged, ADSM will maintain two copies of every file sent by FSF, but more importantly, if a file is deleted from the FSF client cache that has already been sent to ADSM, ADSM will still maintain one copy for 60 days. This can have major ramifications in terms of storage space at the ADSM server.

A good recommendation is to retain one backup version if client data exists, and zero backup versions if client data is deleted. The length of time to retain the only backup version should be unlimited.

See 5.2, “ADSTAR Distributed Storage Manager/6000” on page 138 for more information on configuring ADSM for backup.

6. Configure the FSF client node definition at the ADSM server to allow the client to delete backup data.

   On the filespace page of the node properties for the FSF client node (using the administrator GUI), select the option **Can delete backup data**. If using the command line administrator interface, or from the server console, use the `UPDATE NODE` command.

3.2.3.2 FSF File System Configuration

When FSF configuration is started on the client, there may be a requirement to convert existing files into FSF managed files. There are three possible situations:

- Conversion is not required
- Conversion of local files is required
- Conversion of files that reside on the server (NFS-mounted) is required

The following describes the necessary steps to perform each of these tasks.

Configuration Without Existing Files: Run the `smit fsf` program and select the menu **Add an FSF File System**. The display will be as in Figure 13 on page 43. Enter values in the fields as follows:

Mount Point — The pathname of the file system which will become the local cache

Daemon type — Select the NFS or ADSM daemon using PF4

HOST name — NFS users should specify the host name of the server machine. ADSM users must specify the name used in the file dsm.sys

Pathname — NFS users should specify the pathname of the NFS file system on the remote host to be exported as FSF server store. ADSM users must specify the filespace name registered for the client and the high-level name for the file system: `/filespace:/high_level_name`. The high-level name can be the same as the filespace name

FSF volume group name — Any existing volume group name can be specified for the manager, log and cache

FSF log volume — Names may be assigned to the manager and log volumes if required, or the AIX defaults will be used

FSF log size — The size required for about 13,000 file creation operations is 4MB

Create cache — FSF will create the local cache file system, if required

Incorporate files — Select **NONE**

Mount AUTOMATICALLY
: If `yes` is selected, the file system is mounted and the FSF daemon started automatically

Configuration Including Files Residing on the Client: Run the `smit fsf` program and select the menu **Add an FSF File System**. The display will be as in Figure 13. The values that must set, in order to include files already in the FSF cache, are as follows:

Mount Point
: The pathname of the file system that is to be FSF managed

Create cache
: Select `no`

Incorporate files
: Select `CACHE`

For the other fields, look at the explanations in the previous case.

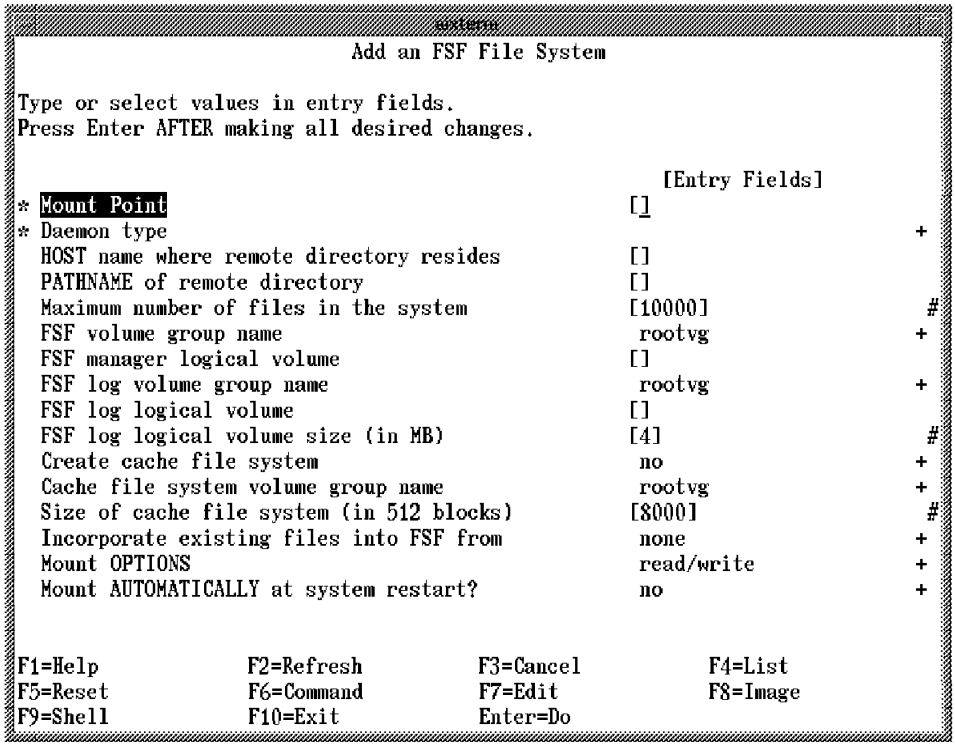

Figure 13. Add an FSF File System Display

Configuration Including Files Residing on the Server: In this case, some additional steps may need to be performed before calling the SMIT FSF interface. If a new JFS is to be created as part of the FSF configuration rather than using an existing file system, the first two steps may be skipped.

1. Remove the NFS-mounted home directory. Be sure to unmount the file system and also to remove the entry from /etc/filesystems
2. Add a new JFS, with the same mount point as for NFS, and set the `Mount AUTOMATICALLY at system restart` option to `no`
3. Now FSF configuration can be started by running `smit FSF` and selecting **Add an FSF File System**. The display will be as in Figure 13.

The values that must be set, in order to include the files in the FSF cache, are as follows:

Create cache Select `no` only if the first two steps were performed; otherwise select `yes` to make FSF create the file system

Incorporate files Select `STORE`

3.2.3.3 Mounting the FSF File System and Starting FSF

If everything went well in the previous step, and `no` was selected for the mount automatically option, then the file system can now be mounted selecting the **Mount FSF File System** option from the SMIT FSF interface.

Mount point to mount Enter the mount point

Space retry? This option manages a disk full condition in the local cache. If set to `yes`, FSF will migrate files to the server store and delete them from the local cache to free space for the current write operation. If set to `no`, FSF will not complete the write operation and return with an error message

Start daemon? Set to `yes` to start the FSF daemon on the client workstation

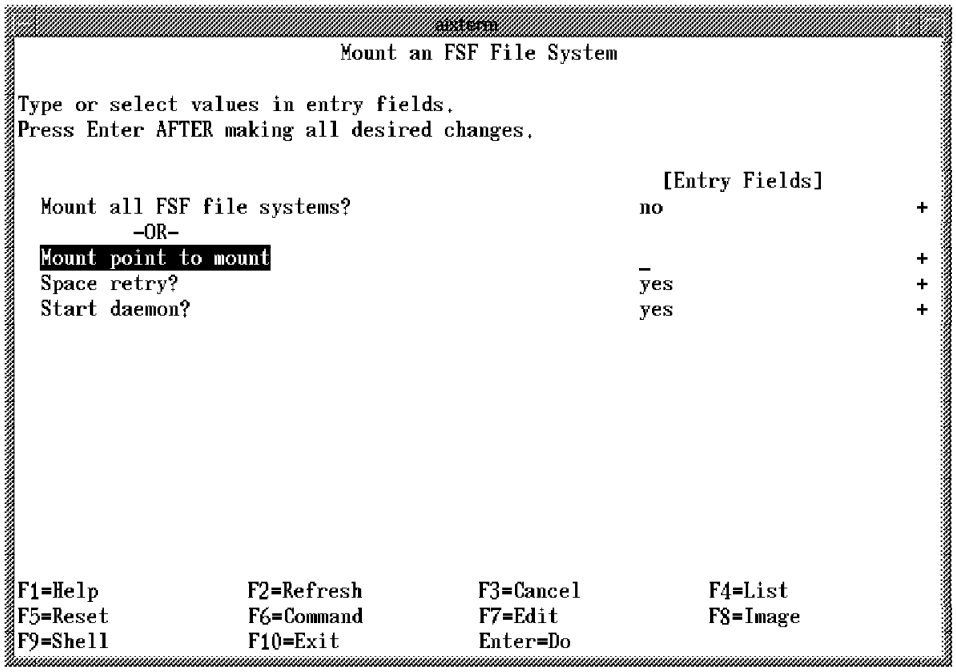

Figure 14. Mount an FSF File System Display

When mounting a new empty cache, the following warning message can be experienced:

```
mount: 0506-341 warning: permission on /your_mount_point are incompatible
        with the permissions in the root directory of /dev/lv0x.
FSF file system "/your_mount_point" is mounted.
```

Exit from SMIT and check the directory of the local FSF file system; the permission bits will be set to `drwxr-s---` instead of `drwxr-sr-x` as they should be. In this case, the permission bits will need to be manually changed before starting to allocate files in the cache; the missing permissions would cause FSF not to correctly manage files.

3.2.3.4 Further Optional Customization

It may be necessary to set the file access permission for the client cache directory in the new file system. Be sure that the server store directory has the same permissions.

From the SMIT FSF interface, some additional parameters can be customized using the **Perform FSF Maintenance** option. For a complete description of these parameters, refer to the administration tasks description in 4.2, “File Storage Facility/6000” on page 90.

3.3 UniTree/6000

There are several tasks that should be performed in order to get UniTree/6000 up and running on the workstation. A summary list follows:

- Planning
  - Hardware and software prerequisites
  - Configuration guidelines
- Product Installation
- Customization tasks
  - Running minimum configuration
  - Enabling NFS
  - Adding disk server volumes
  - Configuring sequential devices
  - User definition
- Running the verification test

3.3.1 Planning

UniTree can be installed on RISC System/6000 models 7012, 7013 and 7015 running AIX Version 3.2 or later. It is designed to work also on an HACMP workstation cluster.

UniTree supports the following LAN adapters:

- IBM Token-Ring High Performance Network Adapter
- Ethernet High Performance LAN Adapter or integrated Ethernet adapter for applicable machines
- Fiber Distributed Data Interface Adapter and FDDI Dual Ring Upgrade kit

The UniTree system requires at least 32MB internal memory, but a minimum of 64MB is recommended in order to have an acceptable response time. It also requires 9MB of disk space for a successful installation and a minimum of 24MB of disk space for storing its system data structures. Additional space is required for the disk server logical volumes (the disk cache), depending on the usage of the system.

3.3.1.1 Device Drivers

UniTree supports multiple disk, tape and optical devices, both from IBM and third parties. It is essential to have the corresponding device driver software installed on the system. In Table 1 on page 46, Table 2 on page 46 and Table 3 on page 46, a detailed list of the available device drivers can be found.

Table 1. Device Drivers for UniTree-Supported Disk Devices	
Device	**Device Driver Requirements**
IBM internal and external disk drives and subsystems	AIX base operating system supports the device; no driver required.
Maximum Strategy RAID storage server	May be obtained from Maximum Strategy Incorporated.
Storage Concepts Concept 550 RAID storage system	AIX base operating system supports the device; no driver required.
Storage Concepts Concept 51-S RAID storage subsystem	AIX base operating system supports the device; no driver required.

Table 2. Device Drivers for UniTree-Supported Tape Devices	
Comtec ATL-8 Model 54	Comtec owned software required. May be obtained from ACSC or Comtec.
Exabyte EXB-10i, EXB-10e and EXB-120 CHS	May be obtained from ACSC or call IBM support.
IBM 7208 External 8mm Tape Driver	AIX base operating system supports the device; no driver required.
IGM-ATL 8mm	AIX base operating system supports the device; no driver required.
LAGO Systems LS/380L DataWheel	AIX base operating system supports the device; no driver required.
StorageTek 4781/4780 cartridge subsystem	May be obtained from IBM.
StorageTek ACS 4400	No additional software required except the 4781/4780 device driver.

Table 3. Device Drivers for UniTree-Supported Optical Devices	
Alphatronix Inspire rewritable optical jukebox	May be obtained from Alphatronix.
DocuStore family of automated libraries (DISC jukebox)	May be obtained from ACSC or call IBM support.
Hewlett-Packard optical disk library system	May be obtained from ACSC or call IBM support.
IBM 3995 optical jukebox	May be obtained from ACSC or call IBM support.

3.3.1.2 Configuration Guidelines

UniTree server processing is managed by four daemons:

unamed Manages the UniTree name server
umigd Manages the UniTree migration server
udiskd Manages the UniTree disk server and the disk mover process
utaped Manages the UniTree tape server, the tape mover, the physical device manager, and the physical volume repository processes

The disk, tape and name servers require their own logical volumes to contain internal data structures. The tape server and name server logical volumes are mirrored for availability purposes. It is recommended you define at least two volume groups located on two different hard disks to protect UniTree internal structures against hardware failures. Also, some of the tape server logical volumes cannot be added or changed after installation, so it is critical to calculate their correct size.

The disk server logical volumes are not mirrored as the UniTree standard archive process itself can provide a good backup solution. Disk server logical volumes may be added at any time after the installation, when data requirements increase. However, since the disk server uses a striping algorithm to allocate space for files, it is recommended for performance purposes, you place the different disk server logical volumes on separate hard disks.

1. Tape server search table

   UniTree requires one logical volume pair dedicated to the tape server for its search table. Its size cannot be changed after installation. A 4MB logical volume is sufficient for approximately 174,000 files. So the required number of partitions can be calculated with the following formula:

   ```
   total_number_of_present_and_future_files / 174,000
   ```

2. Tape server header

   The UniTree tape server requires another logical volume pair for the headers which reference the files residing on the archive volumes. Other logical pairs for tape server headers can be added in the future. A 4MB logical volume can contain about 11,000 headers. Since UniTree will allow multiple copies of each file to be archived on different volumes, the required number of partitions for the tape server header can be calculated using the following formula:

   ```
   number_of_present_files / 11,000 x number_of_file_copies
   ```

3. Name server

   The UniTree name server requires a logical volume pair for its internal structures. Other logical pairs may be added in the future. A 4MB logical volume can contain about 40,000 name server directory entries. The formula to calculate the number of logical partitions is:

   ```
   number_of_present_files / 40,000
   ```

4. Disk server

   UniTree requires at least one logical volume for the disk server to act as the disk cache. Additional logical volumes can be added in the future. The formula to calculate the optimum size and the number of logical volumes required is a bit more complex than in the previous cases. The following steps should be performed:

   a. Calculate the amount of active data on the system

      The number of concurrently active files on the system needs to be known in advance as well as their average size. The minimum for the average file size is considered to be 64KB.

```
number_of_active_files x average_file_size = active_data_size
```

b. Calculate the size of the disk cache required

   An appropriate disk cache ratio should be chosen, or, in other words, that percentage of the active files that are to reside in the disk cache for performance purposes. The cache size must be large enough to contain the largest file. Also there must be enough storage space to contain not only the cache but also the UniTree servers internal data structures.

```
active_data_size x disk_cache_ratio = total_cache_size
```

c. Calculate the optimum size for each disk server logical volume

   Because of internal design, each disk server logical volume cannot contain more than 2,800 files, whatever its size. File headers additionally require about 400KB.

```
( average_file_size x 2800 ) + 400KB = logical_volume_size
```

d. Calculate the optimum number of logical volumes needed for the disk server

```
total_cache_size / logical_volume_size = number_of_volumes
```

3.3.2 Installation

To perform UniTree installation, login as the root user, and check to see that there are at least 9MB free in the /usr directory.

3.3.2.1 Updating a Previous UniTree Release

If there is a previous installation of UniTree on the system, all UniTree NFS clients need to unmount all NFS-mounted UniTree directories and then UniTree must be stopped completely. Those clients failing to unmount may find UniTree NFS not working well for their directories after the update.

A standard SMIT installation can now be performed over your previous version. This update will not change the UniTree configuration, nor will it delete any data stored by the previous installation.

3.3.2.2 Installing UniTree for the First Time

UniTree updates FTP and NFS entries in the following files:

- /etc/inetd.conf
- /etc/services
- /etc/rc.nfs

It may be a good idea to save the original files before starting UniTree installation.

After performing a standard SMIT installation, all the UniTree files will be loaded into the /usr/lpp/UniTree directory.

Edit the root .profile and update the PATH variable, adding the two UniTree pathnames: /usr/lpp/UniTree/bin and /usr/lpp/UniTree/adm/bin.

The product can now be customized. For any additional information on UniTree installation, refer to *UniTree Installation and Planning Guide*.

3.3.3 Customization

Once the UniTree files have been loaded onto the system, the devices required to manage the file system can be defined. First of all, at least two volume groups should be defined, residing on different hard disks for UniTree server logical volumes. The next tasks will be performed by the root user, using the SMIT UniTree dialog:

```
                                    UniTree

Move Cursor to desired item and press Enter.

> Configure UniTree
  Install HACMP/6000 Scripts for UniTree


F1=Help             F2=Refresh            F3=Cancel            F8=Image
F9=Shell            F10=Exit              Enter=Do
```

3.3.3.1 Minimum Configuration Dialog

From the SMIT UniTree interface, select **Configure UniTree**, and from here select **Minimum Configuration**.

```
                               Configure UniTree

Move Cursor to desired item and press Enter.

> Minimum Configuration
  Management of UniTree NFS
  Management of UniTree Users
  Management of UniTree Objects
  Management of UniTree Peripheral Devices


F1=Help             F2=Refresh            F3=Cancel            F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 15. SMIT UniTree Configuration Screen

As shown in Figure 16 on page 50, values should be entered as follows:

Network Information Use the same IP address for disk and tape subsystem.

Name Server Enter the number of partitions previously calculated in 3.3.1.2, “Configuration Guidelines” on page 46, and the names of the two volume groups defined for UniTree.

The default values can be accepted for the other parameters

FTP Daemon UniTree has its own FTP daemon on port 1021. If both FTP daemons are enabled, UniTree users must explicitly specify the port number to open a UniTree connection rather than a normal one.

Standard FTP may also be disabled on the server and port 21 assigned to UniTree FTP, if required.

Normally, the first solution seems to be more flexible.

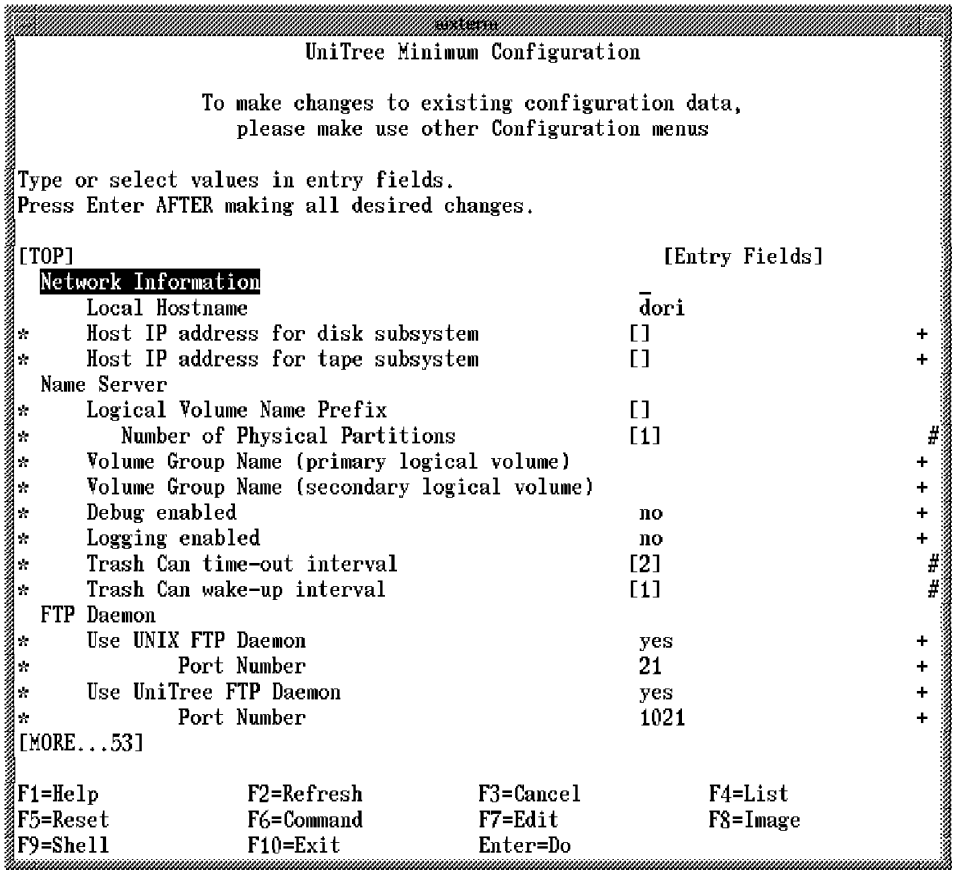

Figure 16. UniTree Minimum Configuration

The UniTree Minimum Configuration dialog continues with further disk server options as shown in Figure 17 on page 51. Update the following fields:

Logical Volume Name The name of the first disk server volume

Use existing log vol Select from the available volumes list, or have UniTree create a new one

BlockAddress mode Set to `yes` if a disk array device is to be used

Volume Group Name The name of one of the volume groups created for UniTree

Number of Partitions The value calculated in 3.3.1.2, “Configuration Guidelines” on page 46 divided by 4MB as follows:

```
logical_volume_size / size_of_partition = number_of_partitions
```

Debug enabled This value can be set to `yes` for installation testing

Take the default values for the other fields at this time. For a complete description of their meaning, refer to 4.3, “UniTree” on page 97.

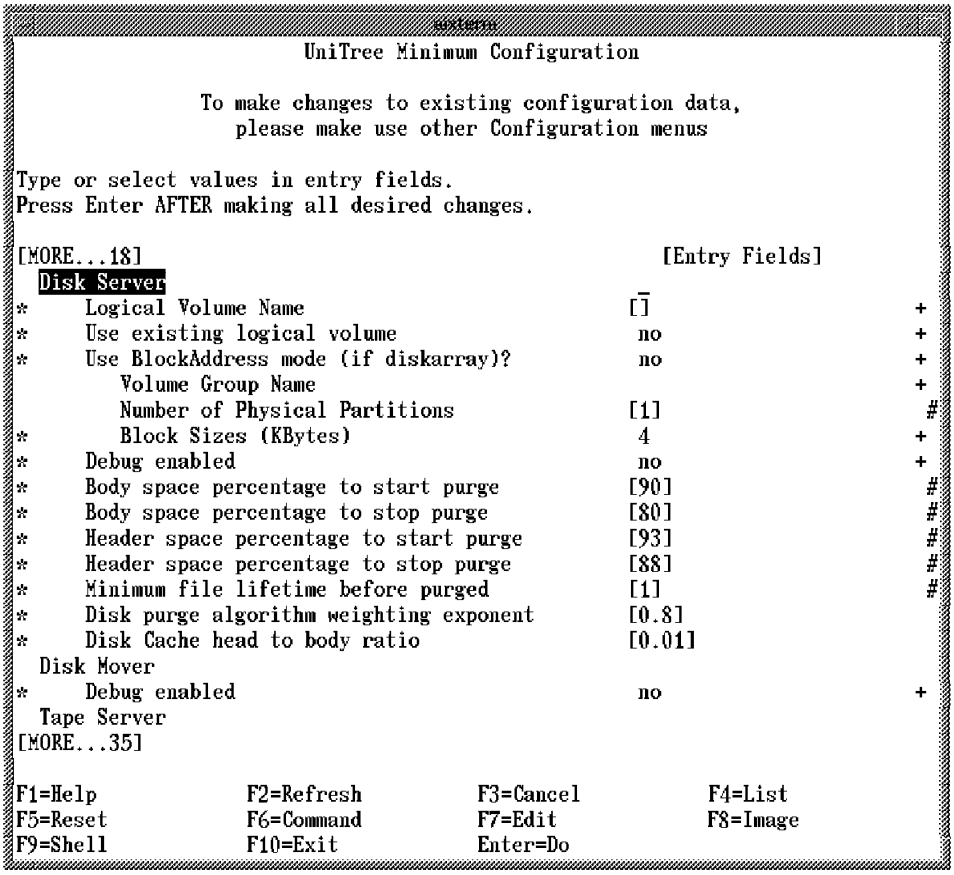

Figure 17. UniTree Minimum Configuration: Disk Server

The UniTree Minimum Configuration dialog continues with the tape server parameters, as shown in Figure 18 on page 52. Values must be entered in the following fields:

Header Name Prefix The name prefix for the pair of tape header logical volumes

Number of Partitions The number calculated in 3.3.1.2, “Configuration Guidelines” on page 46

Volume Group Name The names of the volume groups you created for UniTree

Search Table Name Prefix The name prefix for the pair of tape search table logical volumes

Number of Partitions The number you calculated in 3.3.1.2, “Configuration Guidelines” on page 46

Debug enabled This may be set to `yes` for installation testing

Num of duplicated copies This parameter sets the default number of multiple copies for each file migrated from disk to tape. Each copy is allocated on a different tape volume. It is used for availability purposes, but requires the mounting of multiple tapes (at least one for each copy)

Max number of duplicates This cannot be greater than 15

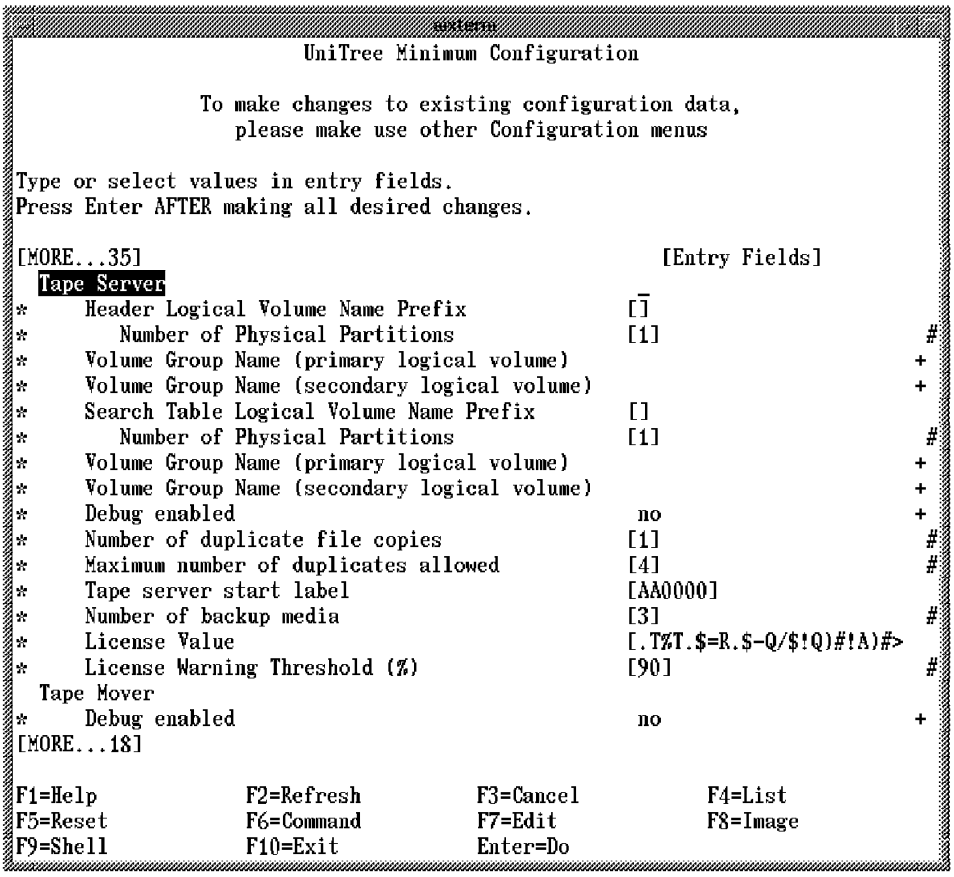

Figure 18. UniTree Minimum Configuration: Tape Server

Tape server start label The tape server creates all volume labels in numerical ascending order, starting from the one specified here. The default for archive volumes is `AA0000`. If a barcode scanner is being used, the starting label must be set to the last six digits of the first barcode label

Number of backup media The number of backup media allocated to UniTree. The minimum is three

All of the other parameters will be explained in 4.3, “UniTree” on page 97.

The remaining part of the configuration dialog is entirely devoted to migration parameters which will be explained in detail in 4.3, “UniTree” on page 97. Referring to Figure 19 on page 53, the only values that may need updating at this time are:

Debug enabled Set the debug feature to `yes` for installation testing for all the displayed UniTree servers: Migration, PVR, PDM and Repack

Polling interval Take the default value of zero (no polling) only if operator-mounted devices will not be used.

In all the other cases, this should be the period in seconds at which the UniTree Physical Device Manager performs polling of devices. A suggested value is 60

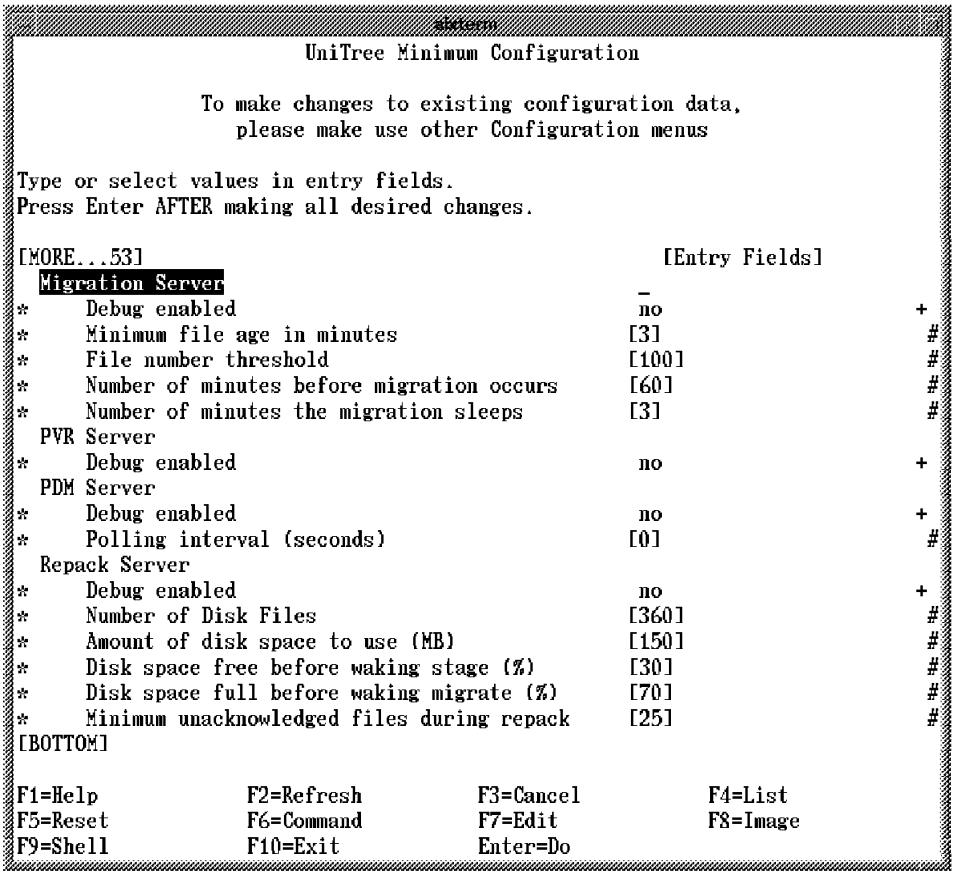

Figure 19. UniTree Minimum Configuration: Migration Server

Press Enter now to create the UniTree configuration.

3.3.3.2 Enabling NFS

The following steps must be performed if NFS is to be used for client access to UniTree. UniTree NFS cannot run together with standard NFS; every time UniTree NFS is enabled, AIX NFS is automatically disabled, and its processes are stopped.

From the SMIT menu shown in Figure 15 on page 49, select **Management of UniTree NFS**. From here, do the following:

1. Select the **Configure Unitree NFS** option, select **ENABLE** and press Enter.
2. Select the **Maintenance of UniTree NFS Export List** menu, and from here the **Add a Directory to Exports List** option, shown in Figure 20 on page 54. Enter the following values:

PATHNAME	If the UniTree directory is to be exported, set it to /
Export mode	Can be read/write or read/only
HOST allowed access	If blank, any host can access the specified UniTree file system. If specific host names are entered here, they will be the only ones allowed to access the specified directory

Root permission Can be `root-squash` or `no-root-squash`. With the latter, remote root access is granted. The former value is the recommended one

```
                        Add a Directory to Export List

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* PATHNAME of directory to export                    []                          /
* MODE to export directory                            read-write                 +
  HOST allowed client access                         []
* Permission on Root                                  root-squash                +



F1=Help             F2=Refresh            F3=Cancel             F4=List
F5=Reset            F6=Command            F7=Edit               F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 20. UniTree NFS Customization Dialog

3.3.3.3 Adding Additional Disk Server Volumes

The minimum configuration process has defined the first disk server logical volume. UniTree can function with this one at the moment, or disk server installation can be completed by adding the number of volumes calculated in 3.3.1.2, “Configuration Guidelines” on page 46.

To perform this task, from the menu displayed in Figure 15 on page 49, follow the path:

```
Management of UniTree Object
Add UniTree Object
Add Disk Server Logical Volume
```

```
                        Add a Disk Server Logical Volume

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Disk Server Logical Volume name                    []                  +
*     Logical volume (or region) already exists?     no                         +
*     Use BlockAddress mode (if diskarray)?          no                         +
  Volume Group Name                                                             +
  Number of Physical Partitions                      [1]                         #
* Block Sizes (KBytes)                               4                          +



F1=Help             F2=Refresh            F3=Cancel             F4=List
F5=Reset            F6=Command            F7=Edit               F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 21. UniTree Add Disk Server Logical Volume Dialog

Enter values according to the guidelines shown in 3.3.3.1, “Minimum Configuration Dialog” on page 49.

3.3.3.4 Configuring Devices for UniTree

A complete description of this subject can be found in *UniTree Installation and Planning Guide*. In this section, the configuration of an IBM 7208 External Tape Drive is used as a example of the actions required to perform this task.

All storage devices allocated to UniTree are used exclusively by UniTree processes and are not available to other AIX processes.

The steps that should be performed are as follows:

1. Set the device block size

   The block size parameter for the tape device to be used must be set to zero (variable). To do this, use the SMIT devices menu.

2. Assign the device to UniTree

   From the display in Figure 15 on page 49, follow the path:

   ```
   Management of UniTree Peripheral Devices
   Add Peripheral Devices
   Add operator Tape Device
   ```

   and enter the tape drive path name (/dev/rmt0 for example).

3. Label tapes

   Tapes must be labelled before UniTree can use them. The label must correspond to the value entered in the tape server configuration dialog.

   Exit from SMIT and use the `tapelabel` command to label the tapes. This command resides in /usr/lpp/Unitree/adm/bin. The syntax is:

   ```
   tapelabel device_path volume_id
   ```

   where `device_path` is the name specified in the previous step, and `volume_id` is composed of the tape label preceded by VOL1, VOL1AA0000 or VOL1BCK000, for example.

   Notice that backup volumes labels must start with BCK.

   At least as many tapes should be labelled as were specified in the default number of duplicated file copies defined in the Tape Server Configuration dialog.

   If, for example, three duplicate copies were specified, and AA0000 was selected as the starting label, then at least three tapes should be labelled AA0000, AA0001 and AA0002.

3.3.3.5 Defining Users to UniTree

Before users can access the UniTree file system, they must be defined to UniTree by the server root user. Only existing AIX users on the server machine can be defined to UniTree.

Notice also that the userid must be the same on the server and on all the clients requiring access to UniTree. This is necessary if NFS access to the UniTree file

system is required from the client, in order to preserve the correct owner and group settings for files.

From the SMIT menu displayed in Figure 15 on page 49, select **Management of UniTree Users** and from here the **Add UniTree Users** option. Insert the user's name and press Enter.

```
                                Add UniTree Users

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* User's UniTree home Directory Name                  /u
* UniTree User Name                                   nick
* UID                                                 200
* GID                                                 1
  Permissions                                        [755]                    #
  Trash Can Time-Out                                 [15]                     #



F1=Help             F2=Refresh            F3=Cancel             F4=List
F5=Reset            F6=Command            F7=Edit               F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 22. Add UniTree Users Dialog

The name of the UniTree home directory that will be created is automatically set to the same name as the AIX home directory for the specified user. Also, the UID and GID parameters are taken from the user's AIX definition. These fields cannot be modified.

The file access permissions for the new UniTree directory can be modified; the default value is 755.

When the Enter key is pressed, a new directory (in the previous example `/u/nick`) is created in the UniTree file system, and the defined UniTree user will have the specified access permission to it.

3.3.4 Running the Verification Test

The UniTree installation verification test can now be run.

- Start UniTree on the server machine:

```
/usr/lpp/UniTree/etc/rc.Unitree
```

- Test the NFS and FTP access from a client machine:

  NFS Login as a UniTree defined user, `mount` the UniTree file system locally on a previously defined mount point and `cd` to it:

```
mount -o bg,intr server_name:/ /your_mount_point
cd /your_mount_point
```

FTP Open a UniTree FTP connection, then login as a UniTree-defined user:

```
ftp server_name 1021
```

- Create files in the UniTree directory:
- As the root user of the server machine, perform the following tasks:
  1. Issue the `forcemig` command in order to force the migration process to start
  2. Browse the migration log file /usr/lpp/UniTree/adm/log/migsrv.t
  3. Monitor manual tape mount requests with the command `mntdisp`
  4. Mount the UniTree file system on the server
  5. Issue the `uls` command to check if the status of the files has changed from `disk` to `both` (after migration, user's files reside both on disk and tape)

The UniTree product has been successfully installed and customized. To perform further customization tasks, refer to 4.3, “UniTree” on page 97.

3.4 Legato NetWorker

Legato NetWorker is a client/server program providing network-wide backup and recovery capability. The major steps necessary to get this product up and running are as follows:

- Planning for product requirements
- Installation steps:
  - Server setup
  - Client setup
- Basic customization steps:
  - Enabling and registering software
  - Defining users
  - Labeling backup volumes
- Running the verification test

3.4.1 Product Requirements

NetWorker Version 4 can be installed on any RISC System/6000 diskfull workstation running AIX 3.2.3 or later. In order to use the GUI, AIXwindows System Version 2.3 or later should be also installed. At least one backup device should be configured on the server workstation.

The portmapper must be running in order for NetWorker to function. Use SMIT to start portmap or issue the `startsrc -s portmap` command from the AIX prompt.

NetWorker programs require 10MB of disk storage, about 430KB is needed for man pages, plus another 10MB of temporary space that can be removed after successful installation. Some additional space is required on the server disk for maintaining NetWorker file and media indexes.

The 10MB of storage required may be saved on client workstations by mounting via NFS the server directories containing NetWorker programs.

3.4.2 Installation

In this chapter, the main installation steps for NetWorker will be described, both on server and on client workstations. Any additional information can be found in *Legato NetWorker Installation and Maintenance Guide*.

3.4.2.1 Upgrading from Previous Releases

If the product is being installed for the first time, simply skip this section and go directly to 3.4.2.2, “Server Installation.”

Software updating is not bidirectional; once updated to version 4, regression to earlier versions is no longer possible. Be sure therefore, that all online indexes have been backed up before completing the update.

Before starting the installation of the new version, follow these steps:

- Complete a successful scheduled backup to ensure that there is a recent backup of the online indexes
- If jukeboxes are in use, deinstall the device driver and the directory where it resides
- Deinstall the NetWorker software, using the command: `nsr_ize -u -k` to kill NetWorker daemons and remove old executables
- Perform a standard installation. The indexes and configuration will be carried over from the previous version

3.4.2.2 Server Installation

The basic steps to be performed to install the server programs are:

1. Create a temporary directory
2. Extract software from the distribution media
3. Save local configuration files
4. Run the installation program
5. Update the .profile of NetWorker users

Login as root on the server machine and create a temporary directory large enough to hold about 10MB of data.

Decide the definitive location for NetWorker code; the default location is in /usr/bin for programs (all of them are prefixed by `nsr`), while a new directory named nsr is created in /usr for the index structures. The default for man pages is the /usr/man directory.

Now the software can be extracted from the distribution media. The block size of the tape device that will be used should be changed before running the AIX `tar` command.

These preliminary steps are performed with the following commands:

```
# mkdir /usr/tmp/nsr_extract
# cd /usr/tmp/nsr_extract
# /etc/chdev -l rmt0 -a block_size=10240
# tar xvpf /dev/rmt0
```

Before running the installation program, notice that NetWorker changes some configuration files:

- /etc/inittab
- /etc/rpc
- /etc/syslog.conf

It is prudent to save a copy of these files before going on with the installation.

Run the installation program:

```
# nsr_ize -i -s
```

and answer the installation script requests about the definitive location for the NetWorker programs.

After the successful installation of the product, add the path of NetWorker programs to the PATH variable in the .profile of root and any other NetWorker user on the workstation.

The temporary directory may now be removed. It may be exported first in order to install client systems that do not have tape drives and removed after their installation.

3.4.2.3 Client Installation

Login as root on the client workstation. First of all, the type of installation required needs to be decided:

- A complete installation from the distribution tape is the same as for the server; in this case, 10MB of temporary disk space is required, plus the 10MB required for NetWorker programs.
- Mounting the temporary directory created on the server via NFS, saving 10MB of temporary space on your workstation.
- Not loading the NetWorker programs on the client. In this case, the server directory in which the NetWorker program code resides must be mounted, both at installation time and at any time the client product is to be used. If this option is selected, it is recommended you install the NetWorker programs in a separate directory in order to avoid the mounting of the server /usr/bin directory on the client.

After the previous choice, the remaining steps for the client code installation are as follows:

1. Update the /etc/hosts file
2. Save local configuration files
3. Run the installation program
4. Update the .profile of NetWorker users

Edit the /etc/hosts file, locate the record with the NetWorker server name and add to it the NetWorker alias `nsrhost`:

```
9.3.1.99      pippin
9.3.1.85      dori nsrhost
```

If installing the product on a client machine, it may be prudent to save the configuration files before running the installation program, as described in the previous section.

Run the installation program:

```
# nsr_ize -i -c
```

If not installing the code on the client, answer `no` to the prompt:

```
Install the ibmrs6000 NetWorker programs? [yes] __
```

and enter the complete path of the mounted directory in which the programs reside.

When the installation script completes, add the path of the NetWorker programs to the .profile of the NetWorker users on the workstation.

When the product has been installed on all the clients, the temporary directory may be removed from the server file system.

For information about the installation of jukebox drivers, refer directly to *Legato NetWorker Installation and Maintenance Guide*.

3.4.3 Customization

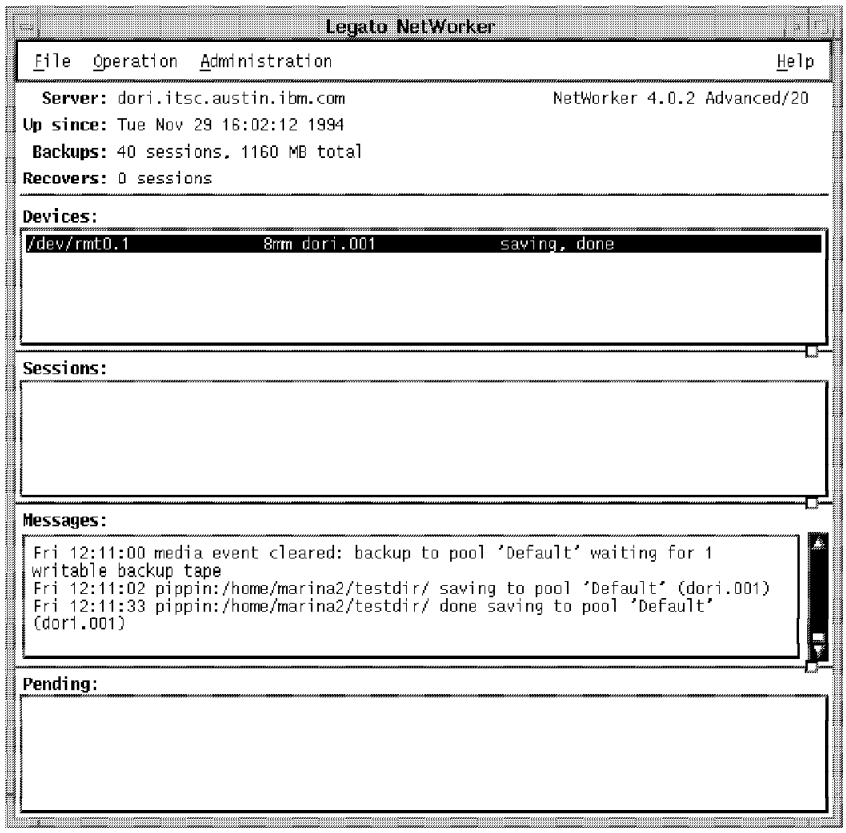

Figure 23. Legato NetWorker GUI Main Panel

The server daemon has been started by the installation script. To perform initial customization, login to the server system and enter:

```
networker &
```

This command starts the NetWorker GUI. The main panel appears as shown in Figure 23.

3.4.3.1 Enabling and Registering the Code

NetWorker also provides a character-based administrative program to manage the server without using the AIXwindows program; refer to 4.4, “Legato NetWorker” on page 113 for additional information and examples.

NetWorker commands are also available to perform the most common user tasks. They are also useful when recovering from failures or any time there is a main memory shortage. The complete list of available commands can be found in Appendix D of *Legato NetWorker Installation and Maintenance Guide*. More details about command syntax may be found using the man pages located in the /usr/man/man8 directory.

For each NetWorker product ordered, a separate 18-digit code will be received which is necessary to enable the product on the system.

Login as root on the server, and from the AIX command prompt, issue the command:

```
# nsrcap -v -c xxxxxx-xxxxxx-xxxxxx
```

Once all of the NetWorker products have been enabled, ensure that they are registered within 45 days. After this time, all unregistered code is automatically disabled.

To register the code from the GUI interface, select the **Administration** menu and from here the **Registration** panel. From here, select first the **View** menu, and change the display mode to **tabular**.

```
                                  Registration
 File  View                                                                Help
================================================================================
name                 enabler code         host id     expiration date  auth code

10 clients           4ccece-050091-c9161f 01319810    Jan 14, 1995
NetWorker Advanced/1 4dcccf-1a4192-c1851e 01319810    Jan 13, 1995
```

Now the information on this screen can be printed by selecting the **File** menu and the **print** option.

Mail or fax this information to Legato in order to obtain the authorization code. When the authorization code is received, enter this information in the Auth code field and click on the **Apply** button. Now the product is registered, and the expiration date is set to `none`.

3.4.3.2 Defining Clients

After the installation, the server workstation can be automatically backed up as a client of itself, using the default parameters. In order to back up other client workstations, define them by selecting the **Clients** display from the Administration menu. The display will be as shown in Figure 24 on page 63.

Click on the **Create** button, and enter the client name in the Name field. At this time simply accept the default values, so therefore click on the **Apply** button. A complete description of this function can be found in 4.4.2, “User Administration” on page 122.

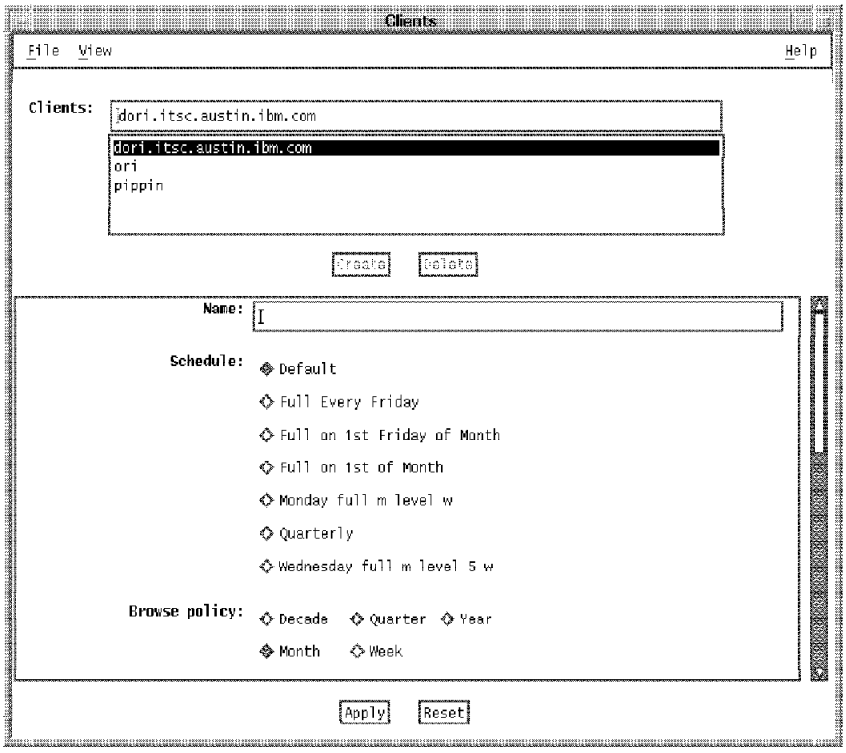

Figure 24. NetWorker GUI Client Window

3.4.3.3 Labelling Backup Volumes

All the backup volumes must be properly labelled before being used by NetWorker. Any backup volume belongs to a volume pool. There are some predefined volume pools that may be used, or new pools can be defined. In either case, each volume pool has an associated label template, and every backup volume in that pool is labelled according to it.

At startup, the only enabled pool is the default pool in which all volumes can be defined and labelled with the name of the server. This can be used initially, and 4.4.1, “Device Administration” on page 114 can be referred to for additional information.

Insert blank backup media into the device and from the NetWorker main panel, select the **Operation** menu, and from here the **Label and Mount** screen. NetWorker displays the Label and Mount window and starts analyzing the media. The first time, only the default and archive pools will be displayed since they are the only enabled pools at startup time.

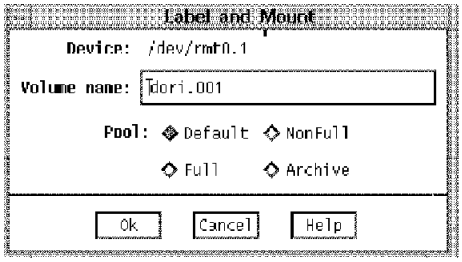

Figure 25. NetWorker GUI Operation: Label and Mount Window

Either select the default pool and accept the default label, or enter a volume name. Click on the **Ok** button.

Now the verification test can be run.

3.4.4 Running the Verification Test

To verify NetWorker's functionality, login as a normal user on a client machine and start the NetWorker GUI. From the main panel, it can be seen that the Administration menu is not available, indicating a normal user interface.

- Select the **Operation** menu and from here the **Backup** display. The Backup window appears showing the home directory
- Expand the current directory if required, by selecting **View** and **Expand One Level**
- Click with the mouse on files or directories that are to be backed up. All nested files and subdirectories will be marked
- Select the **File** menu and from here the **Start Backup** display; the Backup Options dialog will appear

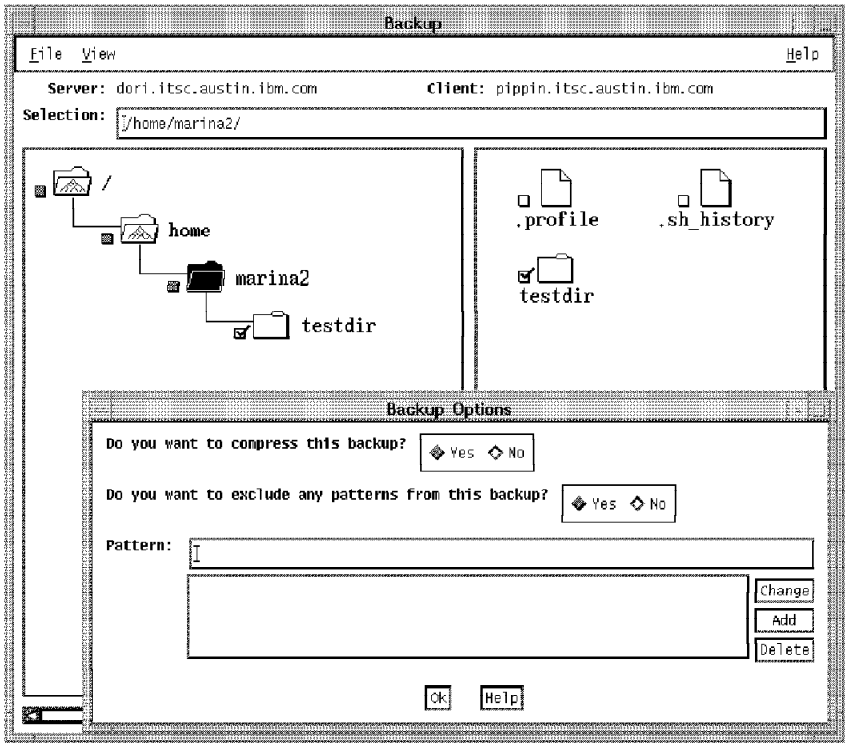

Figure 26. NetWorker GUI Operation: Backup Windows

- Decide if compression is required for the files and if any are to be excluded from the backup. Click on **Ok**
- The Backup Status will appear allowing the progress of the backup to be monitored.

  If the backup is not proceeding as expected, check in the Pending display on the NetWorker main window to be sure that the correct backup device is currently mounted.

When the backup is finished, exit from the window by clicking on the **Cancel** button and selecting **Exit** from the File menu

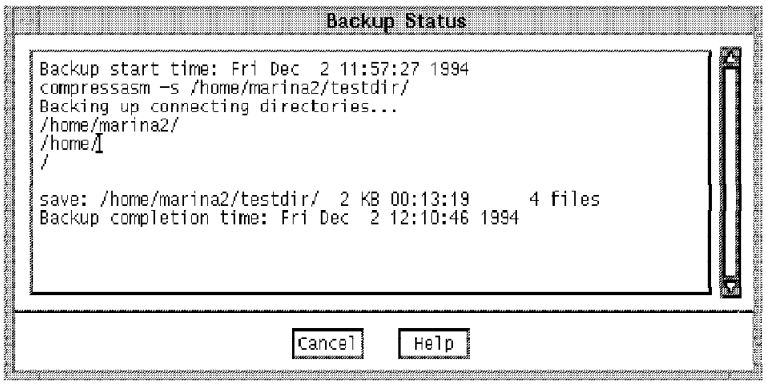

Figure 27. NetWorker GUI Backup Status Display

3.5 Product Comparison Summary

The products discussed in this document will be compared with regard to installation and customization on the basis of the following criteria:

- Central management
- Ease of use

3.5.1 Central Management

Central management with regard to installation and customization refers to how much of the process can be accomplished centrally at the server.

3.5.1.1 ADSM

ADSM consists of server code, administrator code and client code. Each component must be accessible at the system that it will be executed from; it can be installed, or the executable code can be NFS mounted, at the system where it will be executed. Configuration must be performed on the system where the component will be executed from.

3.5.1.2 FSF/6000

FSF/6000 does not include specialized server software. Instead, it makes use of an NFS or ADSM server. Installation of the FSF software is therefore managed from the client machine where the product is to be installed. Customization of FSF is also accomplished locally at the client.

If an NFS server is to be used, the server machine requires that NFS is enabled and a directory or file system be exported for use by FSF. This can be executed remotely if communications has already been enabled.

If an ADSM server is to be used, some configuration of the ADSM server is required. This can be accomplished via a local ADSM administration client, if available; otherwise, it must be performed from the ADSM server console.

3.5.1.3 UniTree

UniTree provides its services through existing NFS or FTP communications. For this reason, installation and configuration of the product is required only at the server system where it is to be installed.

3.5.1.4 Legato NetWorker

NetWorker consists of client and server components that must be installed on the systems where they will be used. Server installation and customization should be performed at the server machine though once this has been done, client installation can be accomplished in several ways.

- Complete installation from distribution media at client
- Mounting server temporary install directory on client via NFS and installing from this
- Mounting server code directory via NFS on the client and using the required programs directly

The latter option minimizes the remote management of the product required as all code can be maintained from the server system.

3.5.2 Ease of Use

Ease of use with regard to installation and customization pertains simply to the scope of effort and general ease of the process.

3.5.2.1 ADSM/6000

The installation interface for all ADSM components is SMIT though on occasion it may be necessary to use the `installp` command directly with the `-F` option. This is only necessary where a previous release requires overwriting, and the version of SMIT utilized does not allow the `-F` option.

The configuration process consists of three parts:

- Updating options files
- Defining required server components
- Loading the kernel extension

Updating the options files is a required process. This can be accomplished manually, by editing the relevant files, or with the GUI provided by the EZADSM interface. Many of the options can be allowed to default in order to get client and server to communicate initially. Further customization can again be manual or via the EZADSM GUI.

Defining the required server components, such as volumes and logs, can be performed manually or initiated via macros. A default set of macros is provided with the product to ease initial setup. Further definition of volumes can be accomplished from the product GUIs.

Loading the kernel extension is accomplished manually, but can be executed from the rc.tcpip file in order to automate the process.

3.5.2.2 FSF/6000

Installation and configuration of FSF is accomplished via the SMIT interface. The following steps need to be performed:

- Creating an FSF file system
- Mounting the FSF file system
- Fine tuning FSF parameters

The process of setting up the product also requires definition of a server store.

If the server store is NFS, then a directory or file system must be made available from an NFS server. This can be accomplished via SMIT at the server.

If the server store is ADSM, then a filespace must be registered manually using an FSF utility locally and definition work performed at the ADSM server to authorize the FSF client.

3.5.2.3 UniTree

Prior to installation of the UniTree product, a number of planning-related calculations must be made in order to determine space requirements for the various elements of UniTree. Once this has been performed, the SMIT interface can be used to install the product.

Configuration is also via the SMIT interface and involves the following steps:

- Defining parameters for each of the UniTree components
- Enabling UniTree NFS if it is to be used as the communications mechanism
- Configuring the UniTree devices
- Defining users to UniTree

3.5.2.4 Legato NetWorker

Installing the NetWorker server product is a manual process requiring the following steps:

- Creating a temporary install directory
- Extracting the software from the installation media using `tar`
- Saving local files that will be modified by the configuration
- Running the install program
- Updating the .profile of NetWorker users

The install program performs basic configuration of the environment though update of any user profiles must be executed manually. As was stated in 3.5.1, “Central Management” on page 65, the client code can be installed in a similar fashion from a client mount of the temporary directory or by mounting the directory in which the NetWorker programs reside at the client. This is a manual process in all cases.

Finally, the NetWorker code at client and server needs to be enabled before it can be used. Registration with Legato is also necessary, via mail or fax, or the code will be disabled after 45 days.

Chapter 4. Product Administration

This chapter describes the various administration tasks required for each product. Examples of the most common procedures can be found here, such as user and device definition and security administration.

The main intent is to provide the most general and comprehensive description of the administrative features of each product in order to allow a comparison to be drawn between ease of use as well as range of function.

4.1 ADSTAR Distributed Storage Manager/6000

ADSM administrators can manage the server from a server console or administrative client almost anywhere in the network. The administrative client is available on most platforms with a command line interface and on AIX, HP, Sun, and OS/2 platforms with a GUI as well.

The major administrative tasks can be summarized as follows:

- Setting up the environment
- Monitoring client/server activities
- Updating the configuration
- Auditing

The two main functions of ADSM system administration are:

1. Managing storage pools and database
2. Registering and controlling administrators and client nodes

These will be the subjects of the next two sections: device administration and user administration. In both cases, there will be a functional description of the major tasks involved, together with examples, in order to allow an easy comparison with the other products.

The final section is devoted to a brief discussion of the security administration capabilities of the ADSM product.

4.1.1 Device Administration

After finishing the initial setup of the ADSM server, the physical storage requirements must be evaluated. By identifying the storage devices and the different types of media available, devices can be categorized by their characteristics in the areas of capacity, availability and performance and the necessary *storage pools* defined. A storage pool can be viewed as a collection of storage volumes. Every storage pool is associated with a device class; ADSM supports the following device classes:

- DISK
- 8MM
- CARTRIDGE
- OPTICAL
- FILE

Since a device class is either random access or sequential access, each storage pool can be classified according to the access strategy used to access volumes. Only storage pools using devices of type DISK are random access.

4.1.1.1 Disk Storage Pools

At installation, ADSM provides two predefined disk storage pools named BACKUPPOOL and ARCHIVEPOOL (ADSM V2 will also provide another disk storage pool named SPACEMGPOOL). In order to define additional disk storage pools:

```
adsm> define stgpool POOL_NAME disk
ANR2200I Storage pool POOL_NAME defined (device class DISK).

adsm>
```

In order to control costs and still provide appropriate levels of performance when accessing storage, multiple storage pools can be linked to form a storage hierarchy. For example, faster disks can be used for a backup pool in order to provide rapid response to user requests, and when the pool becomes full, the older data can be automatically moved to the next defined pool, which could be optical or tape.

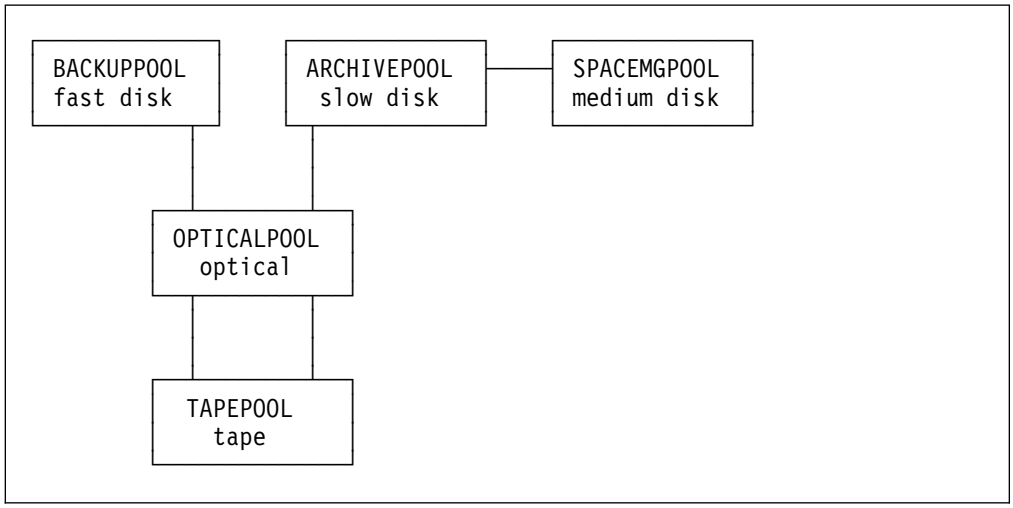

Figure 28. Sample Storage Pool Structure

Figure 28 shows an example of a storage hierarchy where:

- Faster disk volumes are used to backup/restore user data
- Slightly slower disks are used for space management services
- Slowest disks are reserved for archiving
- Optical volumes are next in the hierarchy
- Lastly, 8mm tapes can be used

To define a storage hierarchy with the previously defined storage pools, use the following:

```
adsm> update stgpool POOL_NAME next=NEXT_POOL
```

The sample storage pool structure has been implemented in the test system. The ADSM display is:

```
adsm> query stgpool *
Storage      Device      Estimated  %Util  %Migr  High   Low  Next
Pool Name    Class Name   Capacity                Mig%  Mig%  Storage
                              (MB)                       Pool
-----------  ----------  ---------  -----  -----  ----  ----  -------
ARCHIVEPOOL  DISK            200.0    3.2    3.2    90    70  OPTICALPOOL
BACKUPPOOL   DISK          1,000.0   90.0   90.0    90    70  OPTICALPOOL
OPTICALPOOL  OPTICAL       1,000.0    0.0    0.0    90    70  TAPEPOOL
SPACEMGPOOL  DISK            100.0   14.1   14.1    90    70  ARCHIVEPOOL
TAPEPOOL     TAPE              0.0    0.0    0.0    90    70

adsm>
```

The disk storage pools are empty, and in order to use them, volumes must be defined to them. The volumes must be defined and formatted as described previously in 3.1.1.1, “Installing the ADSM Server” on page 29.

To define volumes in a disk storage pool, use the following command:

```
adsm> define volume POOL_NAME /dev/rdsmvol
ANR2206I Volume /dev/rdsmvol defined in storage pool POOL_NAME (device class
DISK).

adsm>
```

Further examples of using the Graphical User Interface can be found in *Getting Started with ADSM/6000*.

4.1.1.2 Optical and Tape Storage Pools

If tape or optical devices are to be used, the following steps should be followed:

1. Define the device to AIX.

   Start SMIT and define the devices to AIX using the ADSM device drivers. A physical tape drive can be shared by ADSM and other AIX applications.

2. Define a library.

   The library is a collection of one or more drives and possibly robotic devices. Library types are: MANUAL, SCSI and 349x.

3. Define drives.

   For each device defined, an ADSM drive must be defined and associated with a library.

4. Define a device class.

   A unique device class should be defined to support the sequential access devices available at the installation. Only the DISK device class is predefined.

5. Define a new storage pool.

6. Add volumes to the storage pool.

For 8mm and cartridge, specify a one- to six-character alphanumeric volume name. For optical device types, specify a one- to 32-character alphanumeric volume name.

7. Label the volumes before use.

Two possible scenarios follow:

IBM 7208-01 8mm Tape Drive: SMIT defines a device named `/dev/mt0`. The following commands must be issued:

```
define library TAPELIB libtype=manual
define drive TAPELIB tape1 device=/dev/mt0
define devclass TAPE devtype=8mm library=TAPELIB
define stgpool TAPEPOOL TAPE
define volume TAPEPOOL xxxxxx
```

IBM 3995 Optical Library

SMIT defines a device named `/dev/op0`. The following commands must be issued:

```
define library OPTICLIB libtype=scsi
define drive OPTICLIB opt01 device=/dev/op0 element=scsiaddress
define devclass OPTICAL devtype=opt library=OPTICLIB
define stgpool OPTICALPOOL OPTICAL
define volume OPTICALPOOL  xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

4.1.1.3 File Storage Pools

When the device class is defined as `FILE`, the storage pools are not physical units; they are standard files in the specified file system.

```
define devclass FILE devtype=file directory=/usr/local/adsm/files
define stgpool FILEPOOL FILE
define volume FILEPOOL /usr/local/adsm/files
```

If an AIX logical volume is being used, the directory to be entered is /dev/r`xxx` where `xxx` is the logical volume name.

4.1.1.4 Management Issues

This section describes some of the most common issues for storage device administration, together with the corresponding ADSM commands, in order to show the capabilities and functionality of the ADSM administration client. For a complete discussion, refer to *ADSM/6000 Administrator's Reference* and *ADSM/6000 Administrator's Guide*.

Dynamically Changing Storage Pools Access Mode: If user access to storage pools needs to be curtailed (for example during maintenance operations), the `access` option can be used:

```
update stgpool POOL_NAME access=(READWrite | READOnly | UNAVailable)
```

Setting Parameters to Manage the Files in a Storage Pool

- Maximum file size allowed in a pool
- Migration parameters (refer to Chapter 9, “Server Hierarchical Storage Management” on page 209)
- Use of *collocation*. All data belonging to a client can be selected to be *collocated* on a minimal number of sequential devices. In this way, the number of mount operations, and therefore access time, can be reduced. On the other hand, some storage space is sacrificed.

```
update stgpool POOL_NAME maxsize=n collocation=(yes|no)
```

Usage of Scratch Volumes: ADSM can dynamically acquire volumes when needed and automatically delete them when they become empty. Using this method means that there is no control over which volumes are used in the pool, but then, not every volume that is to be used needs to be explicitly defined. Note that the `maxscratch` parameter is not valid for devices of class DISK.

```
update stgpool POOL_NAME maxscratch=n
```

Limiting the Number of Mount Operations: With the `mountlimit` option, the maximum number of volumes that can be simultaneously mounted for a given device class can be specified. It must be less than or equal to the number of associated physical drives available. In the case of FILE device classes, it is used to restrict the number of files concurrently opened for access.

```
update devclass CLASS_NAME mountlimit=n
```

Changing Database and Recovery Log Capacity: ADSM administrators can dynamically increase the database and recovery log capacity by defining new volumes to AIX and allocating them to ADSM. Never change the size of an already allocated ADSM volume because the server uses the initial allocation to calculate the placement of data.

increase Define the new volume to ADSM, and extend the previous total capacity adding the new volume's capacity

decrease Reduce the total capacity, using the information displayed by the system, and then delete the volume(s) from the database.

```
adsm> q dbvol

Available Assigned   Maximum   Maximum    Page    Total    Used %Util   Max
    Space Capacity Extension Reduction    Size   Usable   Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)    Pages
--------- -------- --------- --------- ------- -------- ------- ----- -----
        8        8         0         4   4.096    2,048     216  10.5  46.2

adsm> reduce db 4
ANR2250I Database assigned capacity has been reduced.
```

```
adsm> q dbvol

Available Assigned   Maximum   Maximum    Page    Total    Used %Util   Max
    Space Capacity Extension Reduction    Size   Usable   Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)    Pages
--------- -------- --------- --------- ------- -------- ------- ----- -----
        8        4         4         0   4.096    2,048     215  21.0  92.3

adsm> delete dbvol /dev/rdsmdb1
ANR2243I Database volume /dev/rdsmdb1 deleted.

adsm> q dbvol

Available Assigned   Maximum   Maximum    Page    Total    Used %Util   Max
    Space Capacity Extension Reduction    Size   Usable   Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)    Pages
--------- -------- --------- --------- ------- -------- ------- ----- -----
        4        4         0         0   4.096    2,048     215  21.0  92.3

adsm>
```

Figure 29. Sample Session for Decreasing Database Capacity

4.1.1.5 Monitoring and Auditing

Any administrator can query the server for information about storage volumes. Administrators can also audit or move files from a volume in any storage pool. For example, in order to manage data storage, administrators can:

Monitor the Occupancy of a Storage Pool

```
adsm> q occupancy stg=backuppool

Node Name              Filespace    Storage      Number of     Space
                       Name         Pool Name        Files  Occupied
                                                          (MB)
---------------------  -----------  -----------  ---------  --------
ORI                    /home        BACKUPPOOL           2      0.00
ORI                    /var         BACKUPPOOL          36      0.39
PIPPIN                 /            BACKUPPOOL         517      3.78
PIPPIN                 /var         BACKUPPOOL          36      0.43
PIPPIN                 /home        BACKUPPOOL          15      3.21
PIPPIN                 /tmp         BACKUPPOOL           8      2.97

adsm>
```

Auditing a Volume: In order to check for any inconsistency between the database information and the storage pools, the storage pool volumes can be audited.

```
audit volume /dev/rdsmfast fix=no
```

Moving Files: In order to move files from one volume to another or into a different storage pool:

```
move data VOLNAME stg=NEW_STG_NAME
```

Requesting Information About Volume Contents: In order to find out information about the contents of a volume:

```
adsm> query content /dev/rdsmfast filespace=/home

Node Name               Type Filespace  Client's Name for file
                             Name
----------------------- ---- ---------- --------------------------------
PIPPIN                  Bkup /home      /marina/ .profile
PIPPIN                  Bkup /home      /marina/ .sh_history
PIPPIN                  Bkup /home      /marina/ .xdt3

adsm>
```

Deleting Workstation Files from Storage Pools: In order to delete files from a storage pool:

```
adsm> delete filespace ORI /home
Do you wish to proceed? (Yes/No) yes
ANS5104I Process number 87 started

adsm>
```

Deleting Volumes from a Storage Pool: In order to delete volumes from a storage pool:

```
delete volume VOLNAME
```

In the same way, administrators can easily manage:

- Libraries and drives
- Storage volumes in automated libraries
- Mount operations
- Database and recovery log buffer pools

```
adsm> query log format=detailed

  Available Space (MB): 12
Assigned Capacity (MB): 12
Maximum Extension (MB): 0
Maximum Reduction (MB): 8
     Page Size (bytes): 4.096
    Total Usable Pages: 2.560
            Used Pages: 170
                 %Util: 6.6
            Max. %Util: 17.2
      Physical Volumes: 1
        Log Pool Pages: 24
    Log Pool Pct. Util: 12.38
    Log Pool Pct. Wait: 0.00

adsm>
```

Figure 30. Recovery Log Detailed CLI Display

The database buffer pool provides cache storage to allow pages to remain in memory for longer periods of time. The recovery log buffer pool is used to hold new transaction records until they can be written to the log.

4.1.2 User Administration

The administration issues related to the management of storage devices have just been covered. This next section will look at the issues specifically related to users: administrators and client nodes.

4.1.2.1 Administrative Authority

The ADSM system administrator can register other administrators to the server, specifying the new administrator userid and password. After users are registered as administrators, they can access the server from any computer in the network on which the administrative program is installed.

ADSM provides a hierarchical structure to the authority that can be granted to an administrator. Thus, as flexible an administration scheme as is required can be established while still providing control over the system.

The administrative privilege classes are:

System Allows an administrator to perform any ADSM function.
Policy The administrator can define and assign policy definitions, register clients and schedule backup and archive functions.
Storage The administrator can monitor and control the storage resources for the server.
Operator The administrator can only control server operations.
Analyst The administrator can reset statistics and issue trace commands.

Any administrator can query the server for information about ADSM functions.

```
adsm> register admin admin1 <password>
ANR2068I Administrator ADMIN1 registered

adsm> grant authority admin1 classes=storage
ANR2079I Unrestricted storage privilege granted to administrator ADMIN1.

adsm>
```

Figure 31. Administrator Definition with Overall Storage Authority

Administrative authority can also be divided up by organization. For example, the logical categories of authority can be given to a department but only for the data belonging to that department, using the `restricted` policy and storage privileges:

```
adsm> register admin adm_dep1 <password>
ANR2068I Administrator ADM_DEP1 registered

adsm> grant authority adm_dep1 domains=dep1_dom
ANR2078I Restricted policy privilege granted to administrator ADM_DEP1 - policy
domain DEP1_DOM.

adsm> grant authority adm_dep1 stgpools=dep1*
ANR2078I Restricted policy privilege granted to administrator ADM_DEP1 - storage
domain DEP1*.

adsm>
```

Figure 32. Administrator Definition with Restricted Privileges

4.1.2.2 Managing Administrators

As the system administrator, you can manage administrator access to the server using the following commands:

lock	To temporarily prevent an administrator access the system
query	To request information about administrators
remove	To remove an administrator from the system
rename	To rename an administrator
revoke	To reduce administrator authority
unlock	To resume system access for a previously locked administrator
update	To change an administrator password

```
adsm> query admin format=detailed

       Administrator name: ADMIN
    Last Access Date/Time: 11/01/1994 11:12:34
   Days since last access: <1
   Password Set Date/Time: 10/28/1994 11:10:49
  Days Since Password Set: 4
                  Locked?: no
                  Contact:
         System Privilege: yes
         Policy Privilege: ** Included with system privilege **
        Storage Privilege: ** Included with system privilege **
        Analyst Privilege: ** Included with system privilege **
       Operator Privilege: ** Included with system privilege **
   Registration Date/Time: 10/28/1994 11:10:49
Registering Administrator: SERVER_CONSOLE

       Administrator name: MARINA
    Last Access Date/Time: 11/01/1994 11:03:18
   Days since last access: <1
   Password Set Date/Time: 11/01/1994 11:03:10
  Days Since Password Set: <1
                  Locked?: no
                  Contact:
         System Privilege:
         Policy Privilege:
        Storage Privilege: ** Unrestricted **
        Analyst Privilege:
       Operator Privilege:
   Registration Date/Time: 11/01/1994 11:03:10
Registering Administrator: ADMIN

       Administrator name: SERVER_CONSOLE
    Last Access Date/Time:
   Days since last access:
   Password Set Date/Time:
  Days Since Password Set:
                  Locked?: no
                  Contact:
         System Privilege:
         Policy Privilege:
        Storage Privilege:
        Analyst Privilege:
       Operator Privilege: yes
   Registration Date/Time: 10/27/1994 14:34:56
Registering Administrator:

adsm>
```

Figure 33. Query Administrator Detailed CLI Display

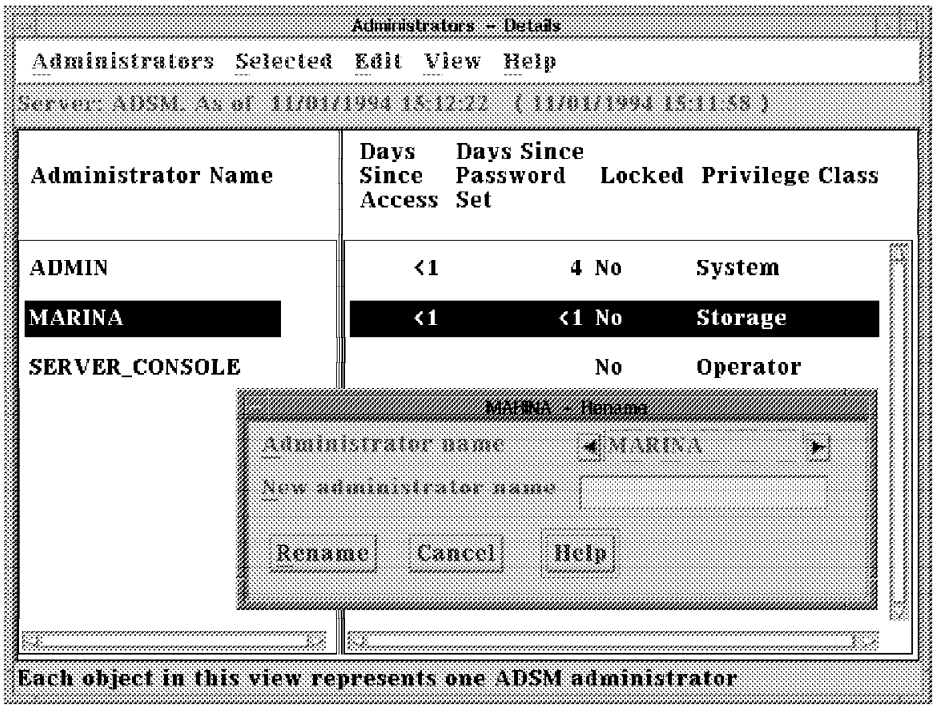

Figure 34. Rename Administrator GUI Display

4.1.2.3 Client Node Registration

An administrator with system or policy authority can register client nodes to the system. ADSM provides two different methods for registering workstations to the server:

open With open registration, any user can register their own workstation. The first time the access is attempted to the server, a prompt for the password is presented, and the server automatically registers the workstation.

closed With closed registration, the administrator must register each workstation as a client node to the server before users can connect to the server.

To register a client node, the administrator must define:

- The workstation node name (it must be the same as the value returned by the HOSTNAME command)
- The password and, optionally, the contact information
- Whether files should be compressed before sending to the server
- Whether the user is allowed to delete backed-up files from storage
- Whether the user is allowed to delete archived files from storage

With open registration, the server allows each user to choose whether to compress files or to delete archived copies from storage. It does not allow deletion of backed-up copies.

To register client nodes, the administrator can use the Graphical User Interface, or the MACRO facility from the Command Line Interface (CLI).

Using Macros: Administering an ADSM system involves using a variety of commands, often repetitively. This can be a tedious job, and one that is prone to mistakes. The ADSM administrator interface provides a means of avoiding some of these problems: the ability to enter one or more administrator commands in a macro. An ADSM macro is a file that contains one or more administrator commands in exactly the same format as would be used through the administrator CLI. When the macro is executed, the commands are entered and processed sequentially.

```
adsm> macro myfile.mac
```

4.1.2.4 API Registration

ADSM provides an Application Programming Interface (API) that can be used by anyone to integrate a storage management solution with existing applications to ensure that critical data is protected and easily recoverable. When an application uses ADSM's API, it becomes an ADSM application client that communicates with the ADSM server to backup, archive or recover objects from ADSM storage.

Before users can request services from the ADSM server through the API, an administrator must register the workstation as a client. On the other hand, after the application client is installed, the user must modify the client options file to identify the node name of the client and the communication method used to connect to the server.

To register an application, using the ADSM API as a client node, the administrator must define:

- Workstation node name
- Password and client information
- Compression status of the client
- Policy domain to which the workstation belongs

4.1.2.5 Managing Client Nodes

System and policy administrators can manage client nodes using the following commands:

delete To delete a client node filespace from storage pools
lock To temporarily prevent the client from accessing the ADSM server
query To request information about client nodes and their filespaces
remove To remove the client node from the server
rename To rename client node
unlock To resume system access for a previously locked client node
update To change client node attributes:

- Password
- Assigned policy domain
- File compression
- Backed up and/or archive file deletion

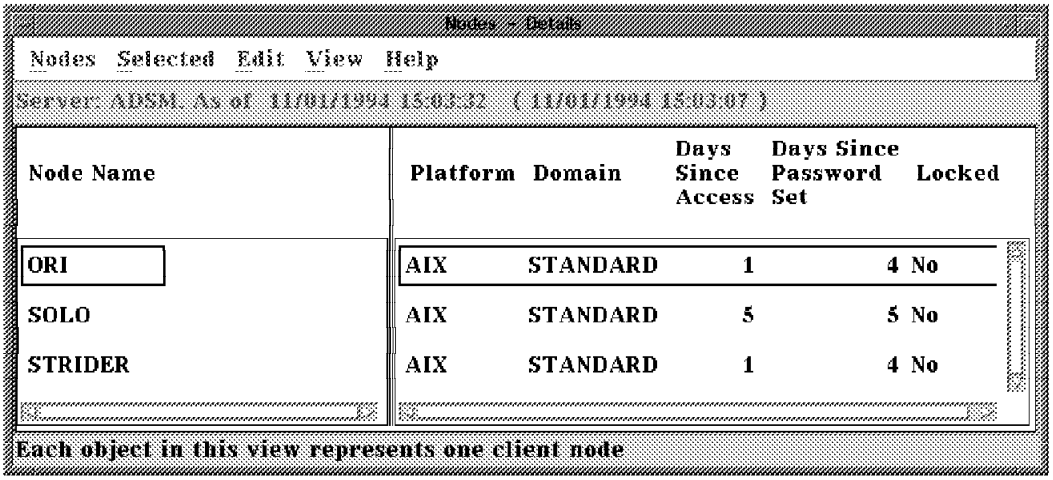

Figure 35. Node Management GUI Display

4.1.2.6 Managing Schedules

The policy administrator can set up schedules to automate the backup and archive processes, and with ADSM V2, restore, retrieve, client operating system commands, ADSM macros, and ADSM server commands. The schedules can be managed with the following commands:

copy schedule	Creates a copy of an existing schedule
define association	Associates one or more clients with a schedule
define schedule	Defines a new schedule
delete schedule	Deletes a schedule from the database
query schedule	Displays information about one or more existing schedules
set maxcmdretries	Sets the maximum number of retries after a failed attempt to execute a scheduled command
set retryperiod	Sets the time between the retry attempts
update schedule	Changes the attributes of a schedule

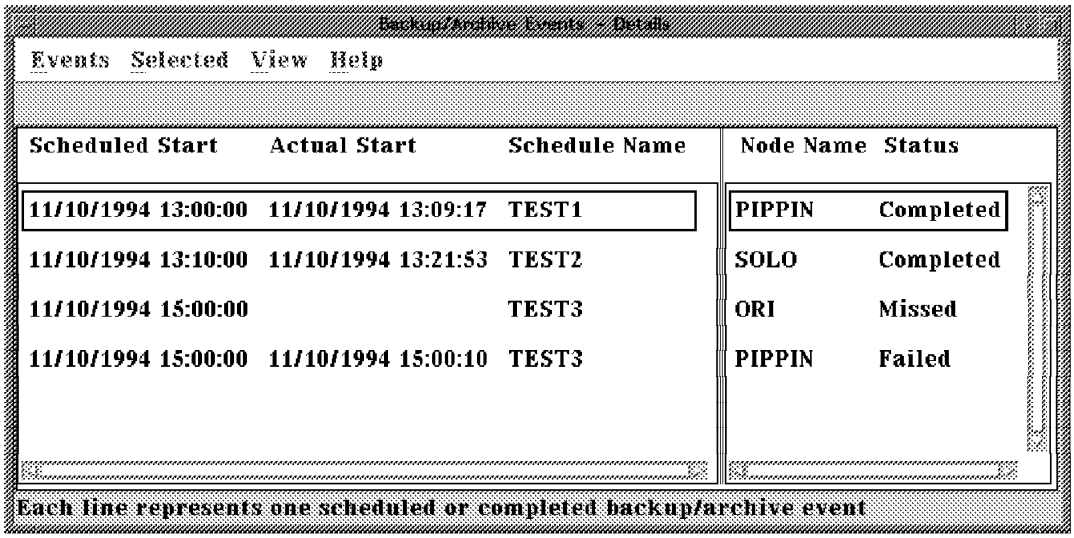

Figure 36. Administrator Event GUI Display

For a description of the ADSM scheduling facility, refer to 5.2.2, “Central Scheduling” on page 141.

The policy administrator will want to ensure that all defined nodes complete their designated operations according to schedule. Scheduled and/or completed events for selected policy domains and nodes can be displayed and the display limited to specific date and time ranges. The command `query event` or the Graphical User Interface can be used.

4.1.3 Security Administration

One of ADSM's strengths is its diverse range of supported platforms. Consequently, it doesn't use an existing system-specific security system. Currently, ADSM uses a client/server mutual suspicion authentication system to ensure that an authorized client is communicating with an authorized server and that the server is communicating with registered administrators and client nodes.

4.1.3.1 Client/Server Authentication

ADSM provides client/server authentication by requiring a password from each client (administrative, node, application) registered with a server. The system administrator can:

- Set on/off the password authentication (the default is `on`)
- Set a password expiration period (the default is 90 days)

```
set authentication on|off
set passexp n
```

4.1.3.2 ADSM Packaging and Licensing

This section looks at the new packaging and licensing options that are employed with ADSM V1.2.1 and V2.

Packaging: ADSM is packaged in two different ways:

1. Build-to-plan (BTP)

   Also referred to as shrink-wrapped, BTP means that the ADSM product is pre-packaged in a specially designed box so that it can be easily sold through local stores. The ADSM OS/2, HP, and Sun servers are available BTP; ADSM V2 is also being considered for this mechanism.

2. Build-to-order (BTO)

   This mechanism uses the traditional IBM ordering process to provide the product with the exact features that the customer requires. The customer places an order stating the exact quantities of each feature required and a box is packaged to these specifications. All servers except HP and Sun are BTO.

Licensing: Licensing compliance is not enforced under AIX, as the ADSM server no longer uses the AIX NetLS license manager. Instead, license compliance is managed by a component of ADSM which uses two avenues of enforcement:

1. Hard compliance

With hard compliance, ADSM will not allow certain operations to complete if the server is not in compliance with the current licensing terms. This is manifested in two ways:

- Quantities

  As a certain value approaches a licensing limit (the number of licensed clients for example), the license manager will issue a warning at 80% of the licensed level. At 120% (a 20% grace period) further attempted operations will fail.

- Features

  Certain features of ADSM now require a license before they can be used at all. Attempting to start a space management operation without having the space management license will fail, for example.

2. Soft compliance

   With soft compliance, the ADSM license manager will not fail the operation, but will issue a warning and allow operation to continue normally.

The type of licensing compliance employed depends upon the type of server license purchased, the operating platform and possibly the feature being used. Only AIX server licensing will be covered here. ADSM V1.2.1 and V2 for AIX use a license scheme known as the IBM Licensed Agreement for Programs (ILAP) in the US and International Program License Agreement (IPLA) in EMEA. Licenses can be obtained for the following elements of ADSM:

- Environment

  Under AIX, the V1.2.1 and V2 servers are automatically licensed for one AIX backup/archive client. Licenses for additional environment support features can be purchased that allow the server to support clients on other platforms. The following environment features are supported:

  – UNIX

    Supports all UNIX based ADSM clients including AIX, AT&T UNIX, DEC ULTRIX, HP-UX, SCO UNIX 386, SCO OPEN Desktop, SINIX-Z, SunOS and Sun Solaris.

  – Desktop

    Supports DOS, OS/2, NetWare, Windows and Macintosh clients.

  A further license can also be purchased for the ADSM V2 space management feature. If this license is not purchased, migrate operations will fail, but other client operations, such as recall, will work.

- Device Support

  Licenses can be purchased for four different device support modules. The modules are inclusive, so purchasing module 4 support includes all devices in module 3. Module 3 includes all devices in module 2 and of course module 2 includes devices in module 1. Attempts to define or use a device that is unlicensed will fail. Devices may be shared between servers either through purchasing additional full licenses or by purchasing a Secondary Server Attachment Module which allows a server to share devices with a server that has a full license. The Secondary Server Attachment Module can only be purchased for Version 2.

- Additional user authorization

Additional licenses can be purchased to allow the server to support other clients. Each of the following is considered to be a single client

- Backup/archive
- API
- Space management

Each additional client license is registered for a single client or for 5-,10- or 50-client registration pack.

The licenses themselves can be of two types:

1. Try and buy

   This license is designed for customers who wish to try out ADSM. It licenses a single client connection with space management support and support for all device features. The license expires after 60 days. This license is enforced with hard compliance.

2. Basic license

   This is the standard license type which enforces soft compliance of the ILAP or IPLA licenses described in the previous section for clients running on the same platform type as the server. Hard compliance is enforced for connection from clients running on other platforms (a license must be purchased).

4.1.3.3 Client Workstation Users Access to the Server

By default, any user defined on a registered workstation can request ADSM services.

ADSM provides two options in the client system option file to prevent users from accessing the server. The root user is the only one who has write access to the dsm.sys file where these parameters are specified:

GROUP Use this option to list the groups on the workstation allowed to request ADSM services

USER Use this option to authorize only the specified users to request ADSM services

4.1.3.4 Other Security Issues

Listed here are some issues that have been already described in the previous sections, but which are also related to security administration

- Hierarchical structure of administrative authorities
- Restricted authority privileges
- Client closed registration (with open registration, only the root user can register the workstation)
- File deletion option: the administrator can decide whether the users are allowed to delete backed-up or archived files

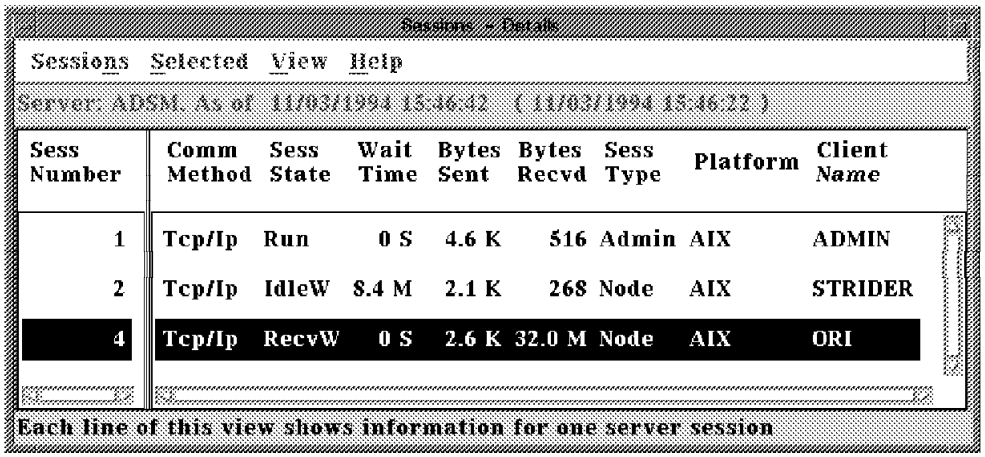

Figure 37. Session Management GUI Display

4.1.4 System Availability

ADSM provides additional features in order to ensure the integrity of the database and the availability of data:

- Database utilities
- Storage pool backup
- Mirroring

4.1.4.1 Database Utilities

The ADSM database is a critical element in the server environment, keeping all the crucial information about accessing the backed up and archived user data. Regular backups of the server database using the new ADSM V2 database backup utility mean that if the server is ever unable to initialize because of a software or hardware error, the new database recovery utility can be used to restore the database. As a last resort, the V1 `DBDUMP` and `DBLOAD` commands can still be used, though a dump cannot be performed online, as previously.

Database Backup/Recovery: The new database backup and recovery feature provided in ADSM V2 is designed to operate without interruption of client service. A consistent copy of the database is made, concurrent with any update activity being performed. The backup can be performed manually by the administrator, triggered automatically by recovery log utilization and scheduled using the ADSM central scheduling utility. The backups are incremental after the first full database backup which reduces the time required to perform the backup. The backups can be routed to any device supported by ADSM and the set of volumes that the backup is stored on are self- consistent; no other information is required to restore the database back to a usable state.

The administrator must balance the trade-offs between running full and incremental database backups. A full backup takes longer to run than an incremental, though recovery time will be faster from a full backup, as only one set of volumes is required to restore the entire database. As many as 32 incremental backups can be taken for each full backup. It is also recommended that backups be scheduled rather than manually initiated, and that additionally, backups be triggered on

recovery log utilization to ensure that the log does not run out of space before the next database backup.

If the database needs to be recovered, the server can optionally save enough recovery log records to ensure that the database can be rolled- forward, by applying the transactions from the records, to bring the database to its most recent transaction-consistent state. The database could also be restored to the exact point in time at which a backup occurred; the granularity of the restore depends upon the frequency with which full and incremental backups are taken. The backup information also contains reference to which volume of a multi-volume database each database page was stored on. Thus individual volumes of multi-volume databases can also be restored, if necessary, though roll forward processing is required to bring the entire database back to a transaction consistent state.

Recovery of the database to a specific point-in-time should normally be used for exceptional situations such as disaster recovery or software errors that have rendered the database inconsistent. Point-in-time recovery is possible even if roll-forward recovery is enabled, though point-in-time can be enabled on its own. The considerations are as follows:

- If only point-in-time recovery is enabled, the recovery log can be defined with a smaller size and mirrored copies of the recovery log therefore consume less disk space.
- Database recovery time is faster as there is no requirement to audit the database after a restore, as compared to using the database salvage utility.
- The database cannot be restored to its most current state, only to the point at which the last backup (full or incremental) occurred.
- Recovery of a single database volume in a multi-volume environment requires restore of the entire database; this is due to the fact that restore of a single volume only, requires roll-forward processing.
- Audit of sequential storage pool volumes based on information in the volume history file and audit of the disk storage pool volumes will be required.
- It is still recommended to use recovery log mirroring if point-in-time recovery is being used.

Roll-forward recovery provides the ability to recover the database to its most current state, if the recovery log is available. With recovery log mirroring, this therefore provides the most comprehensive protection of the ADSM server database. The considerations are as follows:

- Restoration of the database to its most current state protects against loss of client files due to hardware failures on the server.
- No storage pool or database audits are required, so recovery is faster.
- Media failure of a single database volume in a multi-volume environment only requires the restore of that volume, reducing the recovery time.
- A larger recovery log is required as additional log records must be saved. The actual size of the recovery log will depend on the frequency of backups, on whether a database backup trigger is set and on the volume of ADSM transactions processed.

- Mirrored copies of the recovery log will be larger due to the increased size of the recovery log, therefore consuming more disk space. It is recommended that mirroring is performed, if sufficient disk space is available.

Specific information on the setup of these new facilities can be found in the *ADSM/6000 Administrator's Guide*. The following recommendations hold in general for ADSM server database backup and recovery:

- Use roll-forward mode

  If there is enough recovery log space, use roll-forward recovery. With ADSM space management, active client data is now being stored at the server. To restore the database to its most recent state in order to be able to recover this data, roll-forward mode must be used.

  Use the new `RESET LOGCONSUMPTION` command and the new display output on the `QUERY LOG Format=Detail` command to assist in estimating recovery log growth using this mode.

- Complement with database mirroring

  Complement this mechanism with mirroring to provide maximum protection for 24 hour-a-day, 7 day-a-week operations, including protection against media failure.

- Use recovery log mirroring

  If there is sufficient disk space to do so, mirror the recovery log. The recovery log is essential to restore the database to its most recent state; with mirroring, the log is also protected against media failure.

- Define volume history and device class backup files

  These files are used during point-in-time database restores. If they are not available, all storage pools will need to be audited and all device classes, drives and libraries will need to be defined manually.

- Schedule database backups

  Scheduling a full database backup weekly and an incremental backup daily will provide good protection.

- Complement scheduling with automatic backups

  Defining database backup triggers will protect the database from a situation where the recovery log becomes full before a database backup occurs.

- Schedule database backups with storage pool backups

  See 4.1.4.2, “Storage Pool Backup/Recovery” on page 88 for a discussion of storage pool backup. If the storage pool copy facility is being used to take backups of the primary storage pool, taking a backup of the database subsequently will ensure that a database backup that reflects the current storage pool backup exists.

- Store all backup volumes offsite

  It is a sensible precaution to store all backup volumes offsite to avoid losing them in the event of a disaster at the installation site.

Database Dump/Load: Some preparation of the environment is required before the database can be dumped:

1. Define a tape library, the tape drive and the device class as explained in 4.1.1.2, “Optical and Tape Storage Pools” on page 71
2. Update the server option file, dsm.opt, by setting the `DUMPloaddb` option. The following example shows the required settings for an 8mm dump device:

```
dumploaddb library TAPELIB libtype=manual
dumploaddb drive TAPELIB tape1 device=/dev/mt0
dumploaddb devclass TAPE devtype=8mm library=TAPELIB
```

The server must be halted, and from the AIX command prompt, the following command should be submitted:

```
dsmserv dumpdb devclass=tape volumename=tpvol1
```

To restore a dumped ADSM database to a newly installed ADSM server, the following offline command can be used:

```
dsmserv loaddb devclass=tape volumename=tpvol1
```

In this case, the database, recovery log and storage pool files must reside in the same directory they were in before the dump operation.

To ensure that the loaded database is returned to a consistent state, the following offline command can be executed before restarting the server:

```
dsmserv auditdb fix=yes
```

The `fix` parameter causes any inconsistencies discovered to be repaired.

Some examples of the usage of these utilities can be found in *Getting Started with ADSM/6000*.

4.1.4.2 Storage Pool Backup/Recovery

ADSM Version 2 provides a new facility that allows for the backup and recovery of storage pools. It is designed to allow incremental backups of primary storage pools to be taken concurrently with server operation. Storage pool backup is fast since the incremental backup only needs to copy those files that are not in the copy pool or have changed since the last backup. The backups can be scheduled using the ADSM central scheduling facility and individual volumes of a storage pool can be restored in the event of media failures in a storage pool. The copy storage pools used for the backup data can be defined on removable media for offsite storage and ADSM provides a management mechanism to track the location and currency of copy storage pools and the files within them. Should the primary copy of a file become inaccessible for any reason, ADSM will automatically switch to a copy, if

one is available. Finally, storage pool backup and database backup (discussed in “Database Backup/Recovery” on page 85) can work cooperatively to make it possible to restore both database and storage pools in the event of a disaster.

Copy storage pools to which primary storage pool data is backed up, are sequential storage pools. They cannot form part of a hierarchy, nor can client data be directly stored in them. Backup to a copy storage pool is performed asynchronously and therefore does not affect client operations. Granularity is to the individual file level, so a single damaged file can be restored if required.

Specific information on the setup and use of these new facilities can be found in the *ADSM/6000 Administrator's Guide*.

Recommendations on the usage of the new storage pool backup facility are as follows:

- Backup the entire primary storage pool hierarchy to the same copy pool(s).

  ADSM will recognize the fact that a file has migrated down a storage pool hierarchy and will not back it up again to a copy storage pool. This works if the environment has been defined such that each primary storage pool hierarchy is copied to the same copy pool(s).

- Consider using at least one copy pool for onsite recovery and one for disaster recovery.

  ADSM will automatically switch to a backup copy of a file, if such a copy exists onsite. It is therefore a good idea to maintain an onsite and an offsite copy pool; the offsite pool protecting against complete disaster.

- Use administrative command scheduling

  With ADSM Version 2, the central scheduler can be used to automate storage pool backups.

- Backup database and storage pool together

  As already discussed in “Database Backup/Recovery” on page 85, it is a sensible idea to backup the server database subsequent to a storage pool backup.

- Consider running `AUDIT VOLUME` to periodically mark damaged files.

  The `AUDIT VOLUME` command can be used to synchronize the database with the storage pools. Those files in a storage pool with no record in the database are marked as damaged which helps in their restoration the next time a storage pool is restored.

- Run `RESTORE STGPOOL` periodically to replace damaged files

  The `RESTORE STGPOOL` command will cause any damaged files in primary storage pools to be replaced by an undamaged version. This will speed up the access process if a user subsequently requires such a file.

4.1.4.3 Mirroring

To ensure the availability of the database and recovery log, the system administrator can use the mirroring facility of ADSM. Using this feature, up to three copies can be defined for each volume dedicated to the database and recovery log. By separating the mirrored copies on different physical devices, the server is protected against media failure and throughput, and availability of data is increased.

The ADSM implementation of mirroring is conceptually the same as is used by RAID 1 hardware implementations, such as for the 3990 dual copy.

An I/O operation is completed only when all volumes have been updated. ADSM allows control of the database and recovery log processing and performance, using the options: `MIRRORREAD` and `MIRRORWRITE` in the server options file. `MIRRORREAD` has two modes:

normal	Reads only one mirrored volume for each page read
verify	Reads all mirrored volumes for each page read

`MIRRORWRITE` has two modes:

sequential	Writes mirrored volumes in succession
parallel	Writes mirrored volumes at the same time

Mirroring allows the server to automatically handle most hardware problems. For example, if the device on which the database is allocated fails, the server will simply put the volume offline and carry on with the mirrored copies. When the problem has been resolved, the repaired volume can be dynamically resynchronized as a mirror copy.

The administrator GUI can be used to manage volume copies. If the copies are to be defined from the command line, issue the following commands:

```
define dbcopy vol1 volc1
```

To query the server for information about mirrored volumes, enter:

```
query dbvol
```

The status of mirrored volumes can be:

syncd	Synchronized
stale	Not available, as in not yet synchronized
offline	Not available, as in not operative
undefined	No volume defined as a mirrored copy

For additional information about mirroring, see the *ADSM/6000 Administrator's Guide*.

4.2 File Storage Facility/6000

FSF administrator tasks must be performed using the SMIT FSF menu options, by the root user at the client workstation.

```
                              File Storage Facility

  Move cursor to desired item and press Enter

  Add an FSF File System
  Remove an FSF File System
  Mount an FSF File System
  Unmount an FSF File System
  Perform FSF Maintenance
  Manage FSF Files
  Manage Daemons


F1=Help             F2=Refresh            F3=Cancel             F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 38. SMIT FSF Menu Display

4.2.1 System Administration

In this section, the administrator tasks related to FSF system management will be examined in more detail.

4.2.1.1 Device Management

It has already been mentioned in 3.2.3, “Customization” on page 40, that FSF requires four different volumes to manage file systems:

1. Remote server store
2. Local cache
3. Local FSF Manager logical volume
4. Local FSF Log logical volume

More than one FSF file system can be defined at the local workstation, each one managing a different local file system. This can be effected simply through selecting the **Add an FSF file system** menu option and following the steps described in the previous chapter.

It's important to know that for each FSF file system a different remote server store must be defined on the remote server. This is done to avoid a situation where two configurations share the same NFS file system which creates a potential exposure in terms of data loss or corruption.

Because of the internal design, FSF also requires two different logical volumes (Manager and Log) for any new file system it has to manage. The default size for them is 4MB each; so space at the workstation may also become an issue.

4.2.1.2 File System Management

The root user can perform the following operations on FSF file systems:

- Add
- Mount
- Unmount
- Update
- Remove
- Verify

The first two have already been discussed in the previous chapter.

Unmount and Remove an FSF File System: Before removing an FSF file system, it must first be unmounted. From the SMIT FSF main menu, select the **Unmount FSF File System** panel, insert the file system mount point and press Enter. Wait for the SMIT OK command status display:

```
FSF file system "/u/marina" is unmounted.
```

Now the FSF file system can be removed. From the SMIT FSF menu, select the **Remove FSF File System** panel, and insert the required values in the entry fields.

```
                        Remove an FSF File System

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                      [Entry Fields]
  Mount Point                                        /u/marina           +
  Remove Mount Point?                                no                  +
* Remove JFS cache file system?                      no                  +


F1=Help             F2=Refresh            F3=Cancel            F4=List
F5=Reset            F6=Command            F7=Edit              F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 39. SMIT FSF Remove File System Panel

If `Remove JFS cache` is set to `no`, FSF removes only its data structure. If it is set to `yes`, FSF also removes the local JFS file system (which was the client cache). Notice that the server store (either NFS file system or ADSM filespace) cannot be removed by FSF.

Update an FSF File System: From the SMIT FSF main menu, select **Perform FSF Maintenance**, and from here the **Change FSF File System** menu. The mount point of an FSF file system, the size of the manager logical volume, the cache size and the mount options may be dynamically changed.

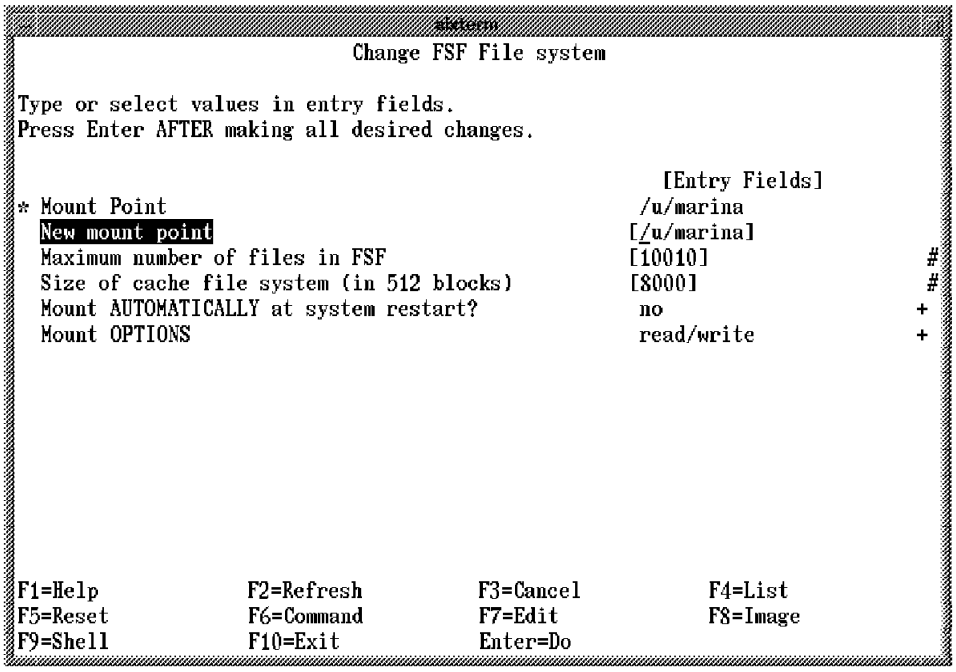

Figure 40. Change FSF File System Display

Verify an FSF File System

From the SMIT FSF main menu, select **Perform FSF Maintenance**, and from here the **Verify FSF File System** panel. FSF provides two different file system verification checks:

fast=yes Verifies that the FSF file systems data structures are internally consistent.

fast=no Verifies that the FSF file systems data structures are internally consistent and consistent with the client cache. It does not check the FSF data structures against the internal store.

After selecting the fast check, press Enter; the system will ask whether it should correct files not marked clean. The output is shown in the following figure:

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instruction may appear below.

File system not marked clean Correct? [yn]
1024 blocks, 984 free, 40 used (2 map, 0 symlink, 1 dir, 2 mnode, 0acl)
15756 (12) mnodes free, 20 used (17 file, 0 symlink, 1 dir, 1 acl)


F1=Help             F2=Refresh            F3=Cancel             F6=Command
F8=Image            F9=Shell              F10=Exit
```

Figure 41. The Verify FSF File System Output

4.2.1.3 Daemon Management

FSF daemons can be managed by selecting the **Manage Daemons** menu from the `smit fsf` main menu. There are options to start and stop both the NFS and the ADSM FSF daemons.

Only one daemon can be running in the client workstation at a time. Even if a second daemon can be successfully started from the **Manage Daemons** panel, it will affect the behavior of the first one, and the results may be unpredictable.

All the error messages from the FSF daemons are directed to the file /tmp/fsf_dmn.err.log. This log contains entries indicating when the daemon has encountered an error synchronizing to, or caching from, the server store. The destination of this log can be changed to any valid file by simply updating the related entry in the /etc/syslog.conf file.

If the log shows that an error has occurred, verify that the connection to the server is up. If the connection is down, FSF cannot cache files that have been pruned or synchronize files which have been modified (for a detailed explanation of these operations, please refer to 4.2.2.2, “Tuning Automatic FSF Operations” on page 95). If problems persist after reestablishing the connection to the server, refresh the FSF daemon by stopping and starting it again from the **Manage Daemons** SMIT panel.

In order to clear the /tmp/fsf_dmn.err.log file, the following actions need to be performed:

- Make a backup copy of the log (`cp /tmp/fsf_dmn.err.log /tmp/fsf_dmn.err.log.bak`)
- Compress the copy (`compress /tmp/fsf_dmn.err.log.bak`)
- Clear the contents of the file (`rm /tmp/fsf_dmn.err.log; touch /tmp/fsf_dmn.err.log`)
- Refresh the log daemon (`refresh -s syslogd`)

4.2.2 User Administration

If the workstation is single user, there probably isn't a real necessity to have multiple FSF file systems defined. If the workstation is shared by multiple users, two possible scenarios can be defined:

1. There are not many users defined and/or the whole system is to be set up. In this case, a good solution could be to define one empty FSF file system with the automatic mount option set to `yes` and all of the new user's file systems defined in it.
2. The environment has already been set up, and multiple users are working on their file systems. In this case, it is possible to mount the FSF cache directory on the /u directory, putting all user's files under FSF management.

   Remember that in any case, any change to the underlying file system (for example, increasing the space of the AIX file system) must only be performed when the FSF file system is unmounted in order to prevent any possible mismatch between the AIX and the FSF mount points.

FSF allows the root administrator to manage user's files by establishing the parameters that control the timing of the automatic FSF operations, and by manually performing the required functions.

4.2.2.1 Manually Managing Users Files

From the SMIT FSF menu, select **Manage FSF fFles**. The menu presented allows the following tasks to be performed:

Pinning Files A file may be *pinned* so that it remains unmodified in the client cache and is never *pruned*. The file will be backed-up to the central store as part of the normal write-back operation, whether or not it is pinned.

Unpinning Files A previously pinned file may be *unpinned* so that it will be pruned as part of the normal pruning operation. Also, the owner of the file can manually perform the unpin operation.

Pruning Files FSF periodically purges local copies of files that are not being used once they have been written back to server store. There may be a requirement to perform this action manually, and any user with write access to the file can manually prune it from the local store.

4.2.2.2 Tuning Automatic FSF Operations

The FSF system parameters can be modified in order to set the best timing options for an environment and to optimize FSF performance for each client at a site. Parameters may also need to be adjusted periodically as a user's requirements change.

When establishing the frequency of the pruning operation, the following facts should be considered:

- FSF must backup a file (write-back operation) before it can be pruned
- Frequent pruning operations maintain free cache space for client users
- High frequency of write-back operations can degrade network response time
- Pruning operations compete with user access to the cache; high frequency of pruning can also degrade user response time

In order to update the FSF tuning parameters, from the SMIT FSF main menu, select **Perform FSF Maintenance** and then the **Fine-Tune FSF Operations** panel (see Figure 42 on page 96).

LOBLK polling The period in seconds at which FSF checks the client cache for the high water mark

High water mark As the client cache fills up, the used disk space approaches the high water mark. This is the percentage of used disk space for the client cache that will trigger a pruning operation

Low water mark The percentage of used disk space for the client cache that stops a pruning operation

Minimum file age The minimum amount of time that a file can remain in the client cache before it will be copied to the server store

Prune on basis of The pruning priority can be set to the *usage* of a file or to its *size*

Maximum time How long a file can remain modified in the client cache before being written back to the server store. This parameter allows the synchronization between file copies to be managed

Max file num The maximum number of out-of-sync files that can exist in the local cache. When this number is reached, FSF starts an automatic write-back operation for these files, regardless of how long they have been out of sync

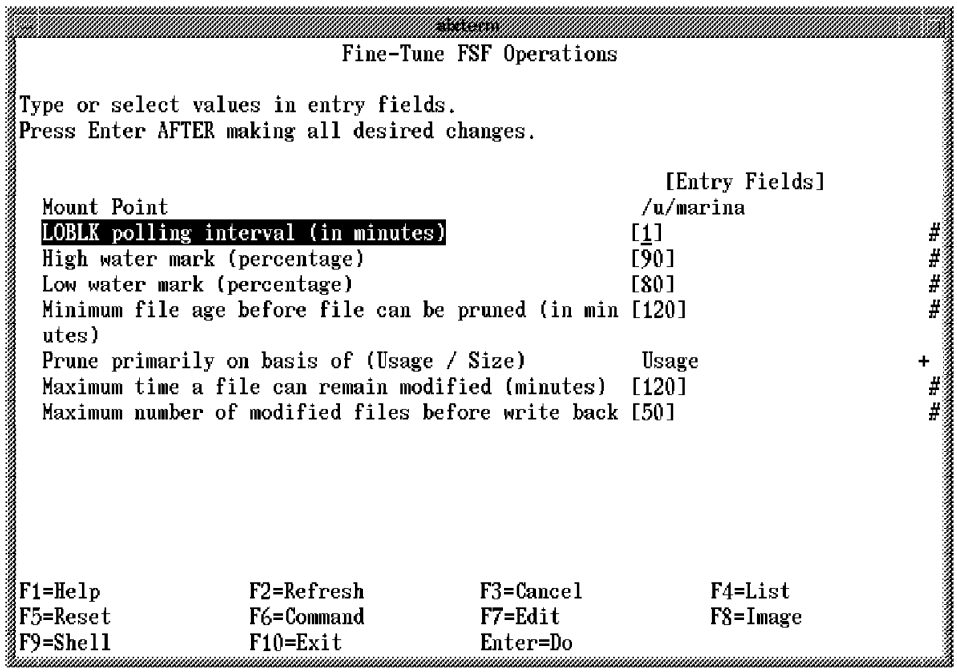

Figure 42. Fine-Tune FSF Operations Display

4.2.3 Security Administration

There are no particular FSF management tasks solely related to security administration. The intention here is to summarize some of the previously discussed administrator tasks that can affect the security of the environment.

The most important of these is probably the setting of file access permissions.

4.2.3.1 Root Users

It should be remembered that root users act as FSF administrators, but they cannot create files in the local cache when they are logged on as root. It is highly recommended to set up the server store to NFS without root access.

4.2.3.2 Normal Users

The FSF administrator is responsible for setting up the file access permissions for the client cache directory in the new FSF file system. In this way, any users on the same workstation can work on their own files and only on them. It should be remembered that FSF manual commands such as `unpin` and `prune` can be issued by any user having write access to the file.

4.3 UniTree

UniTree does not provide a GUI. The administrative tasks are performed by using the SMIT interface and issuing commands from the server AIX prompt.

4.3.1 Device Administration

The server root user can manage UniTree devices using UniTree line commands and change the disk server and tape server configuration using the SMIT interface.

4.3.1.1 Managing the Disk Cache

Information about the disk cache logical volumes can be displayed using the command `statup`:

```
# statup
Fri Nov 25 13:04:47 1994
_____________________________________________________________________
LUnit   PUnit   Stat   H Cpac   D Cpac   Err Cnt  L Err
1       1       UP     714      40960    0        0
2       2       UP     714      40960    0        0
_____________________________________________________________________
#
```

Figure 43. UniTree statup Command Output

Stat The status of disk devices (UP, DOWN, GOING DOWN)
H Cpac The maximum number of headers on the disk volume
D Cpac The size in K of the disk volume
Err Ct The number of disk errors
L Err The number of lincs errors

To display information on the current utilization of the disk cache, the command `statall` can be issued:

```
# statall
Fri Nov 25 12:58:19 1994
_____________________________________________________________________
LUnit  PUnit    H Use       D Use         Hole    Frags  Err Ct  L Err
    1      1        5         672        40228        1       0      0
    2      2        5         672        40228        1       0      0

HeaderUse = 0%          DiskUse = 1%              TotalFiles = 10
_____________________________________________________________________

RcAny   Req     T Req   T Err
8       9       1433    1

File Create Count          = 3
Create Time of Oldest File = Tue Nov 22 13:38:14 1994
_____________________________________________________________________
#
```

Figure 44. UniTree statall Command Output

H use The number of used headers
D use The actual disk usage in K

Hole	The largest free space in K on the cache volume
Frags	The number of fragments
Err Ct	The number of disk errors
L Err	The number of lincs errors
RcAny	The number of available tasks to receive client requests for the disk server
Req	The total number of tasks in disk server
T Req	The total number of requests since disk server startup
T Err	The total number of errors since disk server startup
File Create count	The number of files created after the disk server startup

The `forcepurge` command can be used to force an immediate purge of the disk cache. All the files that have a migrated copy on tape or optical devices are deleted from the cache to free space. The purge operation is performed automatically according to the parameters specified by the system administrator when configuring the migration server. More detailed information is available in 4.3.3, “Tuning” on page 103.

Depending on the occupancy of the disk cache, disk server volumes may need to be added or removed.

Add a New Disk Server Volume: To add a disk volume, use the `smit UniTree` interface as shown in 3.3.3.1, “Minimum Configuration Dialog” on page 49, to cause UniTree to allocate a new volume to be used by the disk server. In order to preallocate the new volume manually, use the following procedure:

1. Define a new logical volume
2. Execute the `zerodisk` command to initialize the volume for UniTree
3. Execute the `disklbl` command to format the volume into UniTree format
4. Run the SMIT UniTree interface, select **Add Disk Server Logical Volume** and enter `yes` in the field `Logical volume already exists?`

Remove a Disk Server Volume: To remove a currently used disk logical volume from the actual configuration, the following steps should be performed:

1. Run the `forcemig -all` command to force migration of all the files in the cache to the second level device (tape or optical)
2. Check in the log that migration is completed successfully
3. Stop the disk server component of UniTree, killing the processes in the following order:
   a. The udiskd daemon
   b. The diskmvr process
   c. The disksrv process
4. Run the `diskdown` command to shut down the logical volume. This utility copies the contents of the volume to other disk server volumes. If there is not enough space on the other volumes, it purges all the migrated files
5. Edit the /usr/lpp/UniTree/erc/cfsdev file and remove the entry relating to the removed volume
6. Restart the disk server by entering `udiskd`

A UniTree disk volume that has been previously disabled with the `diskdown` command may be reinitialized by issuing the command `diskup`.

Forcing File Purge: The system administrator can force an immediate purge of the disk cache by issuing the command `forcepurge`. The purging operation is also automatically performed by the disk server according to the parameters defined in the SMIT UniTree View/Change Disk Server panel. Refer to 4.3.3, “Tuning” on page 103 for more detailed information.

4.3.1.2 Managing Tapes and Optical Devices

The status of all tapes and optical devices can be displayed with the command `readmap`:

```
# readmap /dev/rSearch_Table_Logvol_name
Using automatic settings...
Tape map starts on sector 523
Start tape label is AA0000
AA0000 TAPE IN USE
AA0001 TAPE IN USE
AA0002 count = 147
AA0003 count = 137
AA0004 count = 0
AB0239 TAPE UNAVAILABLE
#
```

Figure 45. UniTree readmap Command Output

For each device, the utility shows the current status, which can be:

TAPE IN USE UniTree is currently using the volume

TAPE UNAVAILABLE
The volume has been marked unavailable

count = 0 The volume is ready and empty

count = n The volume is ready with n blocks already used

The status of a device can be set with the command `setblk`:

```
setblk device_label n
```

where `n` can have the following values:

-1 Sets the volume status to UNAVAILABLE. This option is typically used when a tape needs to be removed from the UniTree configuration

0 Set the volume status to EMPTY (0 used blocks)

n With n > 0. If the number of blocks previously used is specified, this command sets the volume to be available with n used blocks

The currently used devices can be identified with the command `wtlog`:

```
# wtlog /dev/rst1_prim
Using automatic settings...
Tape Write sector is on sector 522
Start tape label is AA0000
Index           Tape ID         Block Count
-----           -------         -----------
   1             AA0000               157
   2             AA0001               157
   3             AA0002               147
   4             AA0003               137
  99             BCK000                 0
#
```

Figure 46. UniTree wtlog Command Output

To monitor the actual occupancy of the tape server header space, the command `tdspace` can be used:

```
# tdspace -f
Number of Available Disks: 4
Number of Header Disks: 1 (logical)  2 (physical)
Number of Search Disks: 1 (logical)  2 (physical)
Logical Header Disk 0:
        Tddevs disks 1 and 2.
        Site is 4096 4k blocks.
        Scanning allocation bitmap and headers, this may take a while...
        Header start at sector 2, Maximum index 126
        1 header sector filled
        4063 headers sectors open
        18 headers in use
        44686 headers available
        44704 headers total, 99% free
#
```

Figure 47. UniTree tdspace Command Output

Depending on the usage of the tape server header space, it may be necessary to add a tape server header logical volume pair. This task can be performed using the SMIT UniTree interface as shown in 3.3.3.1, “Minimum Configuration Dialog” on page 49.

It is also possible to change some configuration parameters for the tape server logical volumes. This task will be detailed in 4.3.3, “Tuning” on page 103.

Peripheral Devices: The SMIT UniTree interface can be used to add and remove UniTree peripheral devices. In order to label single tapes or optical devices, the following utilities may be used:

tapelabel	Labels a single tape for UniTree
readlabel	Reads and displays the tape label
opt_label	Labels a single optical device for UniTree
opt_read_label	Reads and displays the optical device label

If adding tape or optical libraries, it may be easier to handle, initialize and label the tapes and optical disks in the library at the same time. UniTree provides some utilities which are specific to the different UniTree supported devices. These utilities

are listed here for completeness; refer to *UniTree System Administrator's Guide* for a complete description.

Most of these utilities must not be executed when the UniTree system is running.

The following are not related to specific device types:

optmapcheck Check the correctness of the slot map file
optslotcheck Prints a map of the slots and drives in the library

The following utilities are device specific:

- Alphatronix Inspire Rewritable Optical Jukebox

  alphalabel To label Alphatronix optical disks

- Comtec ATL-8 Model 54

  caslabel To label all tapes in the carousel

- DocuStore DISC jukebox

  discutil Moves optical disks in the jukebox (load, mount and unmount)
  disclabel To label DISC jukebox optical disks

- EXABYTE EXB 10i/10e

  chmmvelem Moves cartridges from one slot to another
  chmpos Positions the media changer arm in front of a slot
  uld Ejects a tape
  chmlabel Labels tapes in the library

- EXABYTE EXB 120 CHS

  chseep Extends and retracts the entry/exit port in the tape library
  chsinitelem Scans the barcode labels and creates the slot map
  chslabel Labels tapes in the library according to the barcode labels
  chsmodesense Obtains mode sense data from the media changer
  chsmvelem Moves a tape in the library from one slot to another
  chspos Positions the media changer arm in front of a slot
  chsprevent Locks and unlocks the entry/exit port of the library
  chsscroll Display messages in scrolling format
  chssteady Display messages in steady format
  chsgetaddr Retrieves the slot number of a specified volume label
  chsrdelem Reads the volume label of a tape

- HP Optical Disk Library System

  hputil Moves optical disks in the library (mount and unmount)
  hplabel Labels the optical disks in the library

- IBM 3995 Optical Jukebox

  ibmutil Moves optical disks in the library (load, mount and unmount)
  ibmlabel Labels the optical disks in the library

- IGM ATL 8mm

  igmutil Moves tapes in the library (load and unload)
  igmlabel Labels the tapes in the library

- StorageTek Automated Cartridge System

  stk_label Labels single tapes in the library

stk_multi_label Labels groups of tapes in the library

The Repack Utility: Release 1.2 of UniTree provides a new repack feature. The system administrator should repack tapes when they have become fragmented due to file deletion. The fragmented data is moved to other free tapes. This operation frees up the tapes where the files previously resided.

The files on the tape are first staged to the disk cache and then migrated to the new tapes. This operation affects the performance of the system, consuming extensive system resources. This function is automatically performed by the repack server according to the parameters specified in the SMIT UniTree View/Change Repack Server panel. A detailed description of the repack parameters can be found in 4.3.3, “Tuning” on page 103.

The repack operation can be forced at any time by issuing the command `repack`.

4.3.2 User Administration

The system administrator can add and remove users and also change some user parameters. The process for adding a new user to UniTree has already been described in 3.3.3.5, “Defining Users to UniTree” on page 55. When a new user is defined, remember that the uid must be the same on the server and on all the clients requiring to access UniTree system.

To remove a user from the UniTree system, simply select **Delete UniTree Users** as shown in Figure 48 on page 103. The only parameters that may be changed in a user definition are:

- Permission: the user's access permissions to their UniTree home directory
- Trash can time-out: the time a file can remain in the trash can before being deleted

The Trash Can Directory: When a user is defined to UniTree, a directory named .trash is created in the UniTree user's home directory. Any time a user deletes a file from the cache, a copy of the file is stored in the .trash directory for recovery purposes. UniTree renames the files adding a date and a counter. You cannot rename files in the .trash directory.

After the time specified with the trash can time-out parameter, files are permanently deleted from the .trash directory. Users can override the system administrator setting for the trash can time-out.

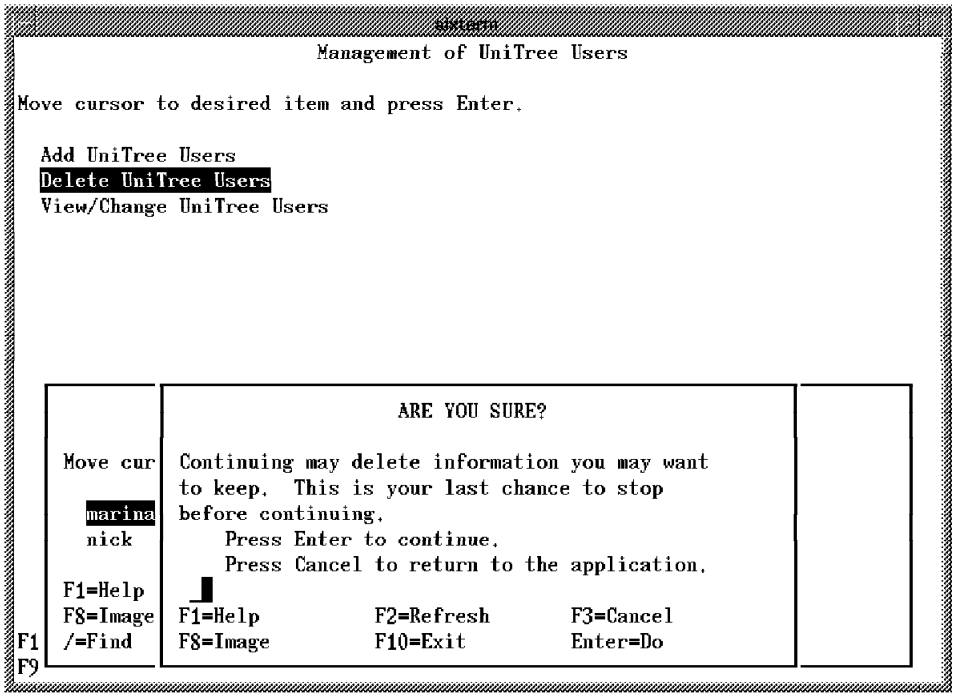

Figure 48. SMIT UniTree Delete User Display

Camping Files: Users can set a flag for their files so that they are never purged from the disk cache. This operation is called *camping*. Camping does not affect file migration. If the availability of disk cache space is critical at a particular installation, the system administrator can decide to prevent users from using this command. This can be done by setting the owner of the `camp` executable to root and the access permission to 0700.

4.3.3 Tuning

The UniTree system administrator can use the SMIT UniTree interface to tune the parameters that control the servers automatic operations:

- Migration
- Purging
- Repacking

From the SMIT UniTree interface, follow the path:

```
Configure UniTree
Management of UniTree objects
View/Change UniTree objects
```

4.3.3.1 Tuning File Migration

Migration is the process of copying files from the disk cache to the second-level storage devices (tapes or optical). It can be done manually when the administrator issues the command `forcemig -all` or automatically according to the administrator specifications.

The disk cache is not mirrored; migration is the only way of providing backup for user files. At the same time, migration is a resource-consuming process; so migration parameters should be carefully set. Migration does **not** include file directories.

Select the **View/Change Migration Server Parameters** configuration panel:

```
                    View/Change Migration Server Parameters

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Debug enabled                                       yes                      +
* Minimum file age in minutes                        [3]                       #
* File number threshold                              [100]                     #
* Number of minutes before migration occurs          [60]                      #
* Number of minutes the migration sleeps             [3]                       #


F1=Help             F2=Refresh            F3=Cancel            F4=List
F5=Reset            F6=Command            F7=Edit              F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 49. SMIT UniTree Change Migration Server Parameters Display

Minimum file age	The minimum file age before a file is eligible for migration
File number threshold	The number of new or modified files necessary to initiate the migration process
Minutes before migration	The maximum time before the migration process starts
Minutes of migration sleep	The interval between migration server checks of the disk cache conditions

Migration starts when the server wakes up (after the migration sleep) and finds either a number of changed files greater than the `File number threshold` value, or that the time from the last migration is greater than the `Number of minutes before migration occurs` parameter. Only those files older than the `Minimum file age` are migrated.

The status of a file is considered *changed* by UniTree even if it has been accessed in read-only mode.

In order to change the number of migration copies, the corresponding parameter in the tape server definition can be changed by selecting the **View/Change Tape Server Parameters** panel. As already explained in 3.3.3.1, “Minimum Configuration Dialog” on page 49, a default value may be entered here and a maximum value for the number of migration copies. Users can override the default value, but they are not allowed to enter a value greater than the maximum specified.

4.3.3.2 Tuning File Purging

File purging is the process of freeing space in the disk cache by removing files that have already been copied to the archive storage. It can be performed manually by the root user on the server (the system administrator), by issuing the command `forcepurge`, or automatically, according to the parameters set up by the administrator.

Users can prevent their files from been purged using the `camp` command.

Select the **View/Change Disk Server Parameters** configuration panel:

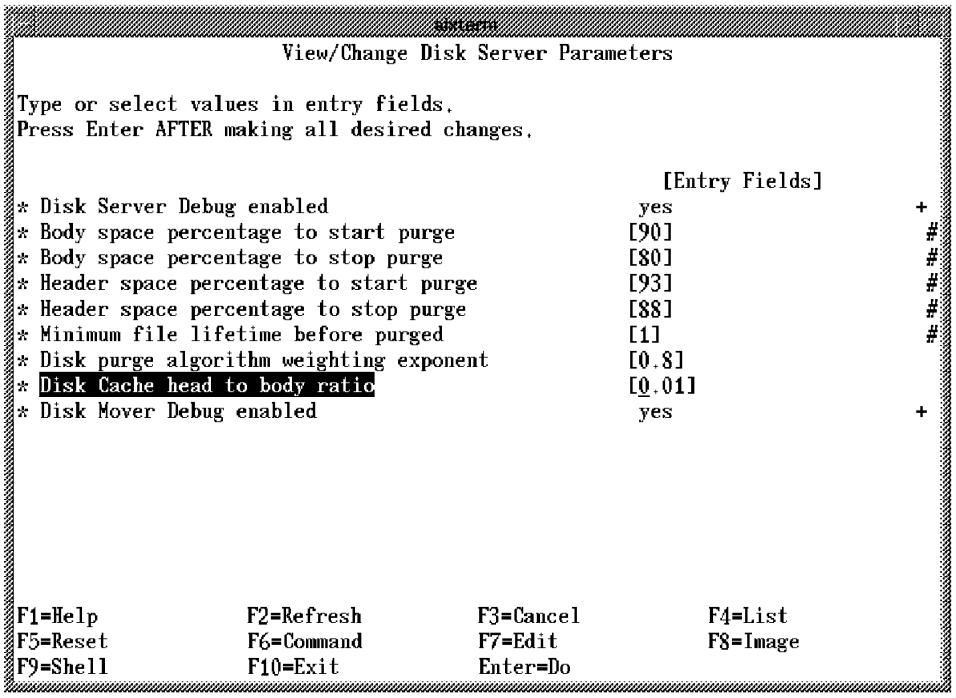

Figure 50. SMIT UniTree Change Disk Server Parameters Display

Body space percentage to start	The disk cache maximum usage before starting a purge
Body space percentage to stop	The disk cache minimum usage before stopping a purge
Header space percentage to start	The disk cache maximum usage for file headers before starting a purge
Header space percentage to stop	The disk cache minimum usage for file headers before stopping a purge
Minimum file lifetime	The minimum file age before it is eligible for purging
Disk purge algorithm	A factor for calculating the purging priority of a file. A value less than one will purge primarily on size; a value greater than two will purge primarily on time, using a least recently used algorithm

Cache head to body ratio

The default value of 0.1% is based on the size of a file header compared with the minimum file size occupancy. It cannot be set to a value greater than 0.5 (50%). In order to change this value, the following procedure must be performed:

1. Issue `forcemig -all` to force migration from cache to second level storage
2. For each disk cache volume issue the `diskdown` command
3. Re-label each volume with the command `disklbl -disk`. This utility enforces the new header to body ratio value
4. Issue the `diskup` command to reinitialize and put online the cache volume(s)

4.3.3.3 Tuning Repack Operation

This is a new feature of UniTree Release 2.0, providing the ability to free and reuse empty space on archive media. Repacking can be performed manually by issuing the command `forcepack`, or automatically, according to the parameters specified by the system administrator.

The repack process copies all files from the second-level storage to the disk cache and from the cache to the new archive media. For this reason, care should be taken in the setting of these parameters.

Select the **View/Change Repack Server Parameters** panel:

```
                    View/Change Repack Server Parameters

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Debug enabled                                       yes                      +
* Number of Disk Files                               [360]                     #
* Amount of disk space to use (MB)                   [150]                     #
* Disk space free before waking stage (%)            [30]                      #
* Disk space full before waking migrate (%)          [70]                      #
* Minimum unacknowledged files during repack         [25]                      #


F1=Help             F2=Refresh            F3=Cancel             F4=List
F5=Reset            F6=Command            F7=Edit               F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 51. SMIT UniTree Change Repack Server Parameters Display

Number of Disk Files

The number of files to copy (stage) from archive storage to the disk cache. If this value is increased, the repack process is speeded up, but cache occupancy is also increased

Amount of disk space

The MB of disk space to dedicate to the repack operation

Disk free before stage The percentage of `Amount of disk space to use` and of `Number of Disk Files` that must exist before staging more files on disk cache.

For example, in Figure 51 on page 106, the staging process can continue only if they are more than 108MB free in the cache, or less than 45 files currently staged

Disk full before migrate The percentage of `Amount of disk space to use` and of `Number of Disk Files` that must be exceeded before file migration can occur.

For example, in Figure 51 on page 106, the migration process can start when more than 252MB has been consumed in the cache or more than 105 files are staged.

Min unacknowledged files The number of files to be written on archive media before putting a filemark. If the system goes down before writing the tapemark, all files since the last one are lost

4.3.3.4 Monitoring System Activity with the Log Files

UniTree produces log files that help in monitoring system activity. All logs are in /usr/lpp/UniTree/adm/log.

UniTree automatically generates the following logs:

udiskd.t The log for the disk subsystem manager

umigd.t The log for the migration server manager

unamed.t The log for the name server manager

utaped.t The log for the tape subsystem manager

```
11/22/94 14:32:09  Starting tape processes:
11/22/94 14:32:09    /usr/lpp/UniTree/bin/fake_el
11/22/94 14:32:09    /usr/lpp/UniTree/bin/ssi.ksh
11/22/94 14:32:09    /usr/lpp/UniTree/bin/tapemovr
11/22/94 14:32:09    /usr/lpp/UniTree/bin/pvrsrvr
11/22/94 14:32:15    /usr/lpp/UniTree/bin/tapesrvr
11/22/94 14:32:37    /usr/lpp/UniTree/bin/pdmsrvr
11/22/94 15:52:08  Received signal 15
11/22/94 15:52:08  Killing all other tape processes
11/22/94 15:52:08  sent signal 9 to SSI pg 9541
11/22/94 15:52:08  sent signal 9 to FAKE_EL pg 13636
11/22/94 15:52:08  sent signal 9 to MOVR_PID pg 13383
11/22/94 15:52:08  sent signal 9 to PVR_PID pg 8777
11/22/94 15:52:08  sent signal 9 to SRVR_PID pg 13139
11/22/94 15:52:08  sent signal 9 to PDM_PID pg 10837
11/22/94 15:52:08  utaped exiting
```

Figure 52. UniTree utaped.t Log File

All the other logs are generated when the system administrator sets the debug parameter to `yes`.

All log files can be cleared at startup with the `clearlogs` option.

The complete list of UniTree log files, together with some example of the information they contain, is shown next:

backup.t The log for the backup utility process

disklabel.t The log for the disklbl utility process

```
Disk /dev/rds1 is to be labeled.
Disk /dev/rds1 labeled properly with 102 header sectors.
Disk /dev/rds2 is to be labeled.
Disk /dev/rds2 labeled properly with 102 header sectors.
```

Figure 53. UniTree disklabel.t Log File

diskmovr.t The log for the disk mover daemon

disksrvr.t The log for the disk server daemon

migsrvr.t The log for the migration server process

```
=== MIGSERVER READING UniTree.conf FOR NEW RUN ===========
11/22/94 17:06:29  =============== MIGRATION SERVER NEW RUN (NLS) ===============
Debug is turned off: 0
Number of files to start migration   set at 100
Minutes before forced backup         set at 60
Minutes to skip migration      set at 3
Minutes for server wakeup      set at 3
11/22/94 17:07:01  Migrate Server Is Up And Running

11/22/94 17:08:08  SLEEP waiting for more files.

11/22/94 17:12:54  Mig Sleep numfiles = 0,numdisks = 2
11/22/94 17:15:55  SLEEP waiting for more files.

11/22/94 17:15:55  Mig Sleep numfiles = 6,numdisks = 2
11/22/94 17:16:43  Called by forcemig;  ForcemigTimeLastWrite = 0
11/22/94 17:16:43  Rumbling through the headers.
11/22/94 17:16:44  Mig build numfiles = 6,numdisks = 2
11/22/94 17:16:45  0) Migrated size = 2,cap = 0x2ed248a6 0xffc70000
11/22/94 17:16:45  1) Migrated size = 5,cap = 0x2ed2490e 0x9e3d0000
11/22/94 17:18:45  2) Migrated size = 3,cap = 0x2ed24906 0xfb580000
11/22/94 17:20:55  3) Migrated size = 3,cap = 0x2ed27ba0 0xeaff0000
11/22/94 17:20:55  4) Migrated size = 2,cap = 0x2ed27b99 0x23920000
11/22/94 17:20:56  5) Migrated size = 5,cap = 0x2ed27ba6 0x46f00000
11/22/94 17:20:56  SLEEP waiting for more files.
```

Figure 54. UniTree migsrvr.t Log File

namesrvr.t The log for the name server process

```
11/22/94 17:06:29  === NAMESRVR READING UniTree.conf FOR NEW RUN ===
11/22/94 17:06:29  =============== NAME SERVER NEW RUN (NLS) ===============
11/22/94 17:06:29  Debug is disabled.
11/22/94 17:06:29  Logging is disabled.
11/22/94 17:06:29  Shiva's default expiration time is 1200 seconds (20 minutes).
11/22/94 17:06:29  Shiva wakes up every 600 seconds (10 minutes) to check cans.
*********  New Run    Tue Nov 22 17:06:48 1994  *********
11/25/94 12:28:40  SigHUP received...
11/25/94 12:28:45  ParmReader working...
11/25/94 12:28:45  Rereading configuration parameters...
11/25/94 12:28:45  Read new values:
11/25/94 12:28:45  Debug is enabled (1).
11/25/94 12:28:45  Logging is enabled (1).
11/25/94 12:28:45  Shiva's default expiration time is 1200 seconds (20 minutes).
11/25/94 12:28:45  Shiva wakes up every 600 seconds (10 minutes) to check cans.
11/25/94 12:28:45  Configuration parameters reread.
```

Figure 55. UniTree namesrvr.t Log File

pdmsrvr.t The log for the PDM server daemon

pvrsrvr.t The log for the PVR server process

```
11/22/94 17:08:03  =============== PVR SERVER NEW RUN (NLS) ===============
11/22/94 17:08:03  Rereading configuration parameters...
11/22/94 17:08:04  pvrserver.debug is set to 0
11/22/94 17:08:04  pvrserver.RetryInterval is set to 120 seconds
11/22/94 17:08:04  pvrserver.PollingInterval is set to 60 seconds
11/22/94 17:08:44  PVR DeadStart initiated
11/22/94 17:08:44  /usr/lpp/UniTree/etc/PVRINFO:
11/22/94 17:08:44 OPR      80000000 00000000
11/25/94 12:29:24  Rereading configuration parameters...
11/25/94 12:29:25  pvrserver.debug is set to 1
11/25/94 12:29:25  pvrserver.RetryInterval is set to 120 seconds
11/25/94 12:29:25  pvrserver.PollingInterval is set to 60 seconds
11/25/94 12:29:25  RebuildTask:  read device INFO files
11/27/94 18:49:44  MOUNT:  dev req mask = FFFFFFFF FFFFFFFF, VolumeID = AA0000
11/27/94 18:50:38  DEQUE:  VolumeID AA0000 not in a robot queue
11/27/94 18:50:38  DEQUE:  VolumeID AA0000 Mounted in Display queue
11/27/94 18:52:11  MOUNT:  dev req mask = FFFFFFFF FFFFFFFF, VolumeID = AA0001
11/27/94 18:53:10  DMOUNT:  VolumeID AA0000 removed from Mount Display
11/27/94 18:53:44  MOUNT:  dev req mask = FFFFFFFF FFFFFFFF, VolumeID = AA0001
11/27/94 18:54:02  DEQUE:  VolumeID AA0001 not in a robot queue
11/27/94 18:54:02  DEQUE:  VolumeID AA0001 Mounted in Display queue
11/27/94 18:55:22  DMOUNT:  VolumeID AA0001 removed from Mount Display
```

Figure 56. UniTree pvrsrvr.t Log File

repack.t The log for the repack utility process

shiva.t The log for the name server daemon shiva, which cleans up the trash can

tapemovr.t The log for the tape mover daemon

tapesrvr.t The log for the tape server daemon

```
11/22/94 17:08:11  =============== TAPE SERVER NEW RUN (NLS) ===============
11/22/94 17:08:11  Using Automatic parameter configuration

Device 1 is disk device /dev/rth1_prim
Device 2 is disk device /dev/rth1_secd
Device 3 is disk device /dev/rst1_prim
Device 4 is disk device /dev/rst1_secd
Number of Available Disks: 4
Number of Header Disks:      1 (logical)  2 (physical)
Number of Search Disks:      1 (logical)  2 (physical)
Logical Header Disk 0:
                             Tddevs disks 1 and 2.
                             Size is 4096 4k blocks.
Header map is at sector 1
                             Headers start at sector 2
Tape Record Disk device #'s: 1 (logical)  3 & 4 (physical)
Tape Record Disk size:       1024 4k blocks
Search Disk devices #'s:     1 (logical)  3 & 4 (physical)
Search Disk size:            1024 4k blocks
Search Sectors:              1 to 521
Tape Write Sector:           522
Tape Map:                    523 to 726
Book Sectors:                727 to end
File Duplicates:             2
Max File Duplicates:         4
11/22/94 17:08:11  Initial Tape Label: AA0000
11/22/94 17:08:11  Debug is off set to 0
11/22/94 17:08:11  Expected Number of UniTree blocks per tape: 11520
11/22/94 17:08:11  Disksrvr address is: 9.3.1.85
Tape device 1 is device /dev/rmt3
Licensed System Size is 1 GB.
Initing System Size to 0.000 GB; 0.031% full.
11/22/94 17:08:15  Tape Server Is Up And Running
11/22/94 17:08:15

******************   W A R N I N G ******************
If migration server (migsrvr) is waiting for files to be migrated,
restart it now to avoid delay.
*****************************************************

11/25/94 12:29:00  SigHUP received...
11/25/94 12:29:02  ParmReader working...
11/25/94 12:29:02  Rereading configuration parameters...
11/25/94 12:29:02  Read new debug value: 1
11/25/94 12:29:02  Read value for number of copies: 2
11/25/94 12:29:02  Configuration parameters reread.
11/25/94 14:26:16  Set Block Count: tape AA0003, BlockCount -1
11/25/94 14:29:06  Set Block Count: tape AA0002, BlockCount -1
11/25/94 19:03:44  MigFile ID = 2ed67afa.d1b50000, Label 1
11/25/94 19:03:44  Migrate Length 00000000.00c44000,numdups = 2
11/25/94 19:03:44  MountTape = AA0000,Index = 1
11/27/94 18:50:39  Tape mounted = AA0000
11/27/94 18:50:59  Allocate header (3, 7)
11/27/94 18:52:02  MigFile ID = 2ed67afa.d1b50000, Label 1
11/27/94 18:52:02  Migrate Length 00000000.00c44000,numdups = 2
11/27/94 18:52:11  AA0000 migration unmount,Index = 1
11/27/94 18:52:11  MigFile ID = 2ed67afa.d1b50000, Label 2
11/27/94 18:52:11  Migrate Length 00000000.00c44000,numdups = 1
11/27/94 18:52:11  MountTape = AA0001,Index = 2
11/27/94 18:54:03  Tape mounted = AA0001
11/27/94 18:54:21  Allocate header (3, 8)
11/27/94 18:54:21  AA0001 migration unmount,Index = 2
```

Figure 57. UniTree tapesrvr.t Log File

unfsmntd.t The log for the NFS mount daemon

unsau.log The log for the unsau utility process, which adds users to UniTree

unscr.t The log for the unscr utility process, which creates the name server database

unsspace The log showing the available space remaining in the name server database

unstc.t The log for the unstc utility process, which sets the trash can parameters

For any additional information, refer to the *UniTree System Administrator's Guide*.

4.3.4 Security Administration

There are essentially two tasks related to security management that should be performed by the UniTree administrator. They can be summarized as follows:

- Enforcing UniTree License
- Preventing general users from issuing some specific commands

4.3.4.1 UniTree License

At installation, UniTree comes up with a default license allowing the site to use up to 1GB of second-level storage (tape and optical). When this default size is reached, the migration of any files will be disabled, and the following message will be appended to the tapesrvr.t log file:

```
System size has exceeded License size. Call support for a larger license.
```

The following tasks need to be performed in order to obtain a new license:

- Run the command `uname` from the server in order to obtain the machine ID number:

```
# uname -m
000131981000
```

- Contact an IBM representative to obtain the new license key
- Run `smit unitree` to enforce the new license key:

  From the SMIT UniTree interface, follow the path:

  ```
  management of UniTree objects
  View/Change UniTree object
  View/Change Tape Server
  View/Change Tape Server Parameter Configuration
  ```

  Enter the new license value in the corresponding field, as shown in Figure 58 on page 112.

```
                        View/Change Tape Server Parameters

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Tape Server Debug enabled                           yes                          +
* Number of duplicate file copies                    [2]                           #
* Maximum number of duplicates allowed               [4]                           #
* Tape server start label                            AA0000
* Number of backup media                             [3]                           #
* License Value                                      [.T%T.$=R.$-Q/$!Q)#!A)#!A)#!A]
* License Warning Threshold (%)                      [90]                          #
* Tape Mover debug enabled                           yes                          +


F1=Help             F2=Refresh            F3=Cancel             F4=List
F5=Reset            F6=Command            F7=Edit               F8=Image
F9=Shell            F10=Exit              Enter=Do
```

Figure 58. SMIT UniTree View/Change Tape Server Parameters Display

4.3.4.2 Managing User Commands

All the UniTree commands are contained in the directory /usr/lpp/UniTree/adm/bin. All UniTree users can login to the server and issue these commands. It may become a requirement to limit user access to these UniTree utilities.

To perform this task, the system administrator can set the owner of the selected commands to root and change the access permissions to 0700.

4.3.5 System Availability

The UniTree file system is maintained by the name server through the name server data base, which acts as the directory for all UniTree data. The name server and tape server logical volumes are mirrored in order to increase their availability. If one of the two volumes in the pair is damaged, a new one can be defined and recovery achieved with no loss of data by simply following the procedure in the *UniTree System Administrator's Guide*.

The disk server logical volumes have no mirrored copies. The only way of assuring the availability of the data is to set automatic migration to occur frequently enough. Remember that in the case of failure, there is no way to recover data that has not been migrated.

UniTree provides two utilities to enforce the availability of data contained in archive devices:

- `backup`
- `swapback`

4.3.5.1 Backup Utility

The tape server performs an automatic backup of its internal structures onto BCKxxxx labeled volumes. This backup is not a complete backup, and its starting time is not controllable.

The `backup` utility allows a full backup of all tape server logical volumes to be performed. In order to increase availability, this utility could be run automatically. The utility writes on BCKxxxx labeled volumes. All utility messages are appended to the backup.t log file.

4.3.5.2 Swapback Utility

If the tape server has been customized in order to have at least two duplicate copies of migrated files, the `swapback` utility can be used to recover from tape or optical media failures.

UniTree processes only one of the duplicate copies at a time; so it is possible to swap from the failing media to another one. The `swapback` utility replaces the primary copy with the first available copy, allowing access to files even in the case of damaged units.

Automatic swapback occurs every time the `setlbk` command is issued to make a device unavailable.

4.4 Legato NetWorker

Administrative tasks can be easily performed using the NetWorker GUI, as has already been shown in 3.4.3, “Customization” on page 60. If only a TTY terminal is available, or a system which does not have AIXwindows installed, the character-based program `nsradmin` can be used to manage NetWorker:

```
# nsradmin
NetWorker administration program.
Use the "help" command for help, "visual" for full-screen mode.
nsradmin> visual
```

The major tasks of the NetWorker administrator are demonstrated in the GUI Administration menu:

```
Registration...
Clients...
Groups...
Schedules...
Policies...
Devices...
Servers...
Directives...
Notifications...
Pools...
Label Templates...
Jukeboxes...
```

Registration has already been described in 3.4.3, “Customization” on page 60. In the following sections, the remaining tasks will be described. All the examples explicitly refer to the GUI; for some of them, the corresponding character-based interface version will also be furnished in order to allow a better evaluation of the capabilities of this product.

4.4.1 Device Administration

One backup device has been automatically defined at installation time. More devices may be added because NetWorker supports multiple devices, either in series, concurrently or in a jukebox:

- With the basic product installed, the server uses one device at a time until it is full
- With the advanced server, or the concurrent device support module, the server can spread backups over all of the available backup devices
- With the jukebox module installed, the server can perform unattended automatic backups

4.4.1.1 Single Device Administration

The supported devices are:

- Half-inch magnetic tape (qimt) drives
- Quarter-inch cartridge (qic) drives
- 4mm (DAT) drives
- 8mm tape drives
- 8mm 5GB tape drives
- 3480 tape drives
- Optical disk drives

To add one of these devices to the server configuration, select the **Devices** panel from the Administration menu in the GUI main window.

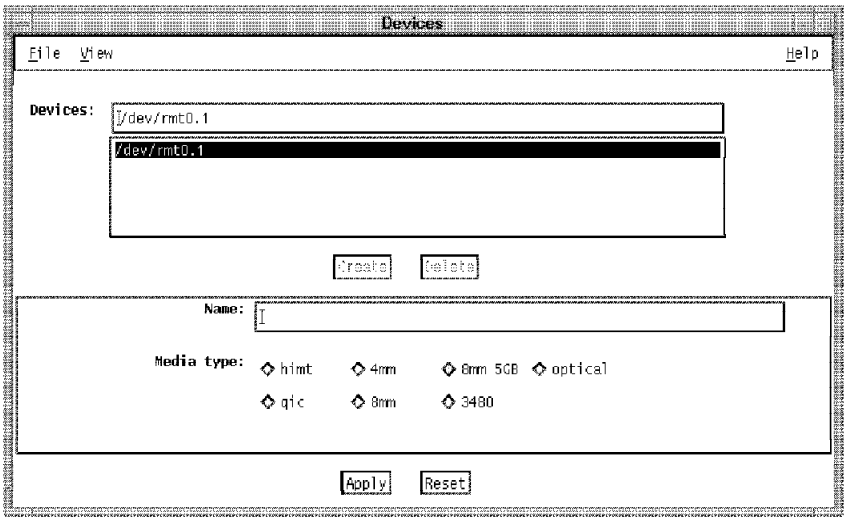

Figure 59. NetWorker GUI Administration: Devices Window

Click on the **Create** button, enter the pathname of the device, select the appropriate media type and click on the **Apply** button.

The same panel can be used to remove a device from the configuration.

Any change effected in this panel is displayed in the main window in the `Devices` section.

The same operation may be performed using the character-based program `nsradmin` in visual mode:

```
Command:  Select   Next   Prev   Edit   Create   Delete    Options   Quit
This is the first resource of 36

                        type: NSR device ;
                        name: /dev/rmt0.1 ;
                  media type: himt     qic    4mm    8mm    8mm 5GB    3480
                              optical ;
                     enabled: Yes   No ;

------------------------------
Keys: tab=next  return=do command  [a-z]=that command  H=help
```

- Move the cursor to **Create** and press Enter
- From the following display, select **NSR device** as the resource to be created and press Enter again
- From the following display, set the name and select, with the cursor, the media type of the new device. Press the Escape key when done

```
Create this new resource?  Yes   No
New resource

                        name: /dev/rmt3.1;
                  media type: himt     qic    4mm    [8mm]    8mm 5GB    3480
                              optical ;

------------------------------
Keys: escape=no  tab=next  return=answer  y=do create  n=no  H=help
```

4.4.1.2 Jukebox Administration

If the jukebox module is installed, the device may be configured by selecting the Jukeboxes panel from the Administration menu. The supported types are shown in Figure 60 on page 116.

Any time more devices need to be added to a jukebox, these devices should first be added using the Device window, and only after this, use the Jukeboxes window to add their pathnames to the jukebox configuration. The device pathnames must be entered in the same order as the devices are physically installed in the jukebox.

Refer to *Legato NetWorker Administrative Guide* for detailed information on jukebox administration.

4.4.1.3 Label Templates

Once the backup volumes have been defined, label templates can be considered. Label templates provide rules for naming and labelling backup volumes. The predefined templates can be used, or new ones added, according to the specific criteria.

Backup volumes are labelled automatically according to the chosen template. The labels generated by the predefined templates are composed by two or three fields separated by one of the following four characters:

. (period)
_ (underscore)
; (semicolon)
- (dash)

The numeric field is a three-digit number from 000 to 999.

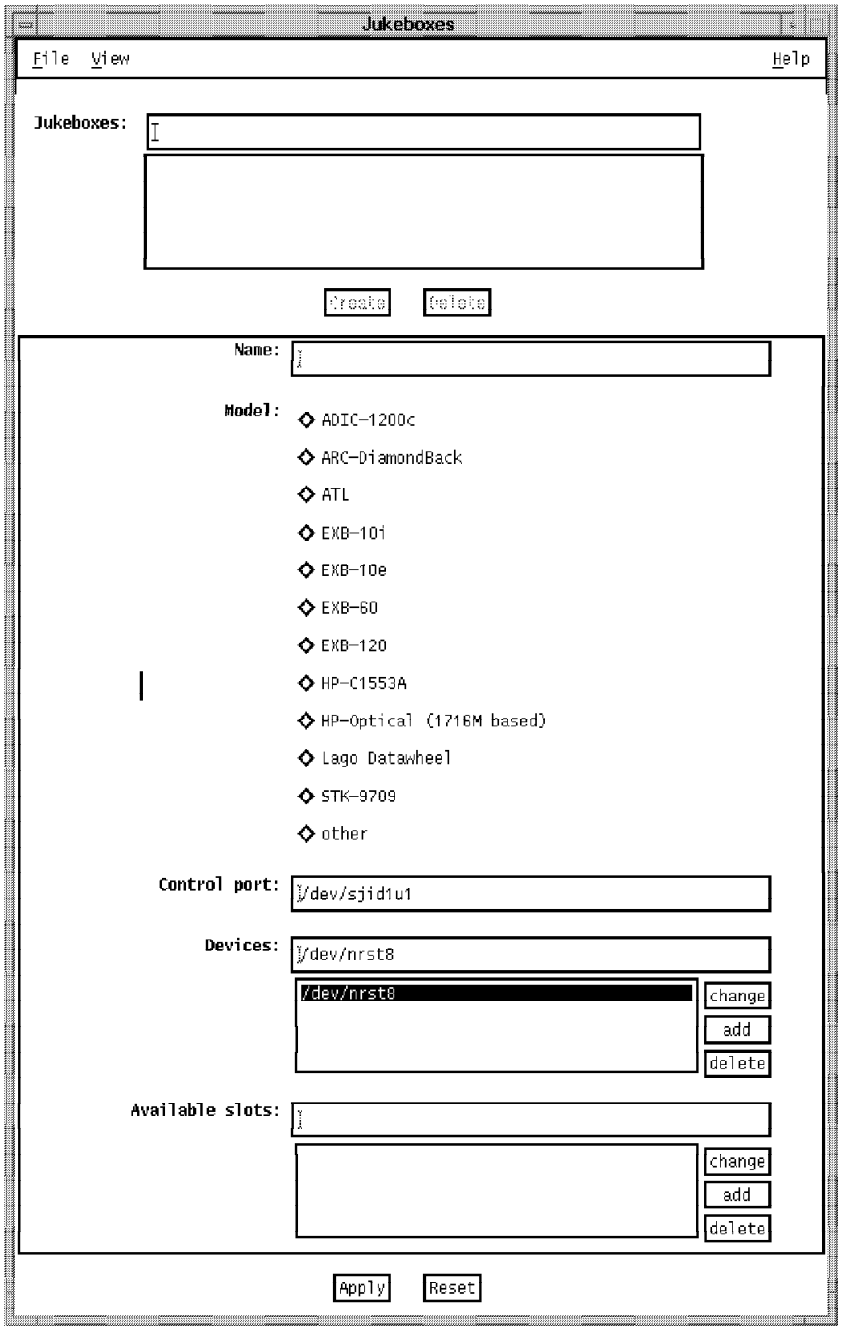

Figure 60. NetWorker GUI Administration: Jukeboxes Window

The NetWorker predefined templates are:

Archive This is used only for clients that need data archiving, such as DOS, OS/2 and NetWare. The generated labels have three fields separated by one of the separator characters. The first field is the server name; the second is the word `archive`, and the third is the volume number. For example:

`dori.archive.001`.

Default This is the one used by default when it is unnecessary to divide backup volumes by type. The generated labels have two fields separated by one of the separator characters. The first field is the server name, the second the volume number. For example:

`dori.001`

Full This is used for volumes containing full backups. Refer to 5.5, “Legato NetWorker” on page 149 for information on backup types. For example:

`Full.001`

NonFull this is used for volumes containing incremental or level backups. Refer to 5.5, “Legato NetWorker” on page 149 for information on backup types. For example:

`NonFull.001`

Offsite This is used for volumes being stored off-sites. For example:

`Offsite.001`

Two Sided This is used for optical media requiring labels on both sides. The generated labels have three fields: the first for the server, the second for the volume number, the third for the media side. For example:

```
dori.003.a
dori.003.b
```

Use the Label Templates window from the Administration menu for:

- Selecting the separator character preferred when using the predefined templates
- Adding user-defined templates

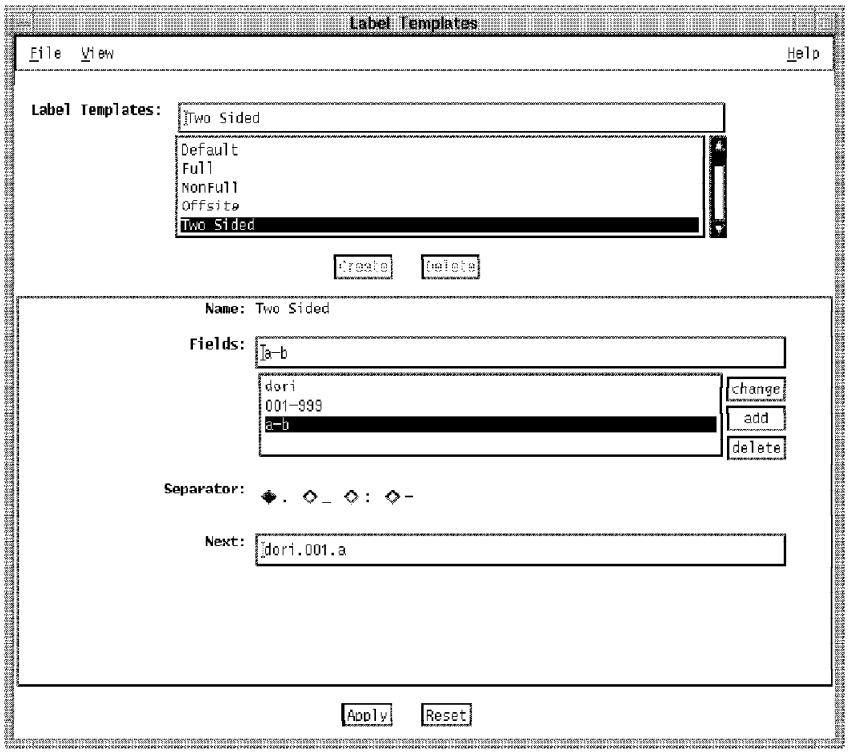

Figure 61. NetWorker GUI Administration: Label Templates Window

Refer to the *Legato NetWorker Administrative Guide* for detailed information on adding new label templates.

4.4.1.4 Pools Administration

A volume pool contains a collection of backup volumes that have specific data stored during the backup process. If a pool is not selected, all volumes will belong to the Default pool.

Each pool is associated with a corresponding label template. This means that there are five predefined pools:

- Archive
- Default
- Full
- NonFull
- Offsite
- Two Sided

Pools allow organization of backup data sorted by:

- Backup groups
- Clients
- Save sets (file systems)
- Backup levels
- Archive data

```
Command:  Select   Next   Prev   Edit   Create   Delete    Options   Quit
2 on 5 (on dori)

                        type: NSR pool;
                        name: Full;
                     enabled: [Yes]   No ;
              label template:  Archive   Default   [Full]   NonFull
                               Offsite   Two Sided ;
                      groups:  Default  [Test];
                     clients:
                   save sets:
                      levels: [full]  1   2   3   4   5   6   7   8
                               9    incr   manual ;
                archive only:  Yes   [No];
                     devices:  /dev/rmt0.1 ;
          store index entries: [Yes]   No ;

------------------------------
Keys: tab=next  return=do command  [a-z]=that command  H=help
```

Figure 62. The NetWorker nsradmin visual: Pool Display. The same pool is shown in graphical format in Figure 63 on page 120.

For example, the predefined `Full` and `NonFull` pools allow backup data to be divided between backup volumes according to the backup level. The full backup should be kept in a secure place, separate from the others.

In order to enable pools or to add new ones, select **Pools** from the GUI Administration menu.

Before creating a new pool, the corresponding label template should be created, as described in the previous section. After this, select the **Create** button in the Pools display shown in Figure 63 on page 120, and start entering the specifications to define the new pool characteristics.

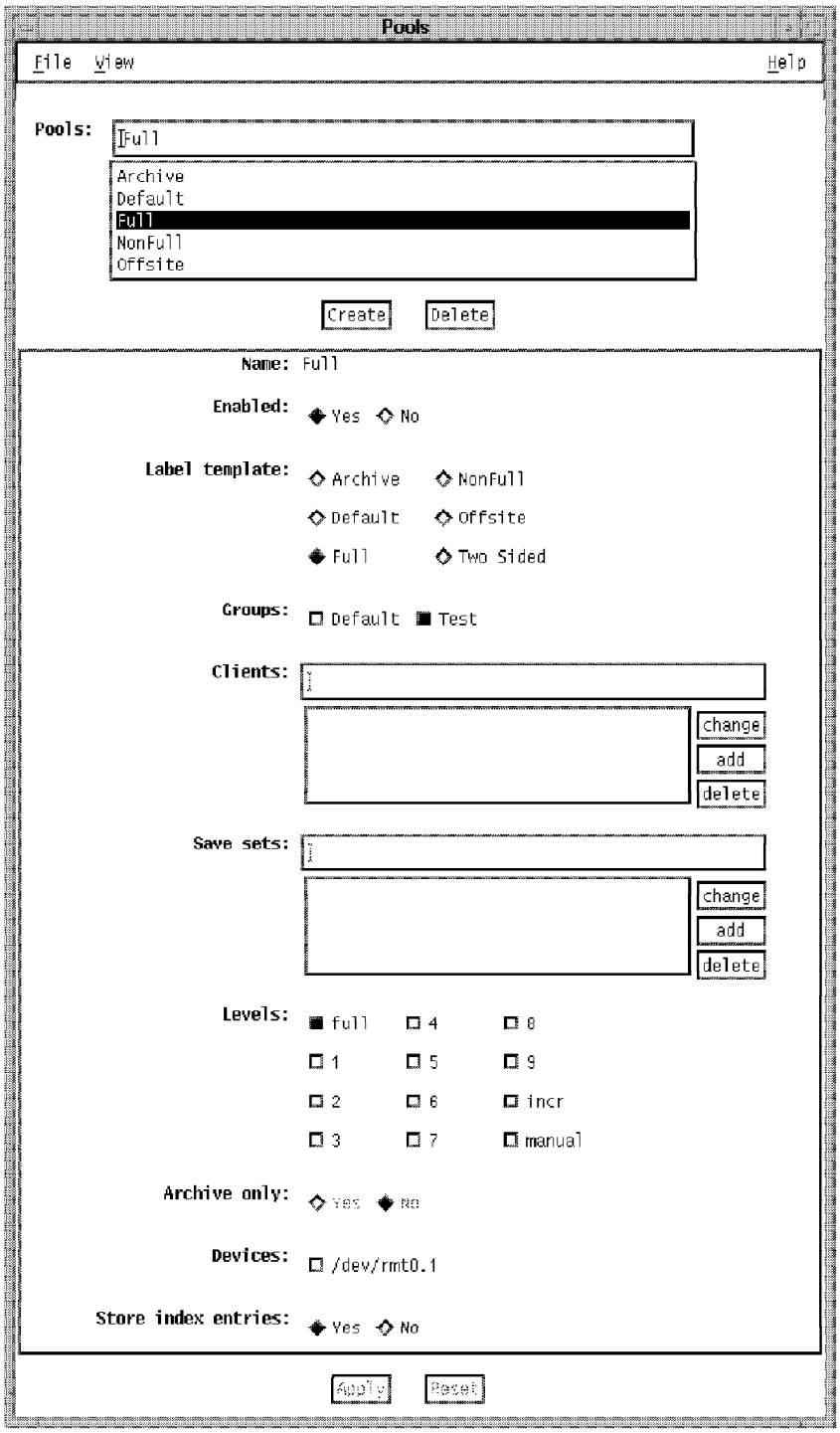

Figure 63. NetWorker GUI Administration: Pools Window

For example, in order to create a Confidential pool for the backups of restricted information from specific clients, thereby allowing storage of the backup volumes in a separate, safe place, enter the names of the clients holding this data in the Clients: field and the path names of the confidential file systems in the Save sets: field.

4.4.1.5 Device Operations

In order to perform daily operational tasks, such as mounting and labeling tapes or monitoring volumes usage, the NetWorker GUI Operation menu can be used:

```
Mount...
Unmount...
Label and Mount...
Label...
Backup...
Recover...
Indexes...
Volumes...
Group Control...
```

Labelling and Mounting Tapes: An example of labelling and mounting tapes has already been shown in 3.4.3, “Customization” on page 60. This section will just describe what happens in the normal day-to-day operations. When the system needs a device to perform scheduled backups, the following message will appear in the Pending display of the GUI Main panel:

```
media waiting: backup to pool 'xxxx' waiting for 1 writable backup
   tape or disk
```

Find a tape labelled according to the requested pool and mount it in the required device. After this, select **Mount** from the operation menu, or issue the `nsrmm -m` command from the AIX prompt.

If a tape of the requested type is unavailable, a new one can be labelled by using the Label display in the operation menu or by issuing the `nsrmm -l` command from the AIX prompt.

Managing Backup Volumes: Managing backup volumes is possible from the Volumes display off the option menu. The operations that can be manually performed on volumes are:

- Set a location for the volume
- Change the mode of the volume
- Remove the volume

From the display shown in Figure 64 on page 122, click on **Volume**: the three different selections will be displayed:

Set Location Allows the volume location to be specified as an optional reminder. If the volume is in a jukebox, NetWorker automatically updates the volume location

Change Mode Allows the volume mode to be changed

- *Appendable* It is possible to append further data to the volume and thus use it for other backups
- *Full* There is no more usable space on the volume

Recyclable The volume can be relabelled and overwritten. Notice that a volume becomes automatically recyclable when all its save sets are recyclable, because they have passed the expiration date set in their retention policy. If a volume is manually changed to recyclable, all its save sets become recyclable

Remove Used to remove a volume from the configuration. The steps to be performed are:

1. Remove the file entries from the file indexes
2. Remove the name of the volume from the media index

If a backup volume containing data that is still required is accidentally removed, the data can be recovered using the commands `scanner` and `recoverindex`, as shown in the *Legato NetWorker Installation and Maintenance Guide*

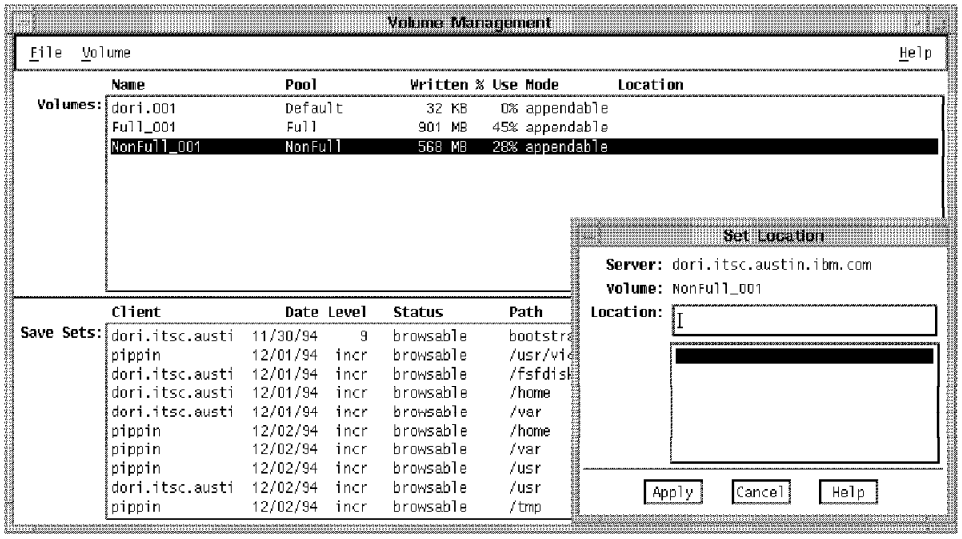

Figure 64. NetWorker GUI Operation: Volumes Management Window

Monitoring Scheduled Backups: The Group Control display from the Operation menu can be used to monitor scheduled backups. From this display, it is easy to:

- Check the status of scheduled backups
- View details of backups
- Immediately start a specified backup
- Immediately stop a specified backup
- Restart a previously stopped backup

4.4.2 User Administration

In the previous section, the administrative tasks involved in device management have been discussed. This section is going to focus mainly on the tasks related to the management of the client/server and users' environments.

4.4.2.1 Server Administration

The Server window from the Administration menu can be used to perform different tasks related to server management:

- defining additional administrators
- setting server parameters

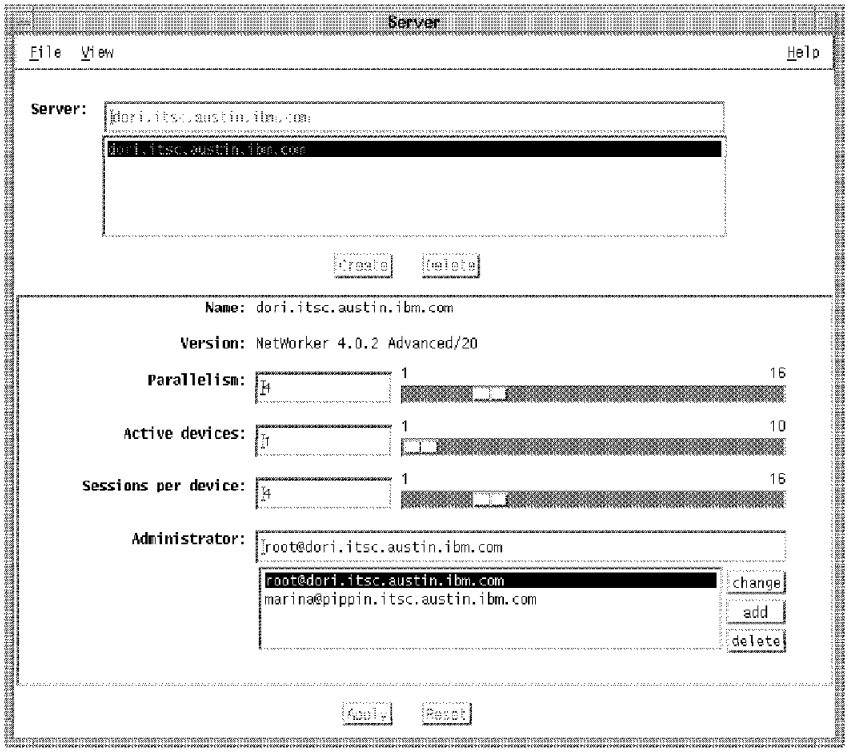

Figure 65. NetWorker GUI Administration: Server Window

Defining Administrators: The default NetWorker administrator is the root user of the server machine. Administrative authority may be required for other users on the same system or on other defined clients. To perform this task, select the **Server** window and enter the user and host name in the `Administrator:` field and click on the **add** button.

Previously defined administrators can be changed or deleted in the same way.

Notice that normal users can be defined as administrators, but they cannot perform any of the following tasks that are directly related to backup volumes:

- Operation: mount, unmount, mount and label, label, group control
- Administration: pools

These tasks can be performed only by root users on the server and on the defined client machines.

Server Parameters: From the Server window, some server parameters can also be defined. The server may back-up files from different clients in parallel, mixing their files on backup devices, in order to increase throughput. This feature is controllable with the `Parallelism` value. This can be changed, using the

corresponding sliding bar, as shown in Figure 65. Up to 16 client sessions can be processed in parallel.

If the NetWorker advanced server is being used or the concurrent device module is installed, the concurred devices feature can be used, which allows the server to use multiple devices for simultaneous backup and recover operations. Up to 10 active devices and up to 16 sessions on each device are supported concurrently. The appropriate changes can be made using the corresponding sliding bars in the Server window.

4.4.2.2 Client Administration

It has already been shown, in 3.4.3, "Customization" on page 60, how to define a user client to NetWorker. This section looks in more detail at the Client window from the Administration menu. The fields in the displayed window have the following meaning:

Schedule — Any client has one or more associated schedules for automatic backup processing. NetWorker provides five predefined schedules. Custom schedules can be defined from the Schedule window

Browse policy — This specifies how long backed up file entries for this user will remain in the online file index to be browsed by the user. Customized policies can be added from the Policies window

Retention policy — This specifies how long backed-up file entries for this user will remain in the online media index to be restored by the user. After the retention period, the entries are marked recyclable

Directive — This contains specific instructions to assist the backup process, allowing specific files to be skipped or data to be compressed. NetWorker provides four predefined directives, but customized versions can be created from the Directive window

Group — The defined backup groups. Every client should be associated to at least one backup group

Save Set — The file systems to be backed up. The default is **All**. The specific file systems requiring automatic backup to the server can be added here. This feature allows the scheduling of file systems to be backed up in different groups, at different times

Recover access — The hostnames of the machines that will be authorized to browse and recover client files. The NetWorker default is that only the client itself can browse and recover its files.

For any additional information about the previous features, refer to 5.5, "Legato NetWorker" on page 149.

4.4.2.3 Notifications

NetWorker displays all the important messages on the Messages and Pending fields in the GUI main panel. Additionally, it provides eight types of notifications for the most important events in the system, together with the ability to customize them according to site-specific needs.

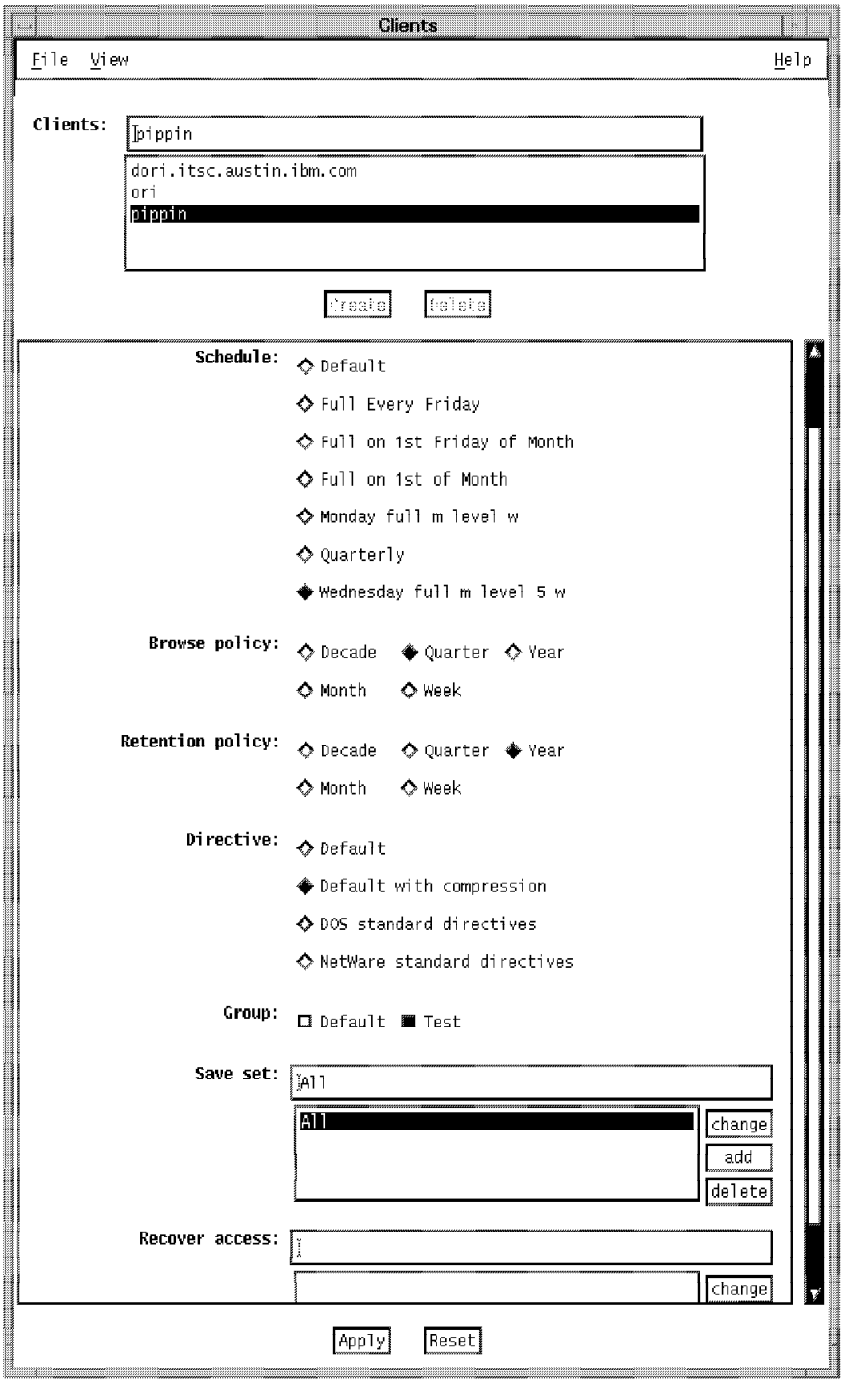

Figure 66. NetWorker GUI Administration: Client Window

Select the **Notifications** window in the Administration menu. The display will be as shown in Figure 67 on page 126.

Name The event notification name

Action The type of action the system will take to promote notification of the event. This can be changed to a method of choice (for example mail to root user, message display at console)

The predefined event notifications are:

Registration	The server sends a message to root user when the products are not properly registered
Log default	Uses the AIX syslog to distribute notification about all NetWorker events
Index size	Sends an electronic mail message to the root user when the online indexes begin to run out of space
Savegroup completion	Sends an electronic mail message to the root user when the server finishes a client backup
Tape mount requests	The server issues three mount notifications for backup media requests. The first has no action, it is simply displayed in the Pending field of the main window; after 15 minutes, as the second notification, the server sends an alert to the logger; after another 37 minutes, mail is sent to the root user
Jukebox request	This is the server response to a media request when the jukebox is in use

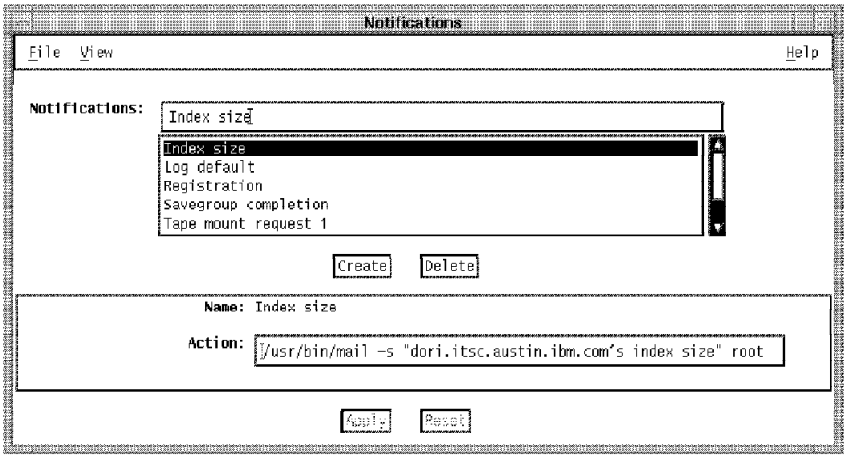

Figure 67. NetWorker GUI Administration: Notifications

The same information can be obtained using the nsradmin utility. Select **visual** at the utility prompt, and on the full-screen menu, choose **Select**. The system will ask for the resource type to be displayed. In this case, just select **NSR notifications**. The following display will appear:

```
Command:  Select   Next   Prev   Edit   Create   Delete    Options   Quit
1 on 7 (on dori)

                        type: NSR notification;
                        name: Log default;
                      action: /usr/ucb/logger -p daemon.notice -f -;

------------------------------
Keys: tab=next  return=do command  [a-z]=that command  H=help
```

Figure 68. NetWorker nsradmin Visual: Notifications Display

Select **Edit** if the action is to be changed or **Next** to see the next notification type.

4.4.3 Security Administration

There are no specific NetWorker administration tasks directly related to security administration. The intent of this section is simply to summarize some previously described issues that are related to the security administration of the NetWorker environment.

4.4.3.1 Enabling and Registering the Product

NetWorker code must be enabled with the `nsrcap` command and registered, using the Registration window from the Administration GUI, before it can be used on the server machine. These tasks have been described in 3.4.3, “Customization” on page 60.

4.4.3.2 User Access

Any user from a previously defined client can request NetWorker services in order to backup, browse and recover their own files.

Only defined administrators can act on files not belonging to them, and administrators that do not have root authority cannot manage backup devices, as previously described in 4.4.2, “User Administration” on page 122.

4.4.3.3 Notification

This utility allows administrators to monitor system activities and to manage any event requiring intervention.

4.4.4 System Availability

Every time NetWorker performs a backup, it saves all the online indexes for the backed-up clients and the server indexes. The server's index backup is called *bootstrap*, and this information is also sent to a printer, providing a hard copy that can be kept in a file for recovery purposes. All of the NetWorker indexes and control structures are contained in the /nsr directory.

In order to be prepared to recover from a hard disk failure, keep a printed copy of the following:

- The output from the `df` command
- The output from the `lslv` command
- The NetWorker bootstrap records

In case of a hard disk failure on the server machine, recovery can be effected as follows:

- Replace the damaged disk with a new one of the same type and recreate all file systems using the saved information. The system will need to be reloaded and rebooted using the same hostname and disk partitioning in order to allow NetWorker to restore files.
- If the /nsr directory is still available, it is possible to simply reinstall the product as described in 3.4.2, “Installation” on page 58. The online indexes and the license enablers will be automatically available.

  All file systems can be recovered from the damaged disk using the `recover` utility

More detailed information on recovery can be found in Appendix A of *Legato NetWorker Installation and Maintenance Guide*. An example `recover` utility session follows:

```
# recover
NetWorker> add /
NetWorker> delete /bootrec
NetWorker> list
NetWorker> force
NetWorker> recover
```

4.4.4.1 Recovering the /nsr Directory

If the /nsr directory has been damaged, the bootstrap information is required in order to recover it. If a hard copy of the last bootstrap records does not exist, one can be obtained by mounting the backup device and issuing the `scanner` command:

```
# scanner /dev/rmt0.1
scanner: scanning 8mm tape NonFull_001 on /dev/rmt0.1
client name  save set          save time     level  size  files  ssid
dori.itsc.au bootstrap        11/30/94 17:40  9    22012    11   4432
dori.itsc.au /fsfdisk         12/01/94 16:30  i      244     1   4448
dori.itsc.au /home            12/01/94 16:30  i      448     2   4449
dori.itsc.au /var             12/01/94 16:30  i   152136    80   4450
pippin       /usr/vice/cache  12/01/94 16:29  i   895052   130   4451
pippin       /var             12/02/94 10:58  i    30944     8   4452
pippin       /home            12/02/94 10:58  i  1178228     9   4453
pippin       /usr             12/02/94 10:59  i        4     0   4455
pippin       /tmp             12/02/94 11:01  i      184     1   4456
pippin       /                12/02/94 11:02  i    25772    31   4457
dori.itsc.au /usr/nsr/index/pippin 12/02/94 11:02  9  113028    2    4458
 ...
dori.itsc.au bootstrap        12/02/94 11:50  9    29140    11   4462
 ...
scanner: done with 8mm tape NonFull_001
#
```

Figure 69. The NetWorker scanner Command Output

Obtain the last bootstrap ssid (save set id) and start a `recoverindex` interactive session. All of the fileindex structures will be recovered. The /nsr/res directory containing clients and registration information will be also recovered. If there was a previous /nsr/res directory on disk, the utility creates a new directory named /nsr/res.R. Shutdown the NetWorker program, rename the directory and start NetWorker again.

4.5 Product Comparison Summary

The products discussed in this document will be compared with regard to administration on the basis of the following criteria:

- Automation
- Central management
- Functions
- Ease of use

- Security

4.5.1 Automation

Automation, in relation to administration, is concerned with the degree to which administrative functions can be automated.

4.5.1.1 ADSM

Administrative tasks within ADSM can be performed at the server or from an administrative client elsewhere in the network. From the server, a command line interface is used, while the administrative client provides both GUI and CLI. When the GUI is used, tasks are performed manually in the sense that there is no way to action similar operations automatically. From the command line interface, however, macros can be used. A macro can contain a sequence of commands that need to be performed on a regular basis, thereby providing a degree of automation to the process. With ADSM Version 2, all ADSM server commands can be scheduled using the central scheduling facility, with the exception of the `Q ACTLOG` command.

4.5.1.2 FSF

All FSF administration is performed manually from the SMIT interface or by issuing FSF utility command from the AIX command line.

4.5.1.3 UniTree

All UniTree administration is performed using the SMIT interface, or again, using utility commands from the AIX command line or FTP command line.

4.5.1.4 Legato NetWorker

All Legato NetWorker administration is performed manually via a GUI or a character-based visual interface though there are certain lower- level recovery tools that are used from the command line.

4.5.2 Central Management

Central management is concerned with the flexibility of the administrative interfaces. Essentially, this relates to where tasks can be implemented from.

4.5.2.1 ADSM

ADSM allows central management of the server from the server console, though it is not recommended to restrict management solely to the central console. Additionally, an administrative client, accessible via a CLI or GUI, is provided that allows full administrative capability from any machine in the network that the administrative client is installed on.

4.5.2.2 FSF

FSF administration must be performed at the machine where the FSF product is installed. This can be effected remotely if a Telnet session to the FSF machine is available.

4.5.2.3 UniTree

UniTree administration must also be performed at the machine where the server is installed. Again, this can be effected remotely if a Telnet session to the UniTree server is available.

4.5.2.4 Legato NetWorker

NetWorker administration can be performed from the server or any client machine in the network. The interface can be via the GUI or the character-based visual interface.

4.5.3 Functions

This section looks at the administrative functionality available and comments on those tasks required to enable effective use of the products. More detail on specific functions is available in this chapter.

4.5.3.1 ADSM

ADSM provides the following administrative capabilities:

- Devices

  Prior to the use of a device type for a storage pool, it must be defined to ADSM. Disk, optical and tape devices are supported. The sequential devices can be standalone or in a library. Volumes within libraries can be defined as scratch, meaning that ADSM can dynamically acquire them as needed. A default set of definitions are provided, but new devices will require definition.

- Storage pools

  Storage pool occupancy can be monitored and their contents audited to confirm consistency with the database. Files can be deleted and moved, and volumes can be added or deleted. A set of default storage pools are provided though these can be updated and new ones defined as required.

- Administrators

  Administrators can be added, updated and deleted. Various levels of authority can be granted. Individual administrator access can be temporarily revoked and reinstated. At least one administrator must be defined in order to manage ADSM administration.

- Clients

  Clients can be added, updated and deleted. Access can be temporarily revoked and reinstated. Client nodes must be defined to the server before they can access server functions.

- Schedules

  Schedules to automate backup and archive for clients can be defined, updated, deleted, and associated with particular clients. In addition, with ADSM Version 2, it is now possible to schedule restores, retrieves, ADSM macros, client operating system commands and ADSM server commands.

- Availability

  The ADSM database can be backed up and restored, and as a last resort dumped offline, for availability purposes. The database can then be restored, if required and any inconsistencies automatically repaired prior to reload.

  The ADSM database can be mirrored to enhance availability.

4.5.3.2 FSF

FSF provides the following administrative capabilities:

- Devices

  FSF requires logical volumes for its operation. These can be defined and managed through the FSF SMIT interface. This process is essential and is part of the initial file system definition.

- File system

  FSF file systems can be defined and managed. Again, this forms part of the initial definition and is a requirement for space management.

- Daemons

  FSF daemons can be started and stopped. Generally, the daemons are started automatically when an FSF file system is mounted. In some cases it may be necessary to stop and restart the daemons to refresh internal data structures.

- Fine tuning

  FSF automatic operation can be adjusted to govern migration behavior. This function is not required but performed to adjust FSF space management behavior to local requirements.

- Users

  Users can use command line administrative utilities to implement FSF migration functions. This allows a degree of flexibility to users in that they can override global file system definitions for individual files.

4.5.3.3 UniTree

UniTree administration offers the following functions:

- Disk cache management

  Disk cache status can be monitored, volumes can be added and removed and files purged from the cache. Disk cache definition must be performed initially.

- Tape and optical devices

  The status of tape and optical devices can be monitored as well as various operational characteristics updated. Tapes can be labelled and information on sequential media repacked to remove fragmentation. Tape and optical servers must be defined if these types of devices are to be used.

- User management

  Users can be added, deleted and their file access permissions changed. Trash can time-outs can be altered and file camping defined. Users must be defined at the UniTree server corresponding to the users on the client machines.

- Tuning

  Various automatic server operations can be tuned, including migration, purging and repacking.

- Monitoring

  System activity can be monitored manually through a variety of log files.

- Availability

UniTree provides utilities to backup internal structures and to recover lost files from copies stored using the duplicate feature.

4.5.3.4 Legato NetWorker

Legato NetWorker provides the following administrative functions:

- Devices

  Devices can be added and removed from the configuration. Volume pools can be defined that consist of a number of backup volumes. Each volume pool has a label template associated with it that enforces how labels will be written to volumes in the pool. Label templates can also be defined and associated with pools. Defaults are provided, but new definitions must be created for any new devices.

- Operations

  Devices can be mounted and unmounted. Labelling operations can be initiated, backups and restores started and volumes managed.

- Server

  Administrators can be defined and server parameters set. At least one administrator should be defined to manage the product. The types of server parameter that can be adjusted include the degree of parallelism allowed, the number of active devices and the number of sessions per device.

- Clients

  Client administration involves defining custom schedules and managing backup parameters such as file retention and directives. Again, defaults exist, but clients will wish to customize these to their own specific requirements.

- Availability

  NetWorker provides several utilities that assist in the recovery of NetWorker in the event of disk crashes or tape media errors.

4.5.4 Ease of Use

This relates simply to the type of administration interface provided and how easy it is to use.

4.5.4.1 ADSM

ADSM provides both command line interfaces and graphical user interfaces for administration. All ADSM administrative functions can be performed from either interface, with the exception of macros which must be used from a CLI, and tasks such as device definitions. The GUI in Version 2 is far easier to use than the previous version. Access from any machine in the network running the administrative client further enhances ease of use.

In addition, the ADSM space management client provides graphical status indication.

4.5.4.2 FSF

FSF administration is a combination of SMIT menus and command line functions. Operation is only from the machine with FSF loaded. FSF administration is not difficult, and the creation of a GUI purely for this task is probably unnecessary. Status indications are also non-interactive at the command line.

4.5.4.3 UniTree

UniTree administration is also a combination of SMIT menus and command line utilities. These operations must also be performed at the server machine. Status information is also displayed from the command line and is not interactive.

4.5.4.4 Legato NetWorker

Legato NetWorker provides both a GUI and a character-based full-screen interface for administration. All administrative functions can be performed from either interface. The ability to send error indications via user-selectable mechanisms also enhances usability.

4.5.5 Security

Security relates to the level of function provided by a product to secure access to the data and facilities within that product.

4.5.5.1 ADSM

ADSM provides a number of security mechanisms:

- Client/server authentication

  A mutual suspicion authentication process is used to ensure that only valid clients can access a server. Any client (be it a node, an administration or an application) must provide a password if authentication is enabled at the server. In addition, a password expiry period, after which passwords must be changed, can be defined at the server.

- License management

  License management no longer relies on NetLS with ADSM Version 2. The process is a lot more flexible, with build-to-order and build-to-plan options as well as try-and-buy or standard ordering of the software. Individual components can be purchased and licensing of the client components is done on a number of clients basis only. Various levels of device support options are also available as well as special licenses to allow the sharing of devices between servers.

- Client user access

  By default, any user at an authorized workstation can use ADSM services. The administrator is able to restrict this usage by group or by individual user at each client.

4.5.5.2 FSF

FSF does not provide any explicit security mechanisms. However, normal file access permission should be used to restrict unauthorized access to information.

4.5.5.3 UniTree

UniTree also requires a storage capacity-based license in order to operate. If the amount of UniTree storage required exceeds the current licensed limit, a new license will be needed.

The only other security possible is simply standard FTP security whereby users of FTP services are required to provide a userid and password in order to access the service. This is not the case for NFS.

In common with FSF, access to UniTree utilities can be restricted by using file access permissions.

4.5.5.4 Legato NetWorker

NetWorker also does not provide any specific security-related functionality. Having said that, only previously defined authorized administrators can act on files not belonging to them, and only previously defined clients can access the server.

The NetWorker code must also be enabled with a security string provided by Legato. If a further registration string is not obtained from Legato within 45 days, the product will disable itself.

Chapter 5. Backup and Restore

As businesses become more dependent on the information stored on file servers and workstations, the ability to recover lost data is becoming more crucial. There are many factors which influence the best backup strategy to use for a specific environment. As there is no overall “best way” to back up data, it is necessary to evaluate the importance of these factors in the current environment.

When developing a backup strategy, it is necessary to define the requirements that the strategy must satisfy. Factors that will require consideration when defining the requirements for a backup strategy include:

Types of events	The incidents that may occur
Speed of recovery	How quickly recovery is required
Backup windows	How often to perform backups
Recovery points	To which point in time recovery is required
Units of recovery	Which files need to be recovered to the same point of time

There are many products on the market today which will perform backup and recovery, and the most appropriate for the particular environment should be selected.

These tasks are discussed in detail in the two main sections of this chapter.

5.1 Backup Strategy

A business's approach to the developing of a backup strategy is largely dictated by application requirements or, in broader terms, by its business needs. Using the business requirements as a guideline, the following arguments should be considered:

Value	The cost in time and money for replacing files that are lost
Change	How often files change. In some environments, losing even part of one day's input would result in lost revenue
Media	The media capacity and the performance of devices are key elements for the selection of the frequency and the completeness of backups
Portability	Media supported by devices in any of the locations where a restore may be necessary is a good idea

A detailed description of data backup concerns and the different backup techniques available follows:

5.1.1 Data Backup

When defining a backup strategy, the first consideration is how data is distributed in the environment.

- In a centralized environment, almost all the data is in one location, making it easy to automate backups. On the other hand, the performance could be affected since all applications are centrally located

- In a distributed environment, data can be found in user workstations and file servers, requiring backups to be performed at the workstation site and making automation more difficult. On the other hand, there is increased productivity and fewer performance problems.

Another important issue is whether or not the backup process should be centrally managed (automatically or manually) or whether it should be locally managed at the individual workstations.

From the data backup perspective, data should be divided into the following categories:

- User data - private, shared, locally customized
- Application data - generally easy to reproduce since this is mainly prepackaged software, and any customization can been easily recreated
- System data - operating system, system applications and configuration files

5.1.1.1 Databases

Many different techniques can be used to backup data managed by RDBMSs. The most common are:

- Disk mirroring - the process of writing the same data to multiple storage devices at the same time. This allows recovery from hardware failures but not from user errors
- Offline backup - this requires shutdown of the database before starting the backup operation
- Online backup - only some RDBMSs support this backup technique because it involves the use of log files during the recovery process to restore the database to a fully consistent state
- Database export - the export/import utilities permit operation on logical objects rather than on the whole database
- Full backup - this involves the backup of data files, RDBMS log files and control files. To perform one offline, database utilities can be used. To perform an online full backup, RDBMS utilities must cooperate with a storage management product
- Partial backup - to backup a subset of the database. It is necessary to consider how to select a subset in order to have a complete logical unit of recovery from the point of view of the application
- Log File backup - if the units of recovery are too large to be backed up frequently or if the load on the network has a detrimental effect on other processes, backup can be confined to the RDBMS' log files. Complete database recovery is effected by:
  - Restoring the database from a full backup
  - Restoring the log files
  - Applying the log files to the database

5.1.1.2 Security

The last but not least issue to be considered is the security of data during the backup/restore operations.

- Media security - almost all of the business' important data is on backup media; so the media should be protected accordingly. Also, bulk erasing of media could be one possible solution to allow recycling.
- LAN security - where all data involved in a backup operation is being sent over a LAN, some level of encryption, or at least compression algorithms, is necessary to avoid the possible exposure from LAN monitors
- Restore security - the backup/restore product should be capable of controlling who (with what authentication) can restore a file

5.1.2 Backup Types and Techniques

Discussions on the implementation of a backup strategy require a good understanding of the different backup types and techniques. There are four main backup types:

- Short-term backups - these are mainly user related. They are taken frequently, perhaps daily, and can be kept on site, to be convenient for immediate restore operations. Only a limited number of backup versions should be retained, and after some months, media can be recycled. Large disk pools and storage jukeboxes make good short-term backup devices
- Long-term backups - these are mainly business related. They are taken at logical times in processing cycles, monthly for example. All related files must be synchronized in order to have a complete restorable image of the application. Removable tapes make excellent long-term backup devices
- Archival backups - required for legal and/or historical purposes, these are intended to last for decades. They are described in Chapter 6, “Archive and Retrieve” on page 165
- Disaster recovery backups - these must be located off-site, and all data must be synchronized

In addition to the previous functional characteristics, backups can be divided into different levels, each corresponding to a different backup technique:

- Full backup - conceptually, the simplest. All the selected files are written to the backup media, regardless of whether they have changed. This makes a restore operation easy and provides the maximum convenience for the user. Of course, it's not practical on a short-term basis because of the cost in terms of media devices
- Incremental backup - done on a regular basis, this includes only that data changed since the last incremental or full backup. It's the cheapest technique in terms of media but also the most complex to manage and restore
- Progressive backup - done on a regular basis, this includes all data changed since the last full backup. It uses less space on backup media when compared to full backup and is faster to restore from than incremental
- Archived backup - all data is backed up together and kept for longer periods of time.

5.1.3 Recovery Issues

In order to select the most suitable backup strategy for an environment, the types of failure which may arise and methods for dealing with them must be considered. The most common types of failure are:

- Disasters (fires, floods)
- Hardware failures
- System management mistakes
- Viruses
- User errors

A well-defined strategy includes contingencies for less serious problems, such as local failures. A local failure, whether it be a media or a network problem, can usually be corrected very quickly and does not require extensive advance planning to resolve. On the other hand, a comprehensive backup recovery plan could help to limit the effects of any type of disaster.

5.1.4 Summary

With a structured process for backing up data in an enterprise, there is greater assurance that data will be available when it is required. The risk of losing valuable data has been substantially reduced.

Advantage may also be taken of more reliable server storage, rather than that provided on local workstations. The system on which the server storage resides may also have a higher level of security enforcement and control. The server storage might provide a higher level of security simply by being in a *glass house* (a central data center with controlled access).

The “perfect” recovery strategy can be conceived, but in most cases will be quite impossible to implement as it often requires substantial investments in hardware, software and networks. While still looking for a compromise, care should be taken in the selection of the storage product that best fits the current environment. The ideal product should:

- Support all of the platforms in the network, with a commitment to supporting any additional platform in the future
- Use a consistent user interface across different platforms so that a single administrator can easily back up and recover dissimilar platforms
- Provide a feature set that is sufficiently rich to back up and recover the environment for several years
- Deliver the performance needed to back up the volume of files stored across the entire network

5.2 ADSTAR Distributed Storage Manager/6000

ADSM provides backup and archive services for all locally mounted file systems, Network File System and Andrew File System. It does not back up character special files, block special files or pipes.

Backup services are requested from ADSM when there is a requirement to save files that can be restored if the original files are damaged or lost. The following

section provides a brief description of the major features and product techniques involved in the backup process together with some usage suggestions.

5.2.1 Policy Management

ADSM allows management of the backup and archive process based on policies established for the enterprise. The granularity of control available is down to the file level. An overall system policy can be established or policies by department, by user or by file name. The elements of ADSM policy management are as follows:

Policy Domain: This is a logical collection of clients working according to the same set of policy needs. It provides a logical way of managing backup and archive policies for a certain group of client nodes. As many policy domains as required can be defined to ADSM. They may be used to group together clients that have similar storage requirements, select the management classes to which users have access, and so on. Each policy domain can contain one or more policy sets.

Policy Set: This is the container for Management Classes. Only one policy set can be active in a policy domain at any one time. Each policy set contains a default management class and other additional management classes.

Management Class: This defines how backed-up and archived data should be managed. It contains a backup copy group and/or an archive copy group. Client data gets bound to a specific management class in the sense that each backed-up or archived file is associated with it. Users can use the default management class or explicitly select another one that is within the active policy set to which they have access.

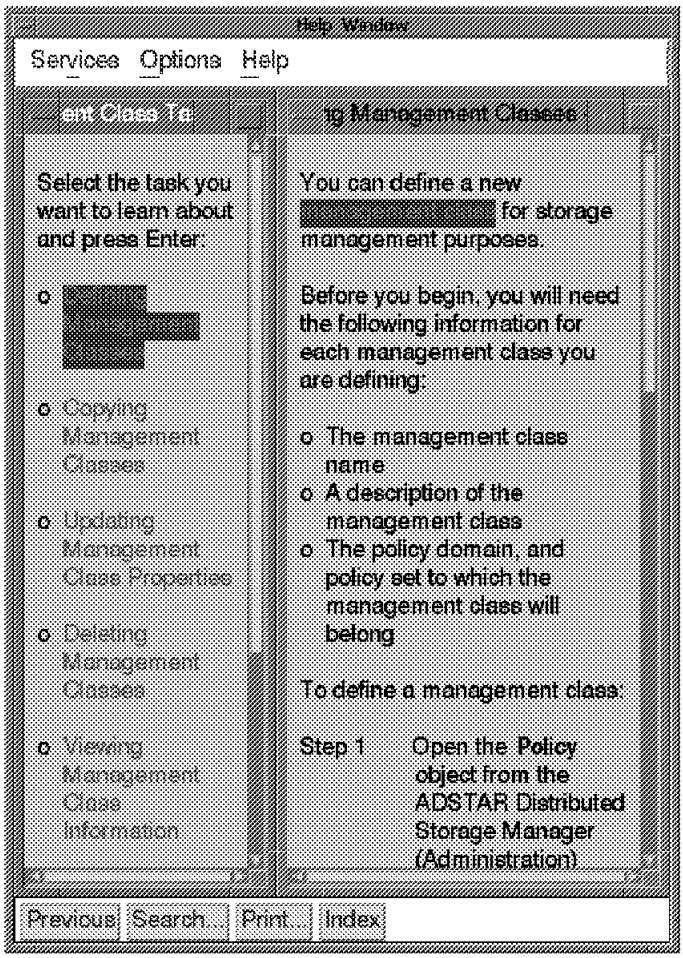

Figure 70. Management Class Help GUI Display

Copy Group: This contains parameters that control the generation and expiration of backup and archive data. The span of control is at the file level. Both backup and archive copy groups have the same set of parameters, with one exception, there is no concept of versioning for archived files. These parameters are:

destination The storage pool where the server stores the backed-up files

frequency The minimum number of days between incremental backups. Archived file frequency is always when required (command)

versioning How many versions of a backup file that should be maintained

retention How long to retain backed-up and archived files

mode Is the mode of backup (`modified | absolute`). For archive files, it is always absolute

serialization How files are handled if they are being modified during the backup or archive process:

- Static - ADSM will not back them up
- Shared static - ADSM will retry the backup the specified number of times
- Shared dynamic - ADSM will backup the file in any case at the last retry
- Dynamic - ADSM will backup the file anyway at the first attempt

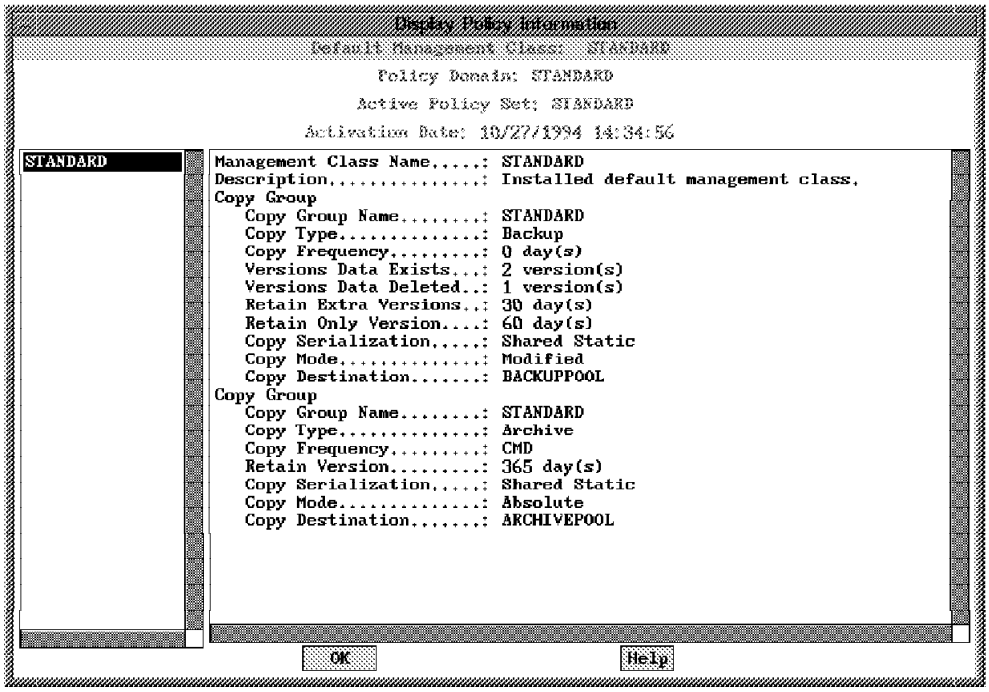

Figure 71. Policy Management GUI Display

5.2.2 Central Scheduling

ADSM can automatically trigger operations on selected nodes. These operations can be backups, restores, archives, retrieves, ADSM macros, client operating system commands or ADSM server commands. The central scheduling facility is a client/server process allowing the domain administrator to set up schedules and then associate client nodes with schedules in a nondisruptive manner. Clients can be set up to query the server for schedule details. They perform their designated operations according to the assigned schedule and return results to the server for logging. If backups need to be automatically run during off hours, a policy needs to be enforced that requires users to leave their workstations powered on.

The administrator can assign a priority to the client nodes so that clients containing more crucial data can be given preferential treatment.

The defined schedule can be periodic, for example every night, or occur at a specific date and time. The duration of a window and the period between windows can be defined. ADSM can spread the triggering of individual node events over a percentage of the duration period with the command `SET RANDOMIZE nn`. The randomized percentage of the duration is always at the beginning of the window to allow activities to complete. The system default is 25 percent.

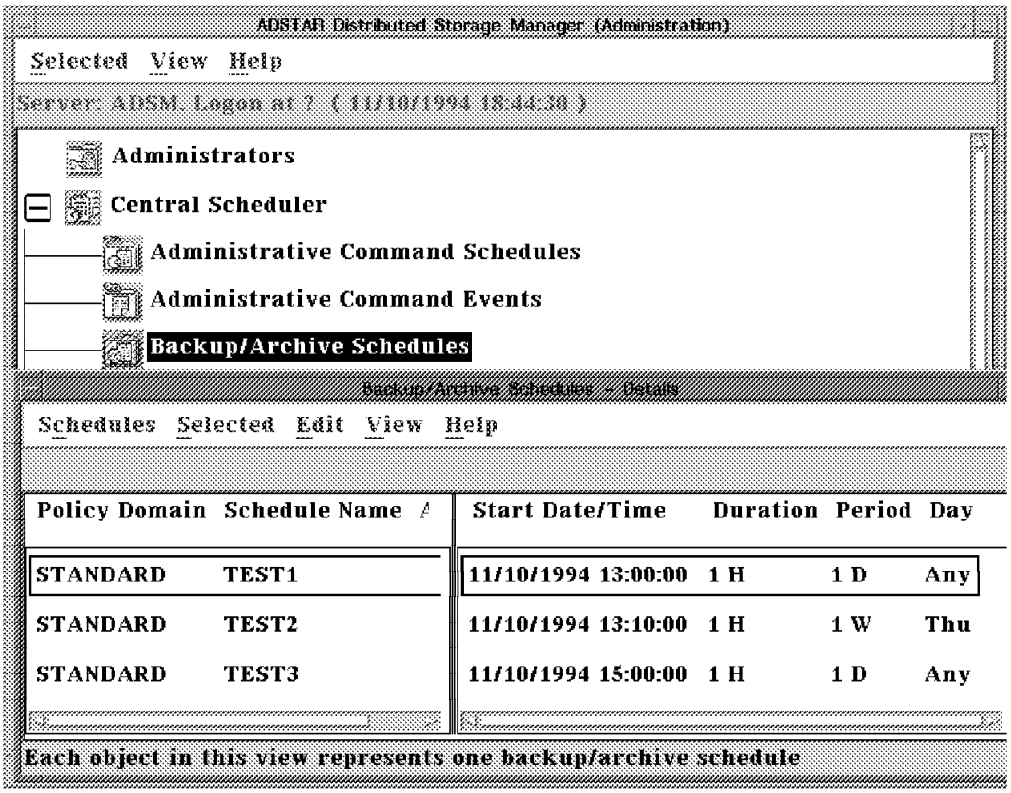

Figure 72. Central Scheduler GUI Display

Two types of scheduling modes are supported:

- Client polling

  This is supported on all client workstations over all communication methods. The client node periodically queries the server for a scheduled operation and the date and time the operation is to start. The server sends the information to the client. When the scheduled starting time arrives, the client notifies the server that a scheduled operation is starting and executes it. When the operation has been completed, the client notifies the server.

- Server prompted

  This is supported on all client workstations using the TCP/IP communication method with the server. The client node waits to be prompted by the server to begin the scheduled operation. When the operation has been completed, the client notifies the server. Server prompted scheduling allows the server to control when clients are contacted to perform the requested operations, maximizing the usage of scheduled sessions.

```
# dsmc schedule

ADSTAR Distributed Storage Manager
Command Line Backup Client Interface - Version 1, Release 3, Level 0.0
(C) Copyright IBM Corporation, 1990, 1994, All Rights Reserved

Please enter password for node "ORI":

Session established with server ADSM: AIX-RS/6000
  Server Version 1, Release 3, Level 0.0
  Server date/time: 11/10/1994 14:06:43  Last access: 11/03/1994 15:49:08

Querying server for next scheduled event.
Next operation scheduled:
------------------------------------------------------------------------
Schedule name:         TEST3
Action:                Selective
Objects:               /home /var
Options:
Server Window Start:   14:30:00  on 11/10/1994
------------------------------------------------------------------------
Command will be executed in 35 minutes.
```

Figure 73. Client Scheduling Program Started in Client Polling Mode

The administrator can set up the Central Scheduler for client-polling mode, server-prompted mode or any. In the first two cases, each client must select the same mode as the server in order to allow the successful execution of the scheduler program. The default client mode is polling. In order to change it to server-prompted mode, client users must add `SCHEDMODE PROMPTED` in the dsm.sys file.

To initiate a schedule, the client must start the client scheduling program by issuing the command `DSMC SCHEDULE`.

The client-scheduling program can be conditionally entered as part of the workstation startup procedure. The program asks the user to enter the ADSM node password and starts the communication with the server.

If the current mode is client polling, the program will start querying the server and executing schedules until the client user explicitly stops the program or the machine is shut down.

In server-prompted mode, the workstation is simply waiting to be contacted by the server.

```
# dsmc schedule

ADSTAR Distributed Storage Manager
Command Line Backup Client Interface - Version 1, Release 3, Level 0.0
(C) Copyright IBM Corporation, 1990, 1994, All Rights Reserved

Please enter password for node "PIPPIN":

Session established with server ADSM: AIX-RS/6000
```

```
  Server Version 1, Release 3, Level 0.0
  Server date/time: 11/10/1994 13:09:14   Last access: 11/10/1994 13:07:39

Querying server for next scheduled event.
Next operation scheduled:
------------------------------------------------------------
Schedule Name:         TEST1
Action:                Incremental
Objects:               /tmp
Options:
Server Window Start:   13:00:00  on 11/10/1994
------------------------------------------------------------
Executing scheduled command now.
Session established with server ADSM: AIX-RS/6000
  Server Version 1, Release 3, Level 0.0
  Data compression forced on by the server
  Server date/time: 11/10/1994 13:09:16   Last access: 11/10/1994 13:09:14

Incremental backup of file system: '/tmp'
Normal File-->         167,936 /tmp/4495_art.doc  Sent
Normal File-->               0 /tmp/DKLoadLog  Sent
Normal File-->          35,754 /tmp/fsf_dmn.err.log  Sent
Normal File-->           9,237 /tmp/lslppL.out  Sent
Normal File-->              74 /tmp/new_file  Sent
Normal File-->           6,485 /tmp/package.out  Sent
Normal File-->              46 /tmp/rc.net.out  Sent
Normal File-->         385,759 /tmp/screen.dmp  Sent
Normal File-->              22 /tmp/tempfile17730  Sent
Normal File-->               0 /tmp/xlogfile  Sent
Successful incremental backup of '/tmp'

Total number of objects inspected:       27
Total number of objects backed up:       10
Total number of objects updated:          0
Total number of objects rebound:          0
Total number of objects deleted:          0
Total number of objects failed:           0
Total number of bytes transferred:     70.2 KB
Data transfer time:                    0.04 sec
Data transfer rate:                1,598.53 KB/sec
Average file size:                     59.1 KB
Compression percent reduction:        88.12%
Elapsed processing time:            0:00:05
Scheduled event 'TEST1' completed successfully.
Sending results for scheduled event 'TEST1'.
Results sent to server for scheduled event 'TEST1'.

Session established with server ADSM: AIX-RS/6000
  Server Version 1, Release 3, Level 0.0
  Data compression forced on by the server
  Server date/time: 11/10/1994 13:09:23   Last access: 11/10/1994 13:09:16

Querying server for next scheduled event.
Next operation scheduled:
------------------------------------------------------------
Schedule Name:         TEST1
Action:                Incremental
Objects:               /tmp
Options:
Server Window Start:   13:00:00  on 11/11/1994
------------------------------------------------------------
Waiting to be contacted by the server.
```

Figure 74. Client Scheduling Program Executed in Server Prompted Mode

5.2.3 Backup/Restore Operations

The backup process creates a copy of a client file on the ADSM server and also of the directory in which the file resides. The files are backed up according to policies that the administrator has predefined. The policies state, for example, how many backup versions should be retained in the storage pools, how long to retain them and whether to backup files that are currently in use.

The client user may choose between three backup modes:

selective — A selective backup copies all selected files to the server, whether they are changed or not. Wild card characters may be used to match files for backing up

incremental — Incremental backup copies only new or changed files to the server. To find out whether a file has changed, ADSM compares the file residing on the workstation with its latest copy at the server, including a set of file attributes. Notice that the first incremental backup is a “full backup” because all files are identified as new

incremental by date — An ADSM Version 2 enhancement, incremental backup by date supplements the capabilities of standard incremental backups. When this feature is used, incremental backup performance is improved, since a query of the server for every active file and the subsequent building of a directory tree in real memory is no longer necessary. Instead, the server is only queried for the date of the last incremental backup and this is compared with client file and directory dates. Note that this will not catch all changed files. Those files whose attributes only have been changed or files with an older modification date that were recently copied, for example, will not be backed up this way

ADSM includes options that control processing for user sessions. These options are contained in three files:

- Client system options file (dsm.sys)
- Client user options file (dsm.opt)
- Include/exclude file

For a description of the first two files, refer to 3.1.2.5, “Customizing Additional Client Options” on page 37. The root user can create an include/exclude file to list the files or groups of files in the file system to be included/excluded for backup services and to associate specific files with different management classes.

5.2.3.1 Backup

A client user can backup/restore files using the GUI or the CLI. In this section, the GUI will be referred to as it makes it easier to show the different backup facilities. For a complete description of the subject, please refer to the *ADSM User's Guide and Reference for UNIX*.

From the GUI, select **Backup**: there are four possible choices:

- Incremental
- Incremental by date

- Selective by file specification
- Selective by directory tree

Use selective backup modes when a piece of work needs to be saved the instant that it is finished. For backup of a group of files with similar naming conventions, choose **backup by file specification**, this allows the inclusion of subdirectories.

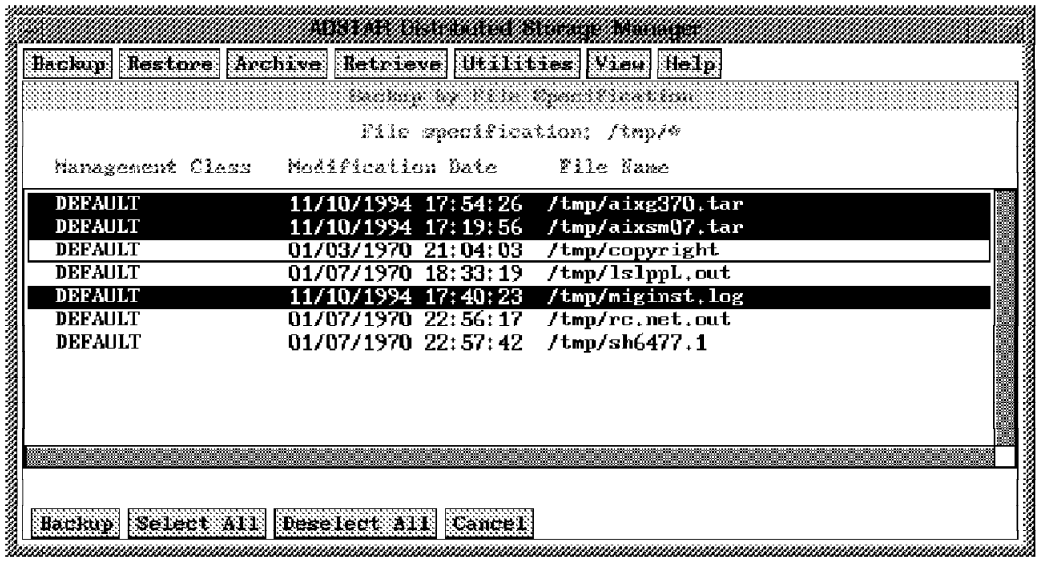

Figure 75. Backup by File Specifications

ADSM allows a list of files to be assembled that are to be backed up from one or more directories. In this operation, all of the files selected are backed up even if identical versions of the file already exist in ADSM storage.

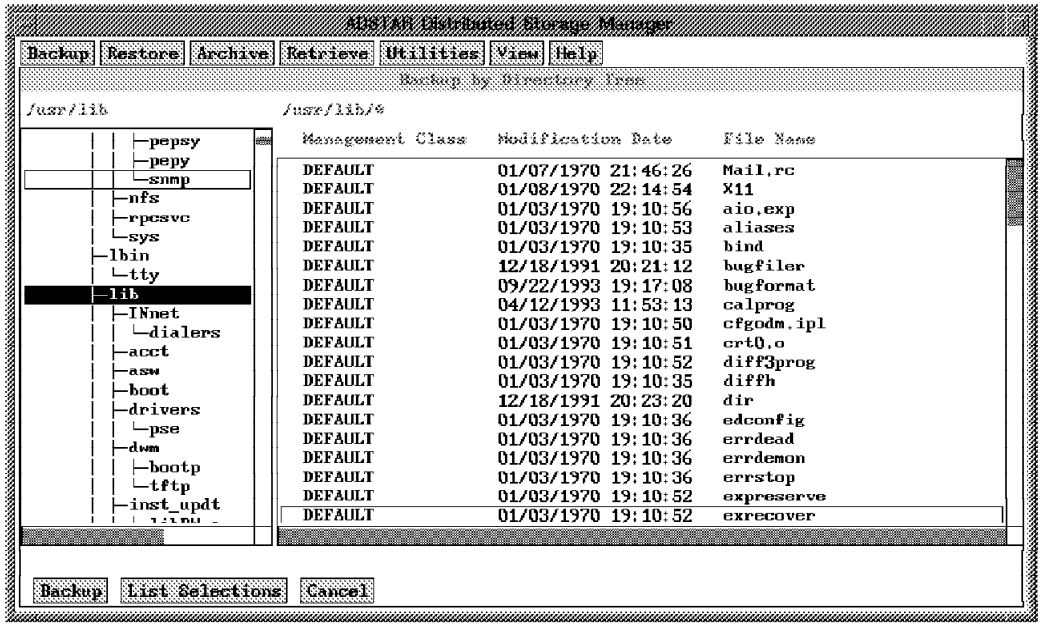

Figure 76. Backup by Directory Tree

The most recent backup version of a file is called the *active* copy. Anytime a file is deleted from a workstation, the active copy is marked inactive the next time an incremental backup of the file system is run. When the incremental backup is selected, ADSM checks all the files in the specified file system and backs up only those files that are new or have been changed from the last backup. Incremental by date compares only the date of the last backup with the dates of the directories or files to be backed up.

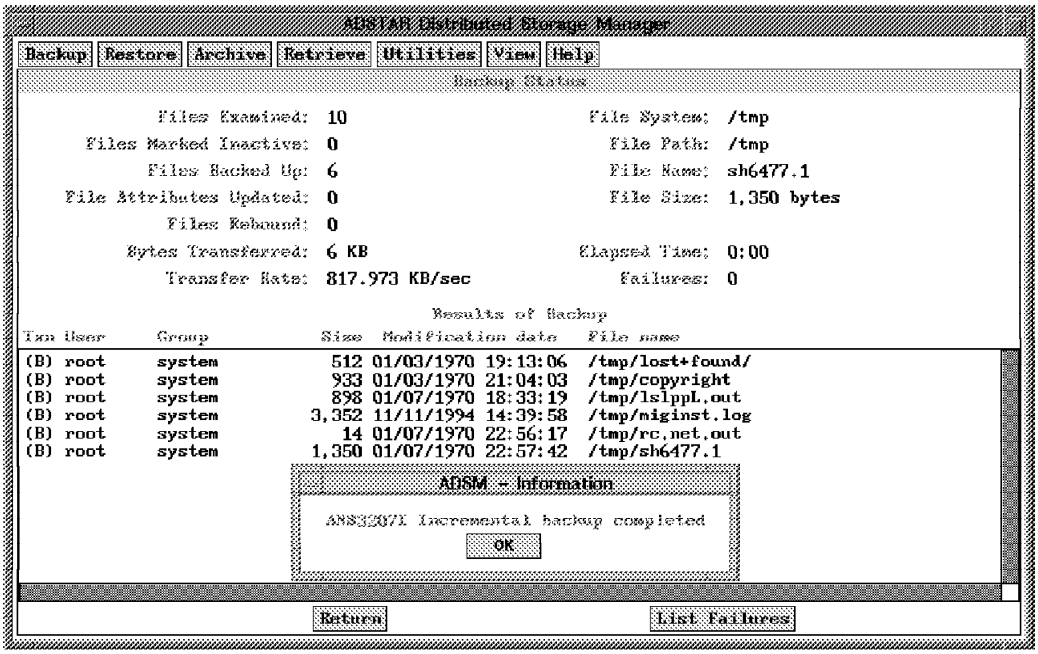

Figure 77. Incremental Backup

Also consider using the CLI for incremental backups, including the command in a procedure used to automate some steps of daily work. The command is `DSMC INCREMENTAL`. Parameters can be entered by hand or set in the options file.

5.2.3.2 Restore

The restore function copies a backup version of a file stored at an ADSM server to a workstation. It does not change the state of the file on the server. Note that as ADSM maintains a single backup image of a clients backed up data (rather than maintaining each full backup and incrementals separately), a full restore only involves copying back the single image rather than applying the full backup followed by the incrementals.

By default, restore recalls only active copies. If it should be necessary to select an inactive copy, the default must first be changed by selecting **Show active and inactive files** after clicking on the **View** button. ADSM will display a warning message, and the restore operation can be continued.

From the GUI, click on the **Restore** button. There are three possible choices:

- Restore by file specification

  Select restore by file specification when it is necessary to restore a single file whose name is known or multiple files with similar names (wild card characters can be used)

- Restore by directory tree

  Restore by directory tree allows specific files from a directory to be selected for restoring. The display is similar to the one shown in Figure 76 on page 146

- Restore subdirectory path

  Restore subdirectory path restores all files from a directory and any subdirectories

A *collision* policy should be defined. Collisions occur whenever a file being restored already exists on the workstation. There are three possibilities:

- Prompt on collision
- Restore only non-collisions
- Restore all with overwrite

Different destination paths can also be selected for the files being restored. The options are selected from the Restore/Retrieve Parameters window, shown by ADSM after the restore button has been clicked on.

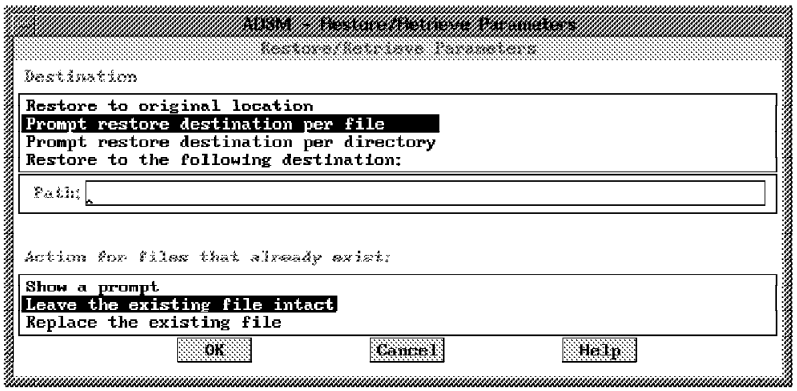

Figure 78. Restore Parameters

ADSM can restore files backed up by one client node to a different client node. Clients can be running the same, or different, operating systems. In order to perform this task, the owner of the backed-up files must explicitly authorize other client nodes to restore the files.

The owner of the backed-up files may select the **Set authorization** panel from the Utilities menu and fill in the requested information:

- Nodes and/or users to be authorized
- The directories and files that they can manage
- The type of authorization (backup and/or archive)

In order to recover another user's backup copy to the local file system, select the **Set user/node name for Restore/Retrieve** panel in the Utilities menu and enter the user and node name of the file's owner. The normal steps can now be followed to restore the files.

5.3 File Storage Facility/6000

The FSF product is not designed to be a backup/restore product. However, because FSF copies the files in the cache to the NFS store directory, a form of backup is done automatically. Also, the NFS store should be in a regular backup schedule as with any other NFS directory.

Recovery is a simple process. If the local FSF file system needs to be recovered, a new FSF file system can easily be created using the Incorporating existing files into FSF from server store menu. If the NFS server store directory becomes corrupted, restore it from the server backups and continue using the existing FSF.

When FSF operates with ADSM, the FSF server store is an ADSM filespace, defined in a ADSM storage pool. In this case, ADSM will be performing standard backups of the FSF server storage.

5.4 UniTree

The scope of UniTree is to manage file migration in a two-level storage hierarchy and not to be used as a backup/restore product. However, migration may be set up to occur frequently enough to copy all of the disk files to the second-level storage (tape or optical). In this case, the migration can be considered to be a form of backup facility and the archive device as the backup of the disk data. If the disk server logical volume fails, all the files can be restored on a new logical volume, using *caching*.

The UniTree file system also provides a *trash can* subdirectory in which all the user deleted files are stored for a specified time and from where they can be retrieved in order to recover from user errors. Also, *versioning* allows a user to maintain multiple versions of the same file in their directory, providing useful protection against loss of user data.

5.5 Legato NetWorker

NetWorker is designed to provide an automatic backup and restore capability for user files across a network of heterogeneous systems. It can perform different levels of backup and provides the ability to create customized backup schedules in addition to four predefined schedules you can use as samples to modify or as they are, to implement network-wide automatic backup.

5.5.1 Backup Types

NetWorker supports four kinds of backups:

Full	All files are backed up
Level (1-9)	Only the files that have changed since the last lower level backup are backed up. Since full backup can be considered as level 0, this means that, for example, a level 2 backup backs up all files changed from the previous level 1 backup or from the last full backup
Incremental	Only the files that have changed since the last backup are backed up
Skip	No files are backed up

The different backup levels provide a most flexible instrument for setting up a backup schedule by matching requirements in terms of performance and availability.

Full backups provide the easiest way to recover from disk crashes, but they take more time to execute and cause the online indexes to grow more rapidly than with other backup types.

If a full backup is performed once a week, for example on Friday, and incremental backups for all the other days, both backup time and index space are minimized. On the other hand, if a disk crash should occur on Thursday, for example, it may be necessary to use all of the backup tapes to recover (the full backup plus all the incremental backups).

Level 1-9 backup types allow an intermediate solution to be designed. A level backup copies all files that have been changed since a lower-level backup. In this backup scale, full backups are considered to be the lower level, incremental backups the higher level and 1-9 the intermediate levels.

Using the previous example, a full backup should still be performed on Friday and incremental backups the other days, but a level backup can be substituted for the incremental backup on Tuesday. In this way, if a disk crash occurs on Thursday, only the following tapes will be required:

- The Friday full backup
- The Tuesday level backup
- The Wednesday incremental backup

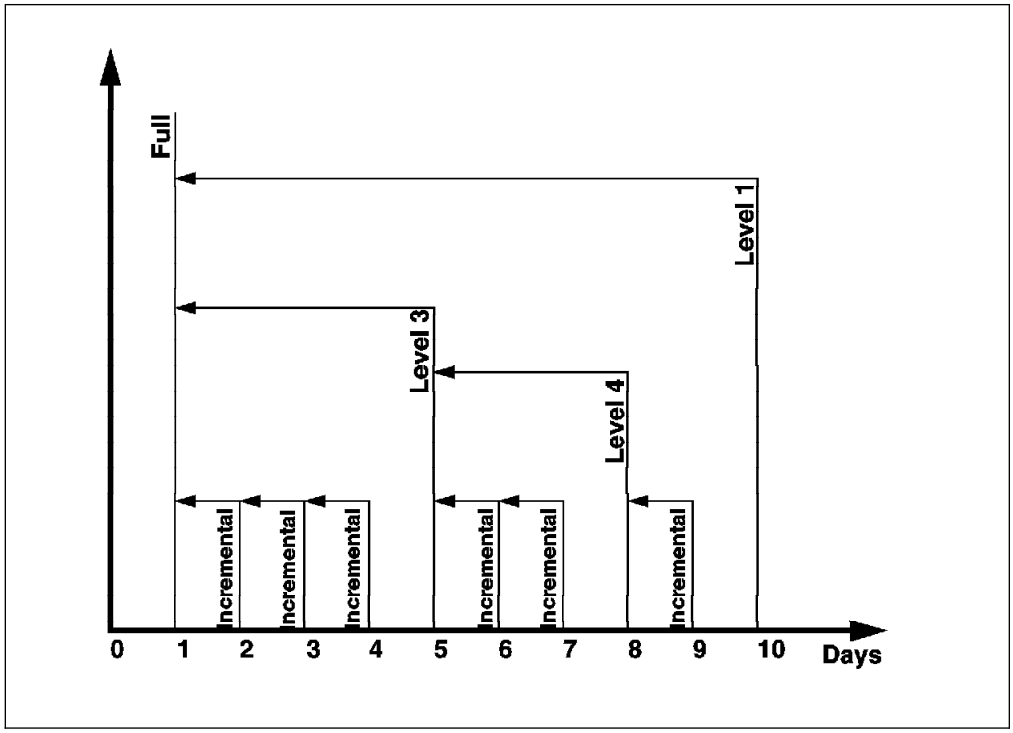

Figure 79. NetWorker Level Backups

The NetWorker GUI offers an easy and flexible instrument for defining customized backup schedules.

5.5.2 Defining Schedules

The server backs up each client in the network according to a backup schedule. NetWorker provides five predefined schedules which can be used as they are, or customized schedules can be defined using the Schedule window in the Administration menu.

The predefined schedules are:

- Default (full every Sunday)
- Full on First of Month
- Full Every Friday
- Quarterly
- Full on First Friday of Month

They are described in details in Chapter 4 of the *Legato NetWorker Installation and Maintenance Guide*

5.5.2.1 Creating New Schedules

This section looks at how a new schedule is created:

- Click on the **Create** button in the Schedule window
- Enter the name of the new schedule
- Enter the period, noticing that:

  week Any backup selection made for a particular day will apply to the same day for every week of every month

  month Any backup selection made for a particular day will apply to the same calendar day for every month

- Place the cursor on a calendar day, and select it; the Backup Level menu will appear, from which the backup level can be selected
- Make the changes required using the `Overrides` feature; this only allows the application of choices to the specified day.

5.5.2.2 Schedule Example

As an example, a new schedule will be defined in which: a full backup is performed on the first Monday of every month, a level 9 backup is performed on the other Mondays, an incremental backup is performed on every other day.

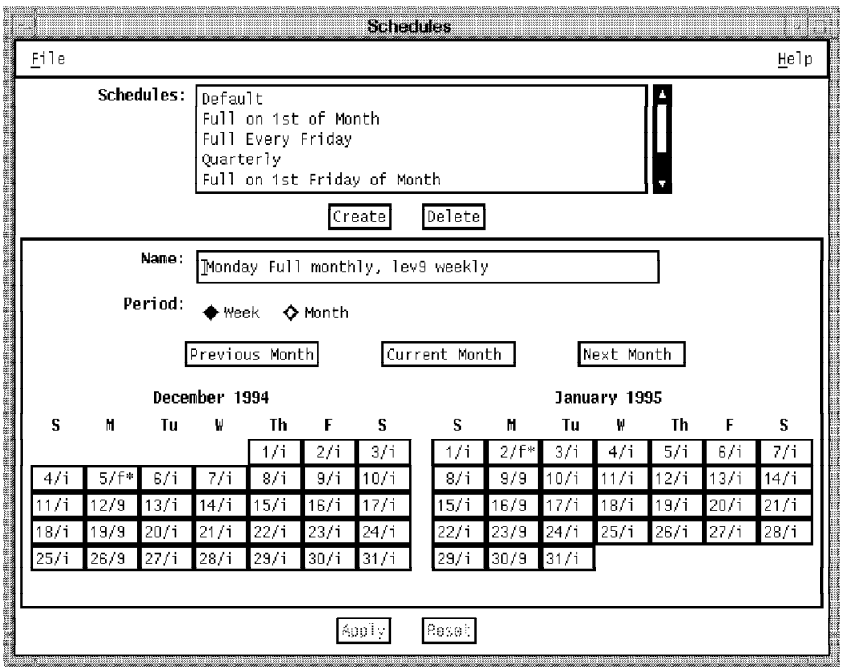

Figure 80. Networker GUI Administration: Schedules Window

1. Click on the **Create** button and enter the name for the new schedule: `Monday Full monthly, lev9 weekly`. The name cannot exceed 32 characters
2. Select **Week** as the period: a default configuration will appear
3. Select any Sunday in the current month and change it to `incr` (`skip` can be selected if an operator is required to manually mount tapes, and the site is unattended on weekends)
4. Select any Monday in the current month and change it to `9`; every Monday in the calendar will change accordingly
5. Go to the first Monday in the month and select **Overrides**. From the new menu, **full** can be chosen. This selection will apply only to the specified day; so this last step should be repeated for every first Monday in the calendar.
6. The Overrides menu may also be used to skip any holidays. Click on the **Apply** button when finished. The final result will resemble Figure 80

5.5.2.3 Defining Schedule Start Time

The server will execute schedules automatically for the specified clients, according to the instructions defined in the Groups window.

Select **Groups** from the Administration menu; it will look like Figure 81 on page 153. Click on the **Create** button to define a new group; the following information should now be entered:

Name The name of the group

Autostart Enable autostart to have NetWorker perform automatic backup at the specified time. The backup operation can also be started at this time, or disabled completely.

Start time Enter the time that the backup is to be started automatically

Client retries Use the sliding bar to select the number of times the server will retry the backup operation if the client does not initially respond

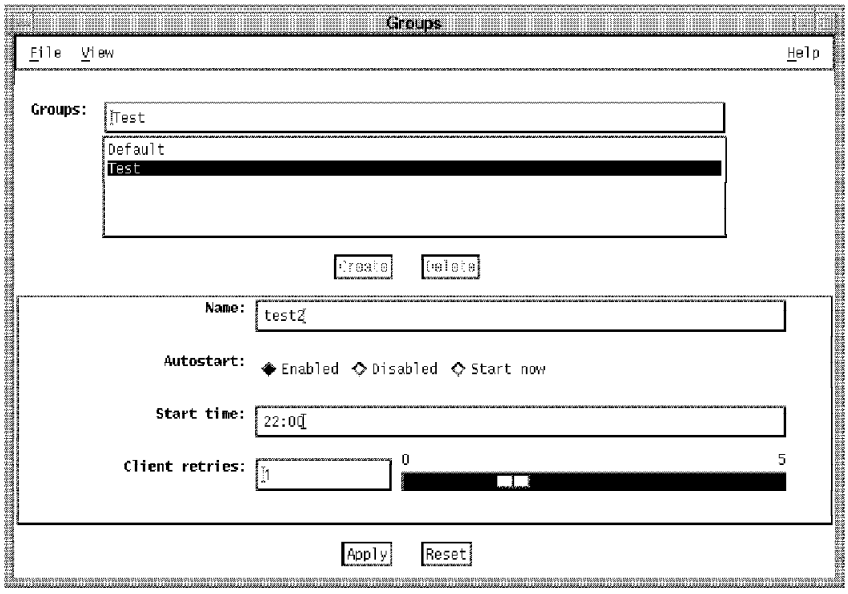

Figure 81. Networker GUI Administration: Groups Window

Click on the **Apply** button to create the new group.

Now select the **Client** window from the Administration menu, and for each client that is to be automatically backed up with one of the previously defined schedules at the specified time, select the corresponding **Group** and **Schedule** from the options displayed in Figure 66 on page 125.

5.5.3 Managing Online Indexes

When NetWorker backs up user data, it adds new entries to the online indexes: the newly saved file entries in the file index and the file location on the backup volumes in the media index. Indexes may be automatically managed using the Policies window from the Administration menu or manually by the administrator or the file owner, using the Indexes and Volumes windows from the Operation menu.

5.5.3.1 Index Policy Management

NetWorker automatically manages the contents of the indexes with policies that are specified for each client. The predefined policies may be used, or they may be updated to support specific requirements.

There are two types of policies required:

Browse This policy specifies how long backed-up file entries for a specified user will remain in the online file index in order to be available for browsing

Retention This policy specifies how long backed up file entries for a specified user will remain in the online media index in order to be available for restore

The browse policy period must be less than or equal to the retention period. When a user save set reaches the browse policy period, the corresponding entry is deleted from the Recover window; so it cannot be browsed anymore. It is still contained in the Volumes window, as `retrievable`.

When the save set reaches the retention policy period, the corresponding entry appears as `recyclable` in the Volumes window. Recyclable files are still recoverable as long as the backup volume is not relabelled.

The default policies are:

Decade	Ten years
Month	One month
Quarter	Three months
Week	One week
Year	One year

Customized policies can be created by selecting the **Create** button in the Policies window from the Administration menu.

The following fields should be updated:

Name	The name of the policy
Period	The unit period
Number	The number of unit periods for the policy

Click on the **Apply** button to add the newly defined policy. To assign the new policy to specific clients, select the **Clients** window from the Administration menu, find the client to be updated and choose the correct policy from the displayed list.

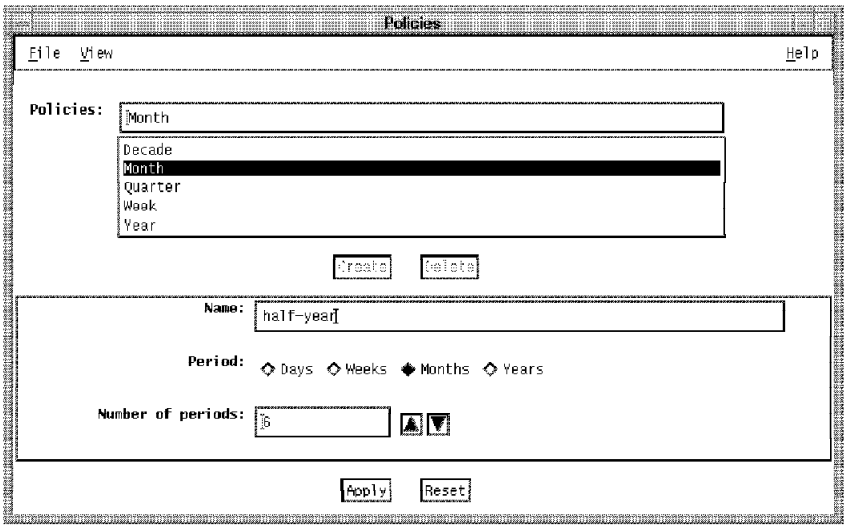

Figure 82. The Networker GUI Administration: Policies Window

5.5.3.2 Manual Index Management

Each time NetWorker starts a backup, it generates a save set for every backed up file system. To recover a complete directory, some files are needed both from incremental and full backups. When the system needs to remove a file set, it checks for all dependencies between the different backup levels of the same file system. No save set is removed until all of its dependent save sets have been removed.

Entries for save sets can be manually removed from the file index by using the Indexes window from the Operations menu. The display looks like Figure 83 on page 155.

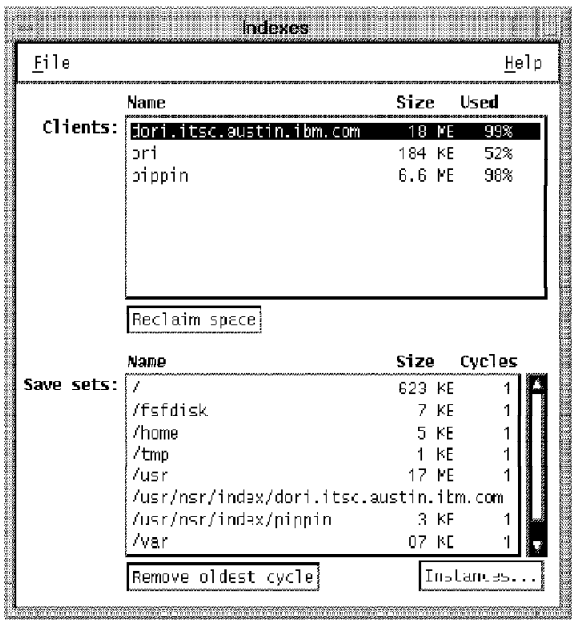

Figure 83. The Networker GUI Administration: Indexes Window

The Clients field allows selection of the specific client on which manual operations are to be performed. In the Save sets field, the save sets can be viewed relative to the selected client, together with the file index space used and the number of cycles. A backup cycle starts with a full backup and includes all the following higher-level backups until the next full backup.

The operations that may be performed are:

- Remove oldest cycle - removes all the index entries of the specified file system relative to the oldest backup cycle. If no save sets in the list are highlighted, the server will remove all the oldest cycles of all the save sets displayed
- Reclaim space - frees all the empty space in the client file index after save set removal. NetWorker reuses the empty space in the index after removal for the new backup entries, but does not free the space; it should be reclaimed manually from the Indexes window
- Instances... - views more detailed information about save sets. Click on the **Instances** button, move the new window aside and select the save set to be displayed from the Indexes window. The display will show the following:
  - Internal ID number for the save set
  - Number of files in the save set
  - Size of the save set
  - Date of the backup
  - Backup level

  The displayed instances can be printed or saved to a file using the **Print** and **Save** buttons respectively.

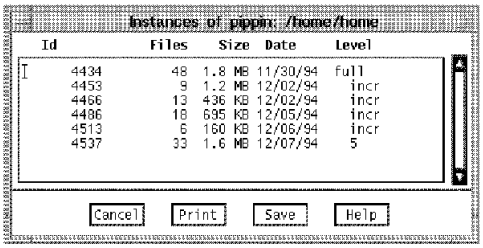

Figure 84. Networker GUI Administration: Instances Window

5.5.4 Managing Directives

NetWorker provides the additional capability of using directives when processing backups of client files.

In a typical directory, there will be several files that do not require saving, core and object files for example. File compression is also a good idea in order to save space on backup devices and reduce the network bandwidth.

There are four predefined directives that can be selected in the backup schedules. Customized directives may also be defined using the GUI Directives window from the Administration menu and referring to the `nsr` command man page for the correct syntax.

The four predefined directives are:

- Default
- Default with compression
- DOS standard directives
- NetWare standard directives

The predefined directives can be viewed by selecting **Directives** in the GUI Administration menu and then selecting **NSR directive** from the nsradmin visual screen. The display will be as in Figure 85 on page 157.

```
Command:  Select   Next   Prev   Edit   Create   Delete    Options   Quit
1 on 4 (on dori)

                        type: NSR directive;
                        name: Default;
                   directive: "
<< / >>
        skip: tmp_mnt
        +skip: core
<< /tmp >>
        skip: .?* *
<< /export/swap >>
        swapasm: .
<< /usr >>
        allow
<< /nsr/logs >>
        logasm: .
<< /var >>
        logasm: .
<< /usr/adm >>
        logasm: .
<< /usr/spool >>
        logasm: .

------------------------------
Keys: tab=next  return=do command  [a-z]=that command  H=help
```

Figure 85. NetWorker nsradmin Visual: NSR Directives

5.5.5 Using NetWorker

NetWorker services can be used by any user on a client machine previously defined to the server by the administrator. The users .profile files need to be updated by adding the path name of the NetWorker programs to the PATH variable. If AIXwindows is installed on the server machine, the command `networker &` can be issued to start the GUI. There are also line commands available that can be used instead of the GUI. They are listed in Appendix B of the *Legato NetWorker User's Guide*.

In any GUI window, there is a Help button available that will provide detailed information on specific functions.

The three main operations that can be performed are:

- Browse
- Backup
- Recover

They are detailed in the following sections, using GUI windows.

5.5.5.1 Browse

From the Operation menu, files may be browsed:

- From the Backup window, directories can be viewed
- From the Recover window, the indexes of saved files can be viewed

Both windows offer the same browsing capabilities. The default directory shown in the left part of the screen is the home directory. Clicking with the mouse on one of

the subdirectories in the left part of the screen will display the files belonging to that directory in the right part of the screen.

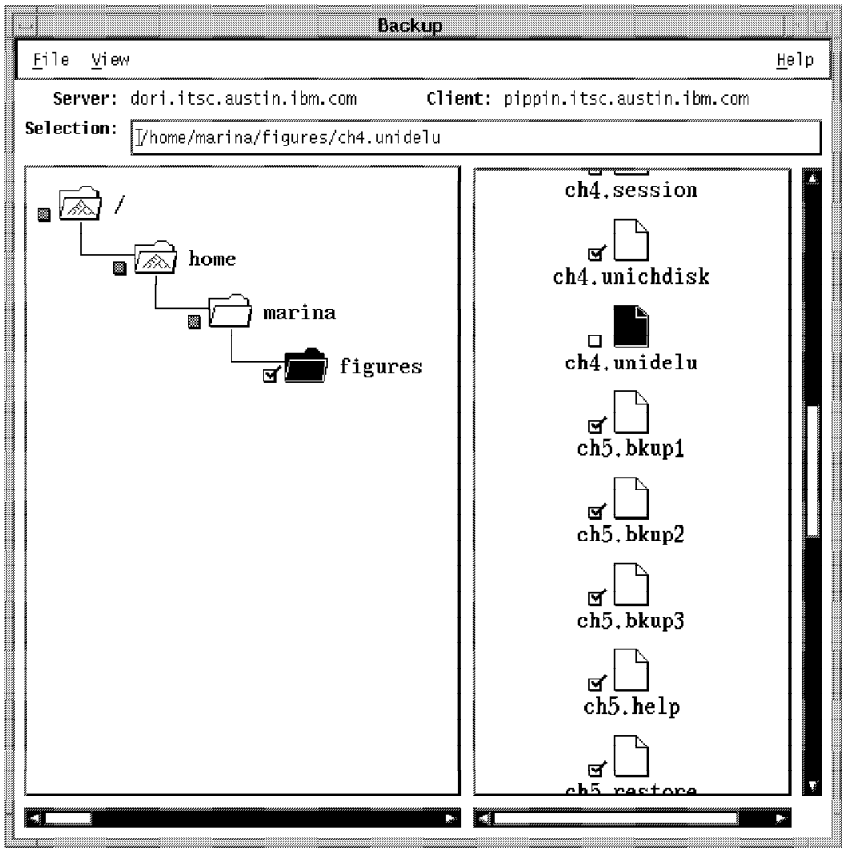

Figure 86. Networker GUI Operations: Browse from Backup Window

The highlighted directory can be changed by entering a different path in the Selection field.

From the View menu, it is possible to:

Expand one level View the next level of the target directory

Expand branch View the subdirectory tree of the target directory

Collapse branch Close all subdirectories and come back to the target directory

File details View the details of the files listed in the right part of the window

With the File menu, it is possible to:

Mark Specify a file or a directory in order to select it for backup or recover. If the highlighted icon is a directory, all its files and subdirectories are marked. If one of the file icons on the right side is also highlighted, only that file is marked for processing. It is also possible to use the mouse and click on the checkbox next to the file icon

Unmark Deselect a previously marked file or subdirectory

Search Look for a specified file or group of files specifying the pathname

5.5.5.2 Backup

After marking files from the Backup window, a user-initiated backup can be started by selecting **Start Backup** from the File menu.

The administrator should create a pool reserved for manual-type backups and label tapes accordingly if user backups need to be saved on separate tapes. Otherwise, any user-initiated backups are assigned to the default pool.

The Backup Options panel will appear as in Figure 26 on page 64. Make selections from the following options:

- Compress the backup (yes|no)
- Exclude any specific pattern from the backup

The progress of the backup can be monitored from the Backup Status window shown in Figure 27 on page 65.

Verification of a successful backup of the files can be effected using the Volumes window from the Operation menu.

Backup Device Problems: If a system failure occurs during a backup operation, the backup device could suffer I/O error problems and not be mountable the next time it is required as NetWorker checks the whole tape before mounting. In this case, the `scanner` command can be used to read the device contents. An example of the usage of this utility has already been shown in 4.4.4.1, “Recovering the /nsr Directory” on page 128. The output from this command can be printed and saved together with the tape.

Use other tapes from the same pool for all future backup operations. If it is necessary to recover one of the save sets contained in the damaged device, the `ssid` of the file system can be found from the printed list and the following command issued:

```
scanner -s ssid /dev/rmt0.1 | uasm -rv
```

substituting the actual device name for `/dev/rmt0.1`.

This command will recover all the selected save sets for the current workstation. During the recover process, NetWorker will prompt anytime it finds a file with the same name in the current directory. Refer to the man pages for more information about the `scanner` and `uasm` commands.

5.5.5.3 Recover

After marking files from the Recover window, the environment can be customized for the recover operation.

The View menu offers some additional functions:

Show marked Lists all the files that are marked for recovery

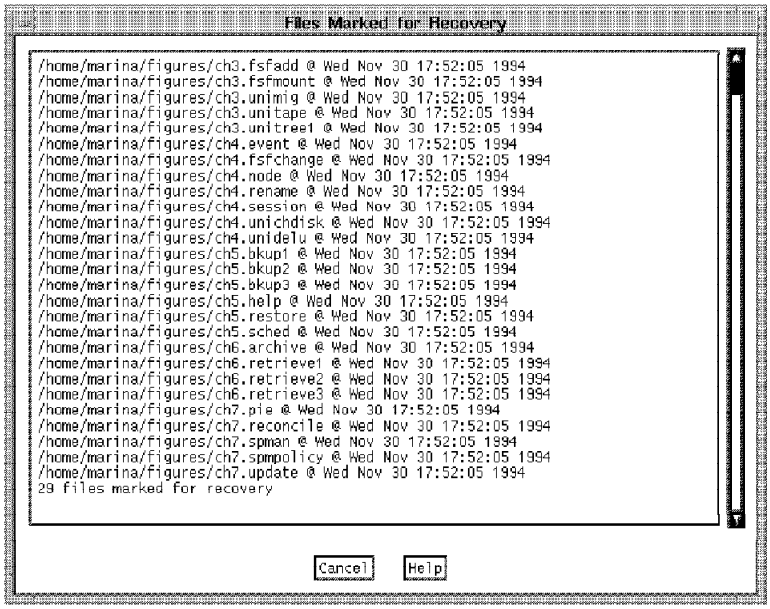

Figure 87. Networker GUI Operations: Recover - Show Marked Display

Versions Lists all existing versions of the selected files. This can be used before selecting browse again

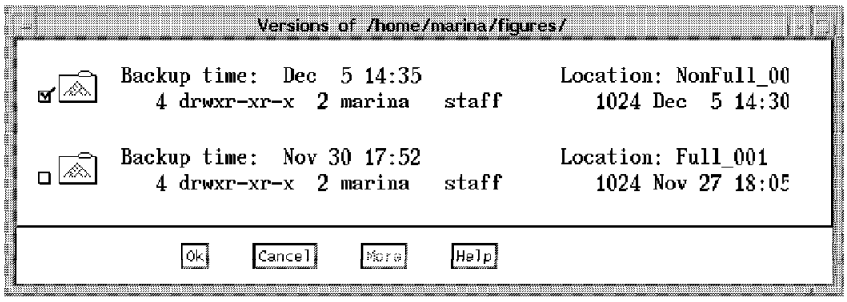

Figure 88. Networker GUI Operations: Recover - Versions Display

The File menu also offers some additional functions:

Change server Changes server if the default server is not the one that manages the files that are to be recovered

Change client Changes clients if files are being recovered from a client other than the local system. Authorization from the administrator is required to recover files for other clients

Change browse time Changes browse options if recovery of a specific file is required rather than just the last copy

Relocate Relocates files if the recovered files need to be placed in a directory other than the original one

Start recover Select to start the recover function

NetWorker will prompt with the Conflict Resolution window before starting the recover process. Select the action that should be performed when recovering a file with the same name as an existing one.

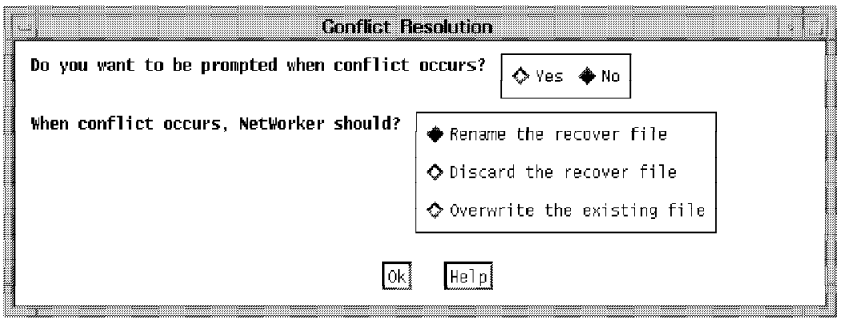

Figure 89. Networker GUI Operations: Recover - Conflict Resolution Display

The recover process can be monitored from the Recover Status window. For any additional information about the backup and recover process, refer to *Legato NetWorker User's Guide*.

5.6 Product Comparison Summary

The products discussed in this redbook will be compared with regard to backup and restore capabilities on the basis of the following criteria:

- Automation
- Function
- Ease of use

5.6.1 Automation

Automation is concerned with the degree to which the backup and restore functions provided by the products discussed in this document can be automated.

5.6.1.1 ADSM

ADSM provides a central scheduling service that allows for automatic backup, archive, retrieve, restore and scheduling of ADSM macros, client operating system commands and ADSM server commands. A scheduling window and duration can be defined, and the backup can be initiated by server prompt or client poll. Schedules can be periodic or at specific times.

5.6.1.2 FSF

FSF does not support backup and restore operations.

5.6.1.3 UniTree

UniTree does not provide backup and restore services.

5.6.1.4 Legato NetWorker

NetWorker provides a central scheduling service that will perform automatic backup operations periodically or at specific times. It is also possible to define the level of backup that will be performed at each occasion. Backup is always initiated by the server.

5.6.2 Function

This subsection is concerned with the functionality available from the backup services provided by the products discussed in this redbook.

5.6.2.1 ADSM

ADSM provides the following backup and restore features:

- Policy management

  Policies allow clients to be grouped together according to backup requirements. Within a policy domain, individual clients can be associated with particular management classes that define how data bound to a class will be backed up. Parameters include the destination storage pool, frequency of incremental backups, number of versions, retention period, mode of backup (modified or absolute), and how files in use will be treated.

- Scheduling

  Scheduling has been discussed in 5.6.1, “Automation” on page 161.

- Manual backup

  Client users can cause files to be backed up selectively or incrementally (changed files only). These operations can be performed from the user GUI or CLI.

- Automated backup

  Client users can create shell scripts that can be run from the command line to automatically back up files if required, though automation is more easily accomplished through central scheduling.

- Manual/automatic restore

  Files can be restored by name, directory tree, or subdirectory path. Initiation can be from the user GUI or CLI. Collision policies can be defined that govern behavior when a restore is done for a file that already exists. The central scheduler can also be used to automatically initiate restores.

  With the appropriate delegation of authority, files can be restored on behalf of other users and/or clients.

- Compression

  Compression of transferred data is optionally available.

5.6.2.2 FSF

FSF does not support backup and restore operations.

5.6.2.3 UniTree

UniTree does not provide backup and restore services.

5.6.2.4 Legato NetWorker

NetWorker provides the following backup and restore features:

- Backup types

  NetWorker supports several different types of backup, including full, level, incremental, and skip. This gives great flexibility in designing an appropriate backup strategy.

- Scheduling

This has already been discussed in 5.6.1, “Automation” on page 161.

- Index policies

  These govern how long a backed-up file will remain in the browse list for viewing and how long a file will be retained for restore selection. After retention period expiry, volumes are marked recyclable for reuse.

- Save set

  NetWorker maintains a save set that defines for each file system the links between full and incremental backups so individual directories can be easily restored.

- Directives

  Directives allow specific selection criteria to be associated with a backup. Thus, core files or tmp directories can be marked for avoidance.

- Manual backup

  Users may use the GUI or character-based interface to select files and/or directories for backup.

- Manual restore

  Previously backed-up files can be browsed and selected by version (if used) for restore. Collision processing is supported.

- Compression

  Data compression is optionally available.

5.6.3 Ease of Use

This subsection discusses the relative ease of use of the products supporting backup and restore.

5.6.3.1 ADSM

ADSM allows backups to be scheduled centrally; so the process can be completed automatically without user intervention. Client workstations must have the client scheduling code active, however.

Both GUI and CLI are available for backup and restore operations at the client. Selection of files to backup or restore is a simple process from the GUI, and command line shell scripts can be used to provide a simple automatic process.

5.6.3.2 FSF

FSF does not support backup and restore operations.

5.6.3.3 UniTree

UniTree does not provide backup and restore services.

5.6.3.4 Legato NetWorker

NetWorker allows central scheduling that can complete automatically, without user intervention. Directives simplify the process of file selection.

User selection of files for backup and restore is also a simple process via GUI or full-screen character-based interface.

Chapter 6. Archive and Retrieve

Archiving is often considered to be a particular type of backup technique. However, the underlying reasons for the two operations are different. In fact, while backup is mostly directed at providing data availability against potential failures, archiving is aimed at furnishing a long term, less expensive, more reliable storage for user data.

Archiving can also be required for legal and historical purposes and is intended to last for decades. In addition, it provides the capability of storing files that will not be referenced in the near future, thus allowing them to be erased from the workstation to create space for more active files and applications.

6.1 ADSTAR Distributed Storage Manager/6000

Archive services are requested when the contents of files at a certain point in time need to be saved (“snapshots”) or when copies need to be placed on long term storage so the originals can be erased from the workstation disk.

Usually, file archive is less frequent than backup. ADSM provides automatic archive services that the administrator can set up using the ADSM Central Scheduler.

The archive process creates a copy of client file(s) on the server. As with backup, archived files are managed on the basis of policies; however, this function does not have a concept of versioning. Multiple versions of the same file can be archived with each archived copy treated as a separate entity. Archive copies of a file are never replaced with a more current version, but are preserved exactly as they are stored. A description of the archived file can also be saved in order to make it easier to retrieve the correct version, where multiple copies of the same file exist.

When a file is archived, ADSM saves fully qualified path information with the file, together with its access permissions.

6.1.1 Archive Process

From the Backup/Archive Client GUI, select the **Archive** menu; there are two possible choices:

- Archive by file specification

  Files selected by matching a file specification pattern

- Archive by directory tree

  Files from a selected directory or subdirectory

After having selected the archiving mode and the files to be archived, just click on the **Archive** button, causing the Archive Option window to appear. This is where a description for the file can be entered. It can be used to store a brief explanation of what the file contains to assist in remembering its purpose. With ADSM Version 2, it is also possible at this point to request that the original copy of the file at the client be deleted after it has been successfully archived.

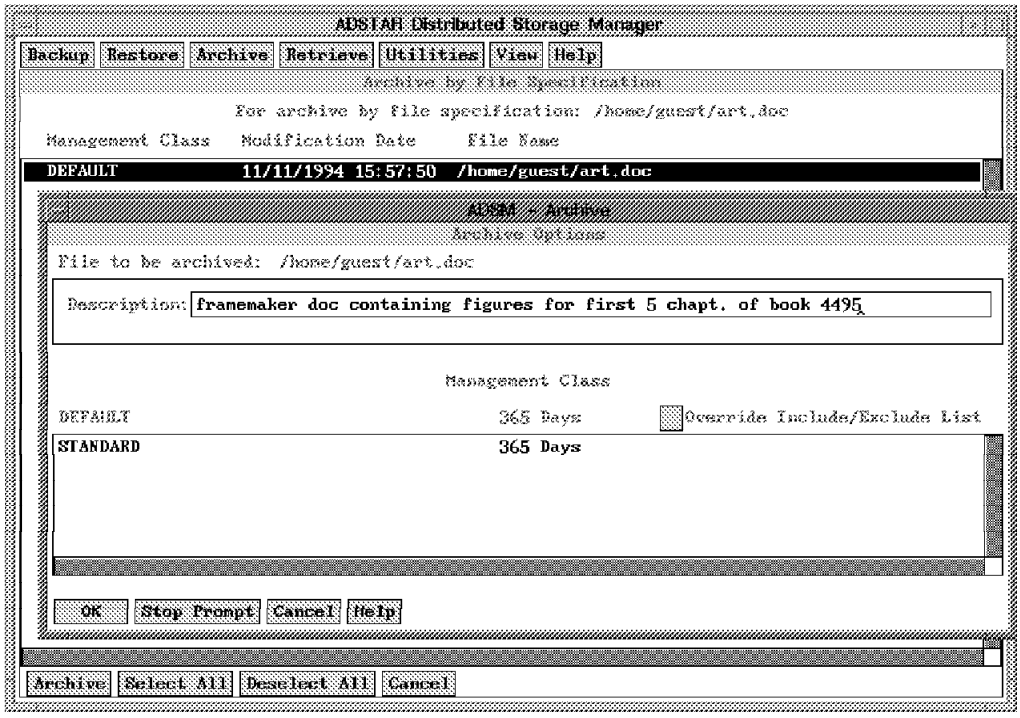

Figure 90. Archive by File Specification

It is also possible to enter the same keyword for each file of a group of related files. In this way, all of the related files can be retrieved together by specifying the keyword. If the same description is to be used for all files being archived at this time, select the **Stop Prompt** button from the Archive Option window.

A different management class can be assigned to the file. This can be done simply by selecting the **Override Include/Exclude List** and choosing among the displayed classes.

6.1.2 Retrieve Process

The retrieve function copies an archived file from the server to the workstation. ADSM does not change the file on the server.

From the Backup/Archive Client GUI, select the **Retrieve** menu; there are two possible choices:

- Retrieve by file specification
- Delete archive files

6.1.2.1 Retrieve Archived Files

When retrieve by file specification is selected, ADSM displays the Retrieve Scope window.

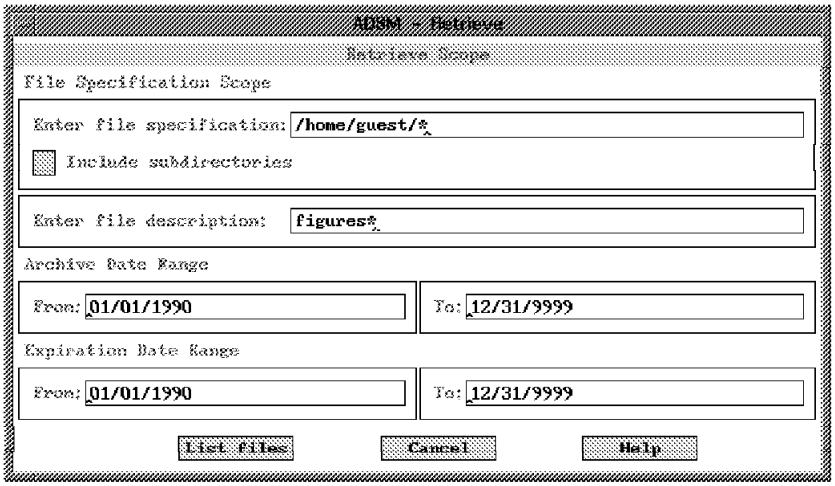

Figure 91. Retrieve Scope Window

Here, the files to be retrieved can be selected using four different parameters:

1. The file specification. It must contain the complete path name starting from the root directory and must end with the file name. The file name can contain wild card characters
2. The file description. A group of files having the same description can be selected. This can also contain wild card characters
3. The archive date range
4. The expiration date range

After making the choices, clicking on the **List All** button causes the system to display the Retrieve from Archive window, where all the selected files can be viewed and final selection made.

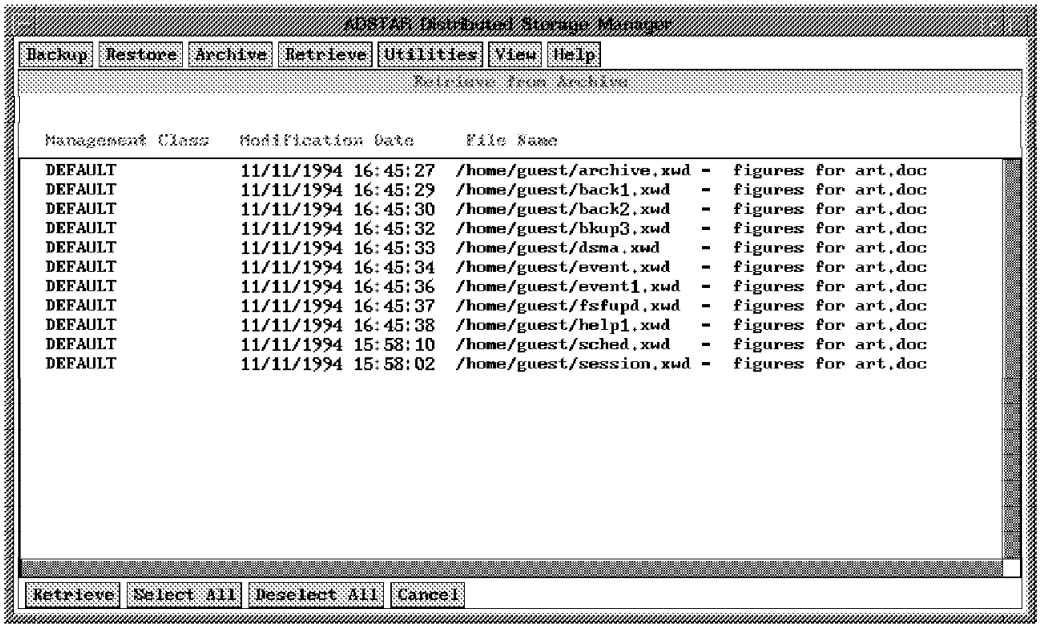

Figure 92. Retrieve from Archive Window

After this, the ADSM Restore/Retrieve Parameters window will appear, as shown in Figure 78 on page 148.

During the retrieve operation, the system displays the Retrieve/Archive Delete Status window, showing all the retrieved files with their characteristics.

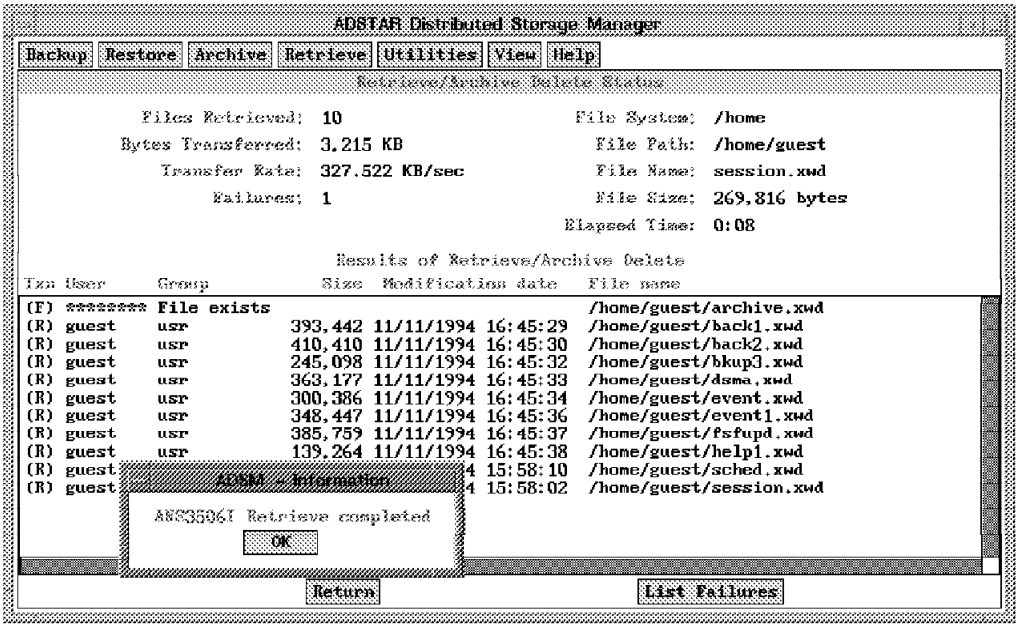

Figure 93. Retrieve/Archive Delete Status Window

6.1.2.2 Delete Archived Files

Archived files can only be deleted if the ADSM administrator has given archive delete authority to this client node, though the administrator can delete the client filespace. This is another security feature of ADSM, protecting archived copies stored for legal or historical purposes from being deleted by unauthorized people.

When the **Delete archive files** option is selected from the Retrieve menu, the Archive Delete Scope appears, looking like the Retrieve Scope window shown in Figure 91 on page 167. The delete process follows the same rules as the retrieve process. Refer to *ADSM User's Guide and Reference for UNIX* for any additional information.

6.2 File Storage Facility/6000

FSF/6000 does not provide archive and retrieve services.

6.3 UniTree

UniTree does not provide archive and retrieve services.

6.4 Legato NetWorker

NetWorker does not provide specific utilities for archiving with the release used in this redbook. With the new V4.1, however, a separate archive application is provided that fully supports archiving. The older version does offer the possibility to setup an elementary archive process in order to store data on separate long-term devices.

NetWorker already provides a pool with a corresponding label template for archive. The steps that need to be performed in order to set up an archive process are:

1. Create a group for archive processing
2. Create a schedule for archiving. For example, start an automatic archive process on the last Friday of every month. Set all of the other days to `skip` because incremental backups are not required, only full backups on the specified days
3. Enable the archive pool for the defined group
4. Choose the appropriate policies for the retention period of the archive data; the long term defaults of `Decade` or `Year` can be used or new policies created for specific requirements
5. For every client that will require archive capability, select the **Client** window. Add the client entry a second time to the server list of clients, and customize it according to the following guidelines:
   - Add the archiving schedule
   - Specify the browse and retention long term policies
   - Add the group for archive
   - Specify the save sets that are to be regularly archived, removing the `all` entry

6.5 Product Comparison Summary

The products discussed in this redbook will be compared with regard to archive and retrieve capabilities on the basis of the following criteria:

- Automation
- Function
- Ease of use

6.5.1 Automation

Automation is concerned with the capability of a product to provide services without user intervention.

6.5.1.1 ADSM

In common with backup and restore services, discussed in 5.6.1, “Automation” on page 161, ADSM provides central scheduling for archive and retrieve of client data.

6.5.1.2 FSF

FSF does not provide archive/retrieve services.

6.5.1.3 UniTree

UniTree does not provide archive/retrieve services.

6.5.1.4 Legato NetWorker

Strictly speaking, NetWorker does not provide archive/retrieve services though it is possible to use backup to a specific device for a similar purpose. In this case, the scheduling mechanism discussed in 5.6.1, “Automation” on page 161, can be used to automate the process.

6.5.2 Function

This subsection compares the functionality available from the product that support archive/retrieve services.

6.5.2.1 ADSM

ADSM provides the following functionality for archive/retrieve:

- Policy management

  In common with backup/restore, policies apply to archive services too. These were discussed in 5.6.2, “Function” on page 162. The only exception is that versioning is not supported for archive.

- Scheduling

  This was discussed in 5.6.1, “Automation” on page 161.

- Manual archive

  Files can be selected from the GUI by specification or by directory tree for archive. The CLI can also be used for file archive.

- Manual retrieve

  Files can be selected for retrieve using the GUI on the basis of file specification. Files can also be deleted from archive here. The CLI can also be used for file retrieval/deletion.

- Compression

  Compression of archived data is supported at the client for transmission to the server.

6.5.2.2 FSF

FSF does not provide archive/retrieve services.

6.5.2.3 UniTree

UniTree does not provide archive/retrieve services.

6.5.2.4 Legato NetWorker

NetWorker provides the following archive/retrieve functionality:

- Scheduling

  This was discussed in 5.6.1, “Automation” on page 161.

- Index policies

Similar to backup processing, the retention period for archive data can be set as well as the length of time that an archived file will appear in the browse list.

- Manual archive

  Files can be selected for full backup from the GUI, using the appropriate archive index policy. Once the file is deleted, this roughly equates to archive.

- Manual retrieve

  Files can be retrieved using the GUI or full screen character-based interface in the same fashion as backed up files.

- Compression

  Again, compression can be used for transmission of data to the server.

6.5.3 Ease of Use

This subsection discusses the relative ease of use of the archive/retrieve services provided by the products discussed in this redbook.

6.5.3.1 ADSM

Files can be easily selected for archive or retrieve from the user GUI. The CLI also provides this function. Different management classes can be associated with specific files as they are selected, if required.

Using central scheduling, files can be automatically archived and retrieved without user intervention.

6.5.3.2 FSF

FSF does not provide archive/retrieve services.

6.5.3.3 UniTree

UniTree does not provide archive/retrieve services.

6.5.3.4 Legato NetWorker

As archive is really backing up of the files with a specialized index policy and storage pool, it is as easy to use as backup services. Deletion of the local client copy, however, will have to be effected manually.

Chapter 7. Space Management

Space management is the process of keeping sufficient free storage space available for users on client systems and making the most efficient and economical use of distributed space resources. It prevents the occurrence of out-of-space conditions on user file systems and increases the transparency of the underlying distribution of file resources in a networked environment.

This is usually implemented by using the client disk as a cache and maintaining the full information space at a server. When a client application requests data that is not in the client cache, it is transparently copied from the server to the cache.

This automatic mechanism will reduce or eliminate the need for manual storage management activities on the client, reduce the disk cost per client, and provide a disk capacity based on a hierarchy of storage devices which is only limited by the server configuration.

7.1 ADSTAR Distributed Storage Manager/6000

Version 2 of ADSM will include a space management function integrating new services with those already provided by the backup and archive functions:

migration The process of moving infrequently used or large files from user workstations to the ADSM server storage

recall The process of transparently bringing back migrated files to the local client storage when required by a user application.

A read-only process can read a migrated file directly from the server store without causing a recall

The system administrator can set up space management policies and manage the file migration/recall process. In addition users at the client node can start selective file migration or recall for their files. To perform these tasks, a new Graphical User Interface is also available, as shown in Figure 2 on page 21.

7.1.1 Space Management Services

This subsection covers the services provided and the functional characteristics of the space management component of ADSM.

7.1.1.1 Migration

ADSM migrates files from the local file system by copying them to the server storage and replacing the original copy with a *stub*. In the server storage pool, the contents of the file are stored in a *bitfile*. The stub file identifies its corresponding bitfile. The actual size of a stub file can be defined. The first few bytes contain required control information, the rest of the file can contain leader data, which is as much of the original file data that will fit into the remaining space in the stub file. The minimum size of a stub file is 511 bytes, the default under AIX is 4095 bytes. As a result of storing as much leader data as possible in the stub, processes that only require access to the first few bytes of data or just to the file attributes, such as `ls`, will not require recall of the original file.

ADSM can manage three different types of migration:

- Threshold migration - ADSM automatically begins to migrate files when the local space usage reaches a specified high threshold and goes on until the used space drops to a specified low threshold.

  This process can be started manually by the root user, with the command `dsmautomig`, independently from the high threshold value, but it will still stop when the local space reaches the low threshold.

- Demand migration - ADSM automatically begins to migrate files when an out of space condition occurs. This migration begins before users or applications are aware that an out-of-space condition is imminent. The daemon will migrate files until the current space utilization requirement is met. ADSM cannot manage out of inode conditions.

- Selective migration - ADSM starts to migrate specified files when the user submits the `dsmmigrate` command. Selective migration can also be initiated from the GUI.

7.1.1.2 Premigration

After threshold migration is complete and the cache utilization has returned to below the low water mark, ADSM starts preparing files for migration, sending a copy to the server store without removing the file from the local system. This is called premigration. Premigrated files allow ADSM to quickly release occupied space before threshold or demand migration processes run. In order to free up space, all that is required to happen for a premigrated file is that the cached copy be deleted and replaced with a stub file.

The root user can specify the percentage of local space to be used by ADSM for premigrated files; the default is the difference between the high and low cache utilization thresholds.

7.1.1.3 Reconciliation

At specified intervals, the space monitor daemon runs the reconciliation process in order to monitor and resynchronize the status of local and remote files and to build the new list of candidate files for migration.

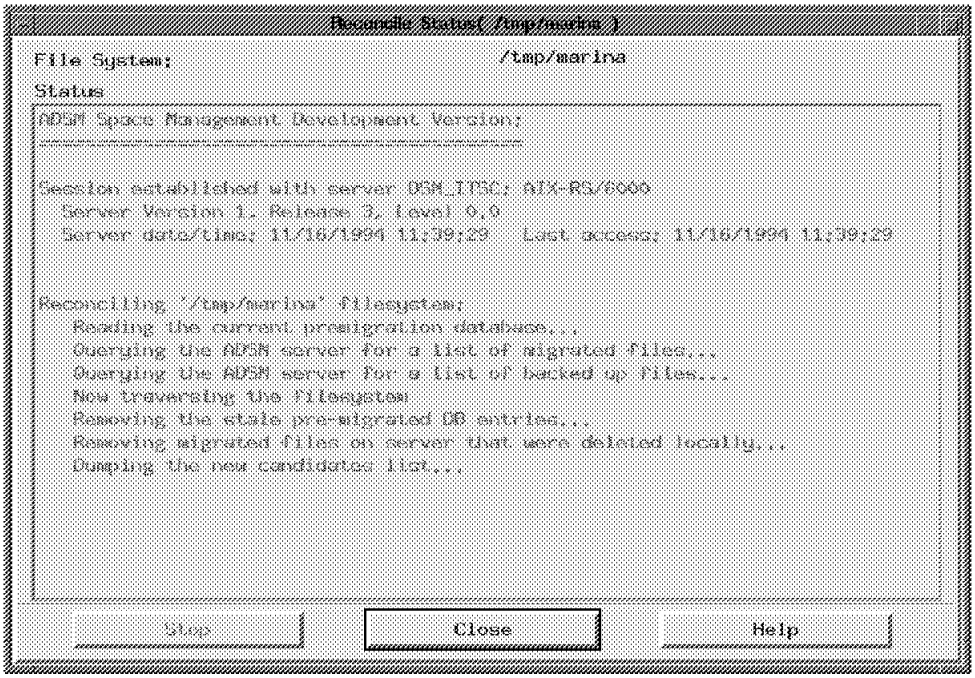

Figure 94. ADSM Space Management GUI Reconciliation Process

The tasks performed by this process are:

- Update the current premigrated file list by:
  - Removing entries for premigrated files that have been accessed

    Once a premigrated file is accessed, the copy at the server is no longer synchronized with the local copy and the server copy must therefore be removed
  - Verifying that all the files in the list reside in the server store
- Check for deleted files and eventually remove the corresponding entry from the server storage
- Build the new migration candidate list by:
  - Getting the current list of backed-up files
  - Getting the current list of files (migrated and premigrated) residing on the server
  - Getting the include/exclude option file information
  - Traversing the local file system to identify the candidate files
  - Reordering these files by a priority score based on the age and size factors specified by the user
- Check for orphaned stub files

  This should not happen normally, but may occur if the server that a file is being migrated to is switched.

The reconciliation process can be started manually using the `dsmreconcile` command or the GUI. By default, reconciliation will run automatically once every 24 hours. This time interval can be modified using an option in the dsm.sys file. The `dsmreconcile` command can also be scheduled using the central scheduleder.

7.1.1.4 Recall

When a migrated file is accessed by a user application, the ADSM recall daemon is started automatically and normally recalls the file from ADSM storage unless the application request can be satisfied by the information in the stub file. Using the `dsmmode` command, the execution modes of the daemon can be changed, tailoring the recall process to site specific needs. The recall process can also be altered for individual files using the `dmattr` command. The execution modes that can be changed are:

- Recall
- Data access control
- Timestamp
- Out of space protection

The recall process can be manually started from the GUI or by issuing the command `dsmrecall`.

Recall Mode: This determines the daemon behavior for access to migrated files.

normal	The recall daemon reads the file from the server storage and copies the file back to the local file system, replacing the stub file. If the file is not modified, it remains in premigrated state. This is the default. If the file is modified, it is put into resident state
migrate-on-close	This is similar to normal mode, but if the file is not modified it is returned to a migrated state
read-without-recall	This is intended to be used for single access sequential reads. When a migrated file is accessed, it is read sequentially from ADSM storage without actually storing the information back in the local client file system.

Data Access Control Mode: This controls how migrated files are accessed or recalled.

normal	This is the default; migrated files can be accessed and recalled
zero length	Such a migrated file is not recalled when accessed, nor is the local stub accessible. This mode can be useful when commands, such as `grep`, need to be run on resident-only files

Timestamp Control Mode: This controls the time set for migrated files when they are accessed.

normal	When the file is accessed, the time is changed to the current time
preserve	The access time and inode change time are not altered. This mode can be useful if file access times should remain unaffected by backup and archive operations

Out of Space Protection Mode: This determines whether ADSM should manage out of space conditions or not.

normal	ADSM attempts to recover from out of space conditions, not returning the error message to the user process
error	ADSM does not intercept out of space condition errors

7.1.2 Space Management Setup

In order to put client file systems under the control of ADSM space management, some tasks must be performed both by the ADSM system administrator and the root user on the client node.

7.1.2.1 ADSM Administrator Tasks

The system administrator needs to set up the server environment for space management:

1. Define additional storage pools for migration.

   At startup, ADSM will automatically set up the **SPACEMGPOOL**. This definition can be changed or a new one defined as shown in 4.1.1.1, “Disk Storage Pools” on page 70

2. Define additional storage volumes for migrated files as shown in 4.1.1.1, “Disk Storage Pools” on page 70

3. Update management class or create a new one with the following attributes:

   - Allowed types:
     - Automatic migration and selective migration
     - Selective migration only
     - No space management
   - Eligibility - the number of days since a file was last accessed before it becomes eligible for automatic migration
   - Backup - whether a backup version of the file must exist before it becomes eligible for automatic migration
   - Destination - the storage pool defined for migration

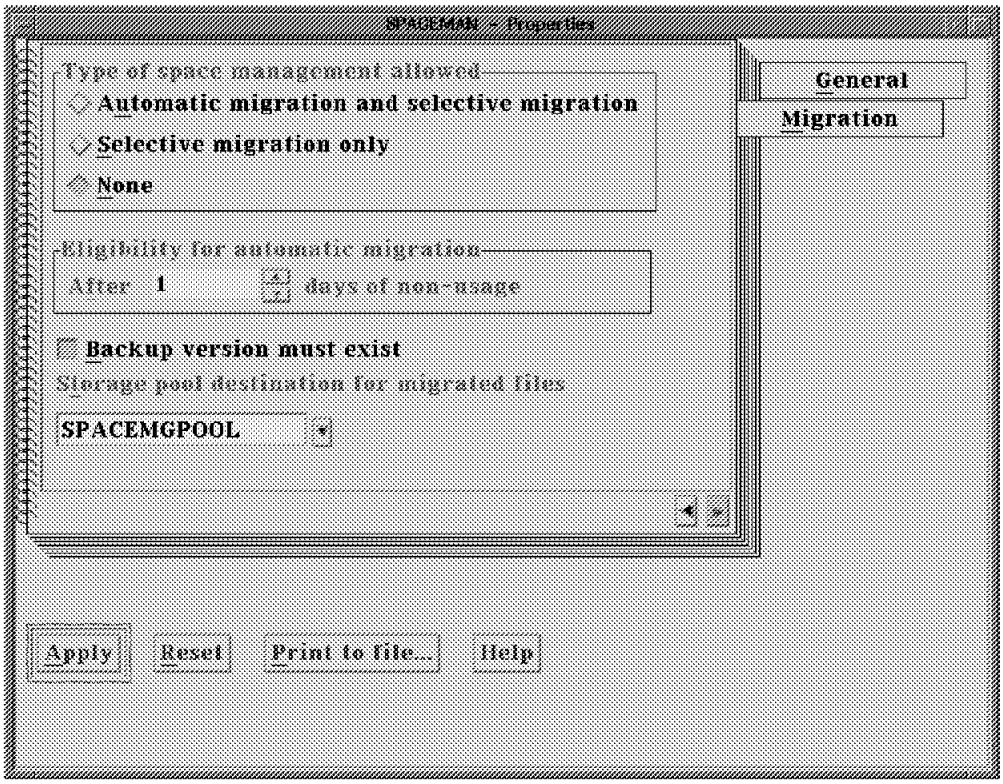

Figure 95. ADSM Space Management GUI Policy Setup

7.1.2.2 Root User Tasks

The root user on the client workstation must perform the following tasks:

- Convert local file systems for space management
- Activate space management on the local file system
- Set the options in the local dsm.sys and dsm.opt files
- Modify the include/exclude options file

Converting Local File Systems: ADSM currently only supports the JFS under AIX for space management. There are statements of direction for HFS on HP-UX, BDS on Solaris, and OS/2. When a file system is converted, ADSM mounts a file system migrator (FSM) over the native one. It's recommended not to convert the /usr and /var file systems.

To add space management to an existing file system, use the GUI or the following command:

```
dsmmigfs add <options> /filesystem
```

ADSM will:

- Create a directory named .SpaceMan in the file system which will be used to store information lists for space management tasks

- Add an entry in the file /usr/lpp/adsm/sm/config/dsmmigfstab containing all the default parameters. These parameters can be overridden by specifying options to the `dsmmigfs` command, manually editing the file or using the space management GUI

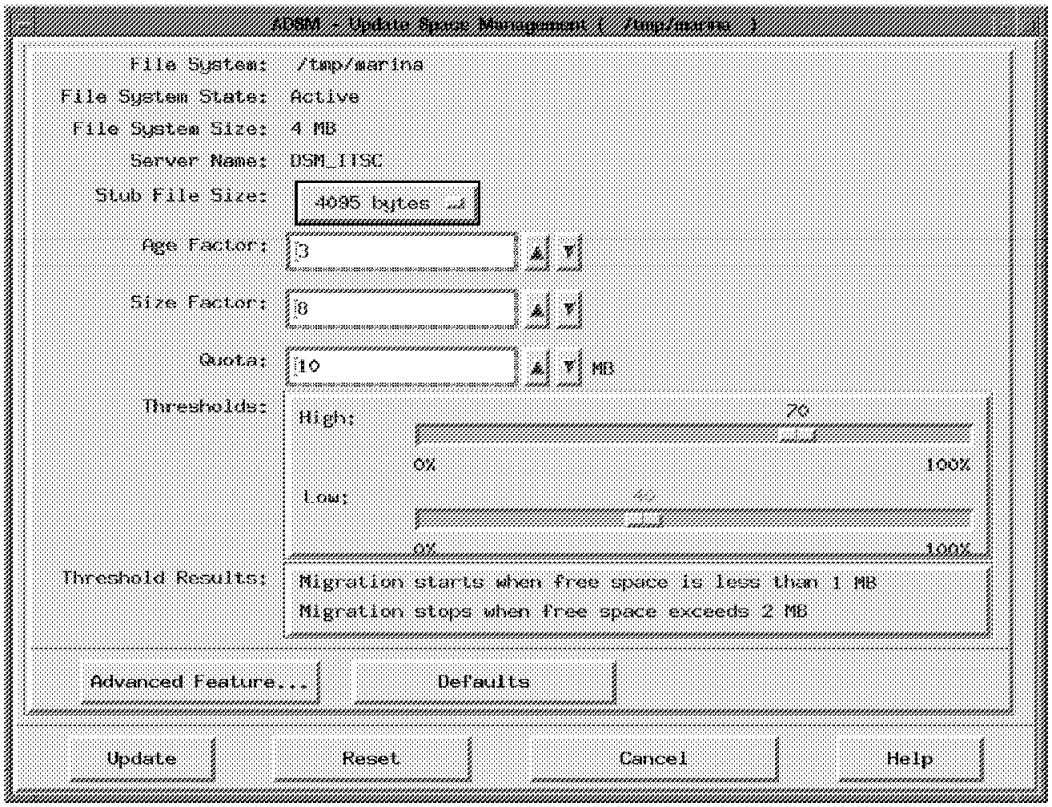

Figure 96. ADSM Space Management GUI Migration Parameters

```
# migfs            HWM(%) LWM(%) PWM(%) AgeF  sizeF  quota  stubsize
# ------------------------------------------------------------------------
# /mig/migfs1       90      80     -     -     -       -       -
/tmp/test          60      40     10     1     1       0       4095
/home              90      80     10     1     352     0       4095
/tmp/marina        70      40     30     3     8       10      4095
```

Figure 97. The migfstab File

The options that can be specified on the `dsmmigfs` command are as follows:

migfs The mount point of the file system

HWM High Water Mark. The percentage of space usage at which ADSM starts automatic file migration. The default is 90%

LWM Low Water Mark. The percentage of space usage at which ADSM stops migrating files. The default is 80%

PWM Premigration Water Mark. The percentage of space that can contain premigrated files. The default is the difference between HWM and LWM

AgeF The age factor: the age weight to determine file migration priority. The default is 1

sizeF The size factor: the size weight to determine file migration priority. The default is 1

quota The amount of server space reserved to store migrated files. The default is the file system size

stub size The size of the stub containing migrated file information. ADSM uses 511 bytes to identify the migrated file; all the remaining bytes contain the leader data of the migrated file. The default stub size is the original block size - 4095 bytes for AIX. Specifying a larger size, allows ADSM to store more information from the migrated file. In this way, the migrated file may not need to be recalled from ADSM storage every time it is required. No file is migrated if its size is less than the stub size

Activating Space Management: After a file system is converted, space management needs to be activated before ADSM can start the migration process. This task can be performed using the GUI or with the command:

```
dsmmigfs reactivate <options> /filesystem
```

ADSM will start performing all automatic space management tasks on the file system, and users will be able to selectively migrate or recall their files.

Setting Options in the Client System Options File: ADSM Version 2 will introduce new system options for the dsm.sys file to allow root users to set space management parameters for the client workstation user files.

CHEckthresholds	Minutes the space monitor daemon waits before checking space usage on local file system
DEFAULTServer	The server to which ADSM backs up and archives files. If the MIGRATEServer is not specified then space management will also use this server
ERRORProg	Specifies a program to which the space management function will send a message if an error occurs during space management processing
KERNELMessages	Determines whether HSM-related messages issued by the kernel are displayed. The default is yes
MAXRECAlldaemons	Maximum number of recall daemons to use
MAXRECOncileproc	Maximum number of reconciliation processes that can run at the same time for a client
MAXThresholdproc	Maximum number of threshold migration processes that can run at the same time for a client
MIGRATEFILEEXpiration	How many days to keep a migrated file after the reconciliation process has determined that it should be deleted. The default is 7 days
MIgrateserver	Server used to store migrated files
MINRecalldaemons	Minimum number of recall daemons to use
RECOncileinterval	Number of hours the space monitor daemon waits before initiating the reconciliation process

Setting Options in the Client User Options File: There is one new option that can be specified in the client user option file, dsm.opt:

OPTIONFormat Format (STANDARD or SHORT) in which the user can enter the space management commands options, when using the CLI

If this option is not specified, the default is STANDARD. This means that users on the workstation cannot use the short form.

```
dsmmigfs update -hthreshold=80 /filesystem
dsmmigfs update /filesystem -hthreshold=80

dsmmigfs update -h80 /filesystem
```

The standard format allows options and file specifications to be entered in any order; the short format is strictly positional, but the parameters can be shortened as in the second example above.

Modifying the Include/Exclude List: Using the standard include/exclude list for both backup and migration may not be the best solution. A file may need to be backed up but not migrated or migrated but not backed up. Different management classes may need to be assigned to files for backup and space management operations.

ADSM Version 2 will provide a solution, introducing new options that can be specified in the include/exclude file:

include Include a file or group of files in both backup and space management

exclude Exclude a file or group of files from both backup and space management

exclude.backup Exclude from backup only

exclude.spacemgmt Exclude from space management

The include options allow the management class that is to be assigned to the specified files to be defined. It can be used to associate different management classes to backed-up and space managed files, as in the following example:

```
include /home/marina/* standard
exclude.spacemgmt /home/marina/development/*
```

7.1.3 Using Space Management

After the setup of the space management services on the workstation's file systems, migration, recall and information display for the files can be performed. The root user can also start threshold migration and reconciliation processes.

The space management GUI display is shown in the following figure:

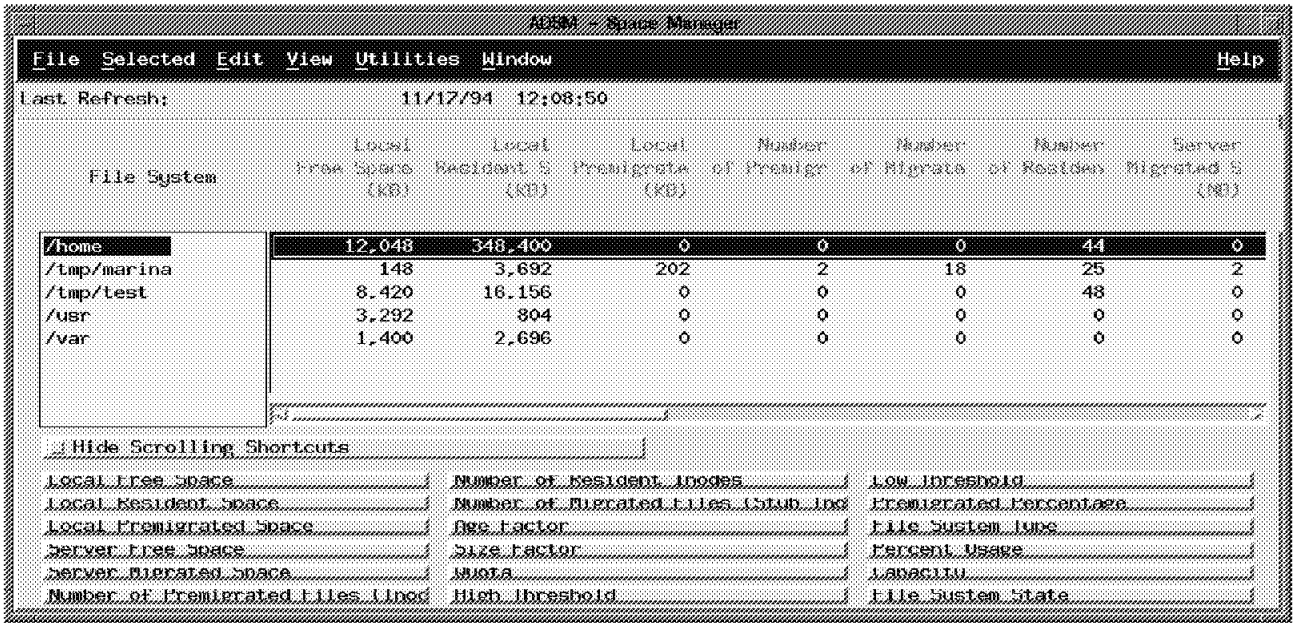

Figure 98. ADSM Space Management GUI

From here, the file system that is to be queried, migrated or recalled can be selected. The information about the file system can also be displayed as a pie chart. Changes in space usage can be observed on both the local file system and the server storage by refreshing the window shown in Figure 99 on page 183.

The CLI allows tasks to be performed by entering space management commands from the AIX prompt. The available commands are displayed in the following list:

dsmattr	Sets or displays the recall mode for a file
dsmautomig	Start threshold migration (only root user)
dsmdf	Display the number of migrated and premigrated files in the selected file system
dsmdu	Display the actual space usage in a directory and all its subdirectories
dsmls	Display migration status information
dsmmigfs	Update a file system (add, activate, deactivate, remove, update)
dsmmighelp	Display the online help for commands
dsmmigquery	Display migration candidates lists
dsmmigrate	Migrate selected files to the server storage
dsmmode	Set the execution modes
dsmmonitord	Starts the ADSM space monitor daemon
dsmq	Display information for files currently queued for recall
dsmrecall	Recall selected files
dsmrecalld	Starts the ADSM recall daemon
dsmreconcile	Build new migration candidates list (only root user)

dsmrm Remove a recall process from the queue

dsmsetpw Changes the ADSM password for the client node

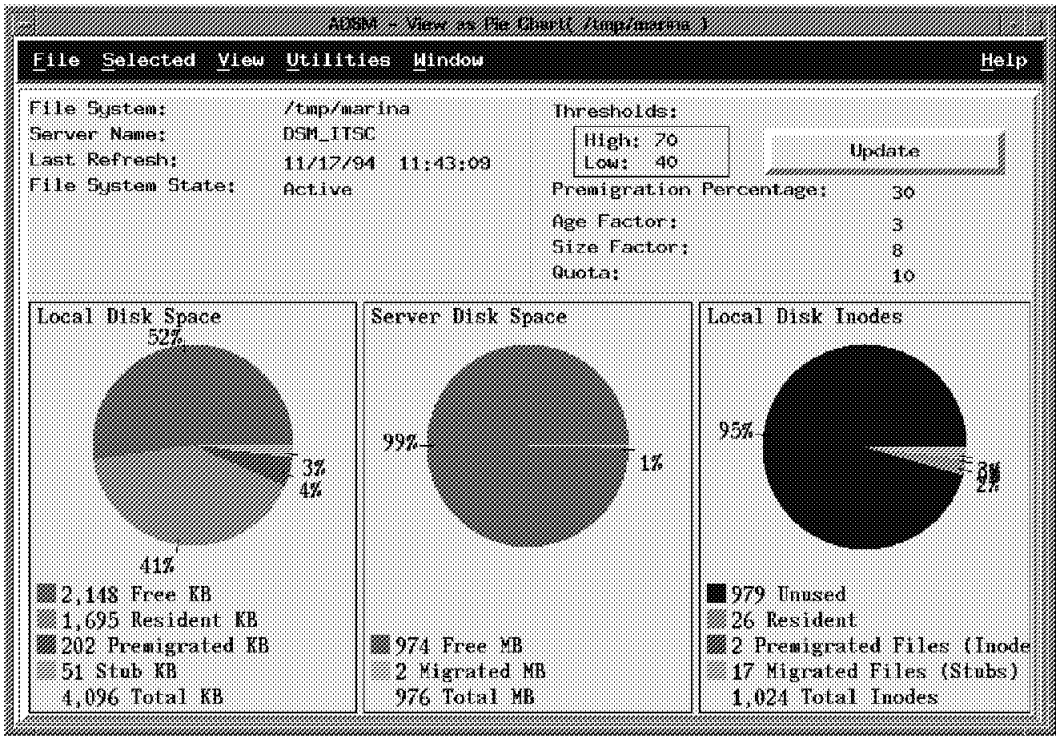

Figure 99. ADSM Space Management GUI Pie Chart View

The following figure shows the output of the command `dsmmigquery`. The default, as in this case, is to show the list of candidate files for migration. The weight is calculated from the parameters specified by the root user and shown in Figure 99. The weight is determined by:

- Multiplying the number of days since last access by the age factor
- Multiplying the size of the file in KB by the size factor
- Adding the two products

```
# dsmmigquery /tmp/marina

ADSM Space Management
=====================
Session established with server DSM_ITSC: AIX-RS/6000
  Server Version 1, Release 3, Level 0.0
  Server date/time: 11/17/1994 16:07:58   Last access: 11/17/1994 16:07:55

Weight            Size              Age  File Name

3360              430158            1    smit2.xwd
3008              385758            1    smit1.xwd
2124              269816            1    session.xwd
1088              140120            1    new.xwd
#
```

Figure 100. dsmmigquery Command Output

The following figure shows an example of the use of the `dsmdf` command. The output shows the state of space managed file systems in terms of whether space management is active, the amount of data migrated, the amount of data premigrated and the numbers of migrated and premigrated files.

```
# dsmdf

ADSM Space Management
=====================

Filesystem       MigFs   mgrtd   pmgrtd   mgrtd   pmgrtd
                 State    KB      KB      files   files

/tmp/test          a     8112    676      8       1
/home              a     0       0        0       0
/tmp/marina        a     4920    404      18      2
#
```

Figure 101. dsmdf Command Output

In the following figure, the output of the `dsmls` command is shown. It lists the sizes of files, the effective occupancy in the local storage and the file status. The file status can be:

m migrated
p premigrated
r resident

In this example, the file `fsfupd.xwd` is resident, while `smit1.xwd` is resident and eligible for migration, as can be seen from the display in Figure 100 on page 183.

```
# dsmls /tmp/marina

ADSM Space Management
=====================

size     ResSize(KB) MigStatus FileName

/tmp/marina:
512      4           r         .SpaceMan
140120   4           m         archive.xwd
393442   4           m         back1.xwd
4096     4           r         fsfupd.xwd
139264   4           m         help1.xwd
140120   140         p         new.xwd
229376   4           m         reconcile.xwd
397759   4           m         rename.xwd
186748   4           m         restore.xwd
385717   4           m         retrieve.xwd
436090   4           m         sched.xwd
269816   264         p         session.xwd
385758   380         r         smit1.xwd
430158   424         r         smit2.xwd
#
```

Figure 102. dsmls Command Output

7.2 File Storage Facility/6000

FSF provides transparent disk space management on client workstations by creating a personal data cache on the client local disk storage and automatically moving files to an NFS or ADSM server.

Users don't know the physical location of their files: if a user process requests to access a file residing only on the server store, FSF automatically copies the file back to the client cache.

7.2.1 FSF Services

In this subsection, the FSF services will be described in order to allow comparison with other space management products.

7.2.1.1 Caching

When an FSF file system is added, a cache is created on the local client disk connected to a server cache residing on an NFS or ADSM server machine. The configured FSF file system will contain both local files residing in the client cache and remote files residing on the server store.

Anytime a file residing on the server is accessed, FSF will automatically copy the file to the local cache. When the file is no longer being used, FSF will write the file back to the server store and eventually remove the local copy, freeing local space for other I/O operations.

The root user, as FSF administrator, can specify how long unused files can remain in the cache before being automatically written back to the server store.

Frequently used files can be pinned in order to prevent the deletion of the local copy. In this way, performance can be enhanced since files do not have to be recalled from server storage.

7.2.1.2 Writeback

The FSF daemon periodically polls the client cache in order to check if there are updated files to be written back to the server store. The frequency of polling can be defined by the administrator using the SMIT Fine-Tune FSF operation panel.

When a file is updated by a user application, the copy in the local cache no longer matches the copy in the server storage. They are said to be out of sync.

A file in the local cache becomes eligible to be written back to the server when the time spent in the out of sync condition becomes greater than the value set by the administrator or when more than a specified number of files all become out of sync.

In both cases, the first time that FSF polls the local cache, it starts the automatic writeback process.

7.2.1.3 Pruning

To avoid out of space conditions in the local cache, FSF periodically deletes (prunes) files that have not been changed for some time.

A file in the local cache becomes eligible to be pruned when its age becomes greater than the value specified by the administrator and if there is a synchronized copy in the server storage.

The automatic pruning process starts when the occupancy percentage of the local cache become greater than the *high water mark* parameter and stops when it drops below the *low water mark* parameter.

The eligible files are pruned according to a priority that the administrator can set. This is either age or size for all file systems. Pinned files are never pruned.

7.2.2 Using FSF

Users don't need to know the physical location of their files because the recall process is transparent and automatically performed by FSF. They can display the status and the location of all their files by issuing the command `cls`, which is an extension of the `ls` UNIX command.

7.2.2.1 File Attributes

The FSF attributes of files can be completely described by two identifiers, one related to the current file status and the other one to the location of the file:

- Status

  M Modified
  A Attributes modified
  C Clean (synchronized)
  N New empty file system

- Location

  L Local
  P Local pinned
  R Remote

```
$ cls -al
total 9216
drwxr-xr-x C L (0)   2 marina2 staff     4092 Nov 14 11:45 .
drwxr-xr-x           6 bin     bin        512 Nov 09 17:05 ..
-rwxr----- C P (0)   1 marina2 staff      424 Nov 10 18:15 .profile
-rw------- M L (0)   1 marina2 staff     1974 Nov 18 11:34 .sh_history
-rw-r--r-- C R (0)   1 marina2 staff   440364 Nov 18 10:32 archive.xwd
-rw-r--r-- C L (0)   1 marina2 staff   397759 Nov 18 10:45 rename.xwd
-rw-r----- A R (0)   1 marina2 staff   482203 Nov 18 11:07 sched.xwd
drwxr-xr-x N L (0)   2 marina2 staff     4092 Nov 18 11:42 testdir
$
```

Figure 103. cls Command Output

From the previous example, it can be seen that:

- The file .profile is local pinned and clean: it is synchronized with the server store copy
- The file .sh_history is local and modified; it is not synchronized with a server copy. This is the normal status when a remote file has been accessed read/write
- The file archive.xwd is remote and clean; the local copy has been cancelled (pruned)
- The file rename.xwd is local and clean; this is the normal status when a remote file has been accessed in read-only mode

- The file sched.xwd is remote, but its attributes have been locally changed
- The directory testdir has just been created; it is local and new

7.2.2.2 User Commands

The root user can perform all tasks using the SMIT interface, as has already been shown in 4.2, “File Storage Facility/6000” on page 90.

FSF provides some line commands that all client workstation users can issue from the AIX command prompt to monitor and change the status of their own files. The following is a list of all the FSF commands available:

cachefiles	Recall a file, a group of files or a file system from the server to the local cache. With the `-p` option, also display the files
cdf	Display the available space in the cache
cls	Display information about fsf files
getcachetimeout	Display the current timeout value for files being cached from the server store
getmdata	Display the internal structure of the specified file
pinfiles	Prevent a local file from being pruned (only root user). If the file status is remote, it should first be recalled to the local cache
prunefiles	Remove a file or a group of files from the local cache, if the current version of the file is already stored at the server
setcachetimeout	Change the current timeout for caching files from server store (only root user)
unpinfiles	Allow a local file previously pinned to be pruned as part of the normal automatic pruning operation

7.2.2.3 Examples

This subsection illustrates the usage of some of the commands discussed in the previous section with a few examples.

```
$ cdf
Filesystem    Total KB    free %used    iused %iused Mounted on
/dev/lv02         4096    3924    4%       49     0% /u/marina2 (mfs)
/dev/hd1         12288      16   99%       65     2% /home (cache)
$ cdf
Filesystem    Total KB    free %used    iused %iused Mounted on
/dev/lv02         4096    3924    4%       49     0% /u/marina2 (mfs)
/dev/hd1         12288    1444   88%       61     1% /home (cache)
```

Figure 104. cdf Command Output.

From the two displays, it can be seen that FSF is performing automatic pruning in order to free space in the local cache.

```
$ cachefiles -p *.xwd
$
------------------
cachefile summary:
------------------
cache 'archive.xwd': Successful
cache 'rename.xwd': Successful
cache 'sched.xwd': Successful
```

Figure 105. cachefiles Command Output

From this example, it can be seen that the `cachefiles` command has caused all of the Xwindow dump files to be copied back to the client cache. The `-p` option triggered the display of the files actually copied.

```
$ getmdata sched.xwd

0: File 'sched.xwd': mdata_t struct:

    Store     filehandle struct:
        fh_fsid:
            fsid_dev    0x230009
            fsid_type   0x3
        fh_id:
            fid_len     0xa0000
            fid_ino     0x8122ecc
            fid_gen     0xdfb60000
    dm_flags        0x100
    dm_family       0x0
    dm_first_dirty  0x2ecceee4
sched.xwd: not pinned, remote: data clean, mode changed
$
```

Figure 106. getmdata Command Output

This example illustrates the kind of information stored by FSF for a file that it is managing. The `getmdata` command displays the information contained in the stub file for the specified file.

7.3 Product Comparison Summary

The products discussed in this redbook are compared with regard to space management capability on the basis of the following criteria:

- Function
- Ease of use

7.3.1 Function

This subsection is concerned with the space management functions provided by the products discussed in this redbook.

7.3.1.1 ADSM

ADSM provides the following functions:

- Migration

  The decision to begin file migration to the server store is made on the basis of thresholds (cache utilization), demand (out of space) or selectively based on user action. During the reconciliation process, files are selected based on a weighting factor calculated from their age and size, and placed on the migration candidates list. When migration then becomes necessary, files on this list are migrated to the server and replaced with a stub file whose size is user-definable, so that it can therefore still contain a variable amount of the actual file data. This enables some processes to operate on the file without the need to recall it.

- Premigration

  A certain percentage of the files in cache can be marked for premigration. This means that ADSM will migrate them to the server store without removing them from the local cache. Should space become critical these premigrated files need only be erased and replaced with a stub file to free up space.

- Reconciliation

  This process runs at regular intervals to monitor and resynchronize the status of local and remote files and to build a list of files eligible for migration.

- Recall

  When a migrated file is accessed by a user process, the ADSM recall daemon locates and copies the required file from the server store. Files can be recalled in a number of modes. They can be returned in premigrated state, merely read from the server and not copied back, copied back normally, or not recalled at all.

- Scheduled operations

  The administration of the space management component as well as the actual migration and recall processes can be automated with ADSM Version 2 using the central scheduling facility.

- Manual operations

  Users can control space management functions from either the command line or via a GUI. Operations include starting threshold migration, migrating specific files, recalling specific files, and setting the operational characteristics of the recall and migrate daemons.

- Display operations

  Cache utilization can be displayed graphically from the GUI. Various commands also exist to display the status of reconciliation, recall and migration.

7.3.1.2 FSF

FSF provides the following space management functions:

- Writeback

  FSF periodically (based on tunable parameters) checks the client cache to see if any files are eligible for migration. Criteria for migration include file age, cache utilization and file size. Any files so found are copied to server store and a stub file left in their place.

- Caching

  When a migrated file is accessed, FSF recalls the file from server store.

- Manual operations

  Users can manually select files to be migrated using a command line call. Files that will be regularly accessed can also be pinned, which renders them permanently ineligible for migration. The converse, unpinning, is also possible.

  A command also exists to change the timeout value for file caching.

- Display operations

  There are a number of status display commands that will show the current cache utilization, the status of files in the cache and the internal structures of stub files.

7.3.1.3 UniTree

UniTree does not provide space management functions.

7.3.1.4 Legato NetWorker

NetWorker does not provide space management function.

7.3.2 Ease of Use

This section is concerned with the relative ease of use of the space management services.

7.3.2.1 ADSM

ADSM provides a complete GUI as well as command line functions for setup and control of space management. This, coupled with the graphical display options, makes ADSM space management very easy to use.

Accelerator buttons also exist to jump quickly to required status displays.

7.3.2.2 FSF

FSF is controlled and used through a combination of SMIT menus and command line functions. No GUI is provided though once setup is complete, there is little requirement for user monitoring of its operation since it should be transparent.

7.3.2.3 UniTree

UniTree does not provide space management function.

7.3.2.4 Legato NetWorker

NetWorker does not provide space management function.

Chapter 8. Remote File Systems

This chapter looks at remote file systems and their relationship to the products discussed in this redbook. For those products that can interact with remote file systems, the necessary steps to configure the products and utilize the file systems will be described.

8.1 Remote File System Configuration

This section looks at the setup necessary to access the remote file systems for access by the products. The file systems that are discussed include:

- AFS
- NFS

8.1.1 Andrew File System

The steps necessary to configure a system for AFS are beyond the scope of this redbook since it will be very much site dependent.

To see whether AFS is available on a system, check for the following:

- AFS file systems

```
[root@solo] /tmp > mount
  node       mounted        mounted over    vfs       date        options

-------- ---------------  ---------------  ------ ------------ ---------------
         /dev/hd4         /                jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd9var      /var             jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd2         /usr             jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd3         /tmp             jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd1         /home            jfs    Nov 17 12:45 rw,log=/dev/hd8
         /dev/lv01        /usr/vice/cache  jfs    Nov 17 12:45 rw,log=/dev/hd8
         AFS              /afs             afs    Nov 17 12:46 rw
```

  There should be a `/usr/vice/cache` and a file system of type `afs`.

- AFS daemons

```
[root@solo] /tmp > ps -ef|grep afs
    root  3190     1   0   Nov 17      -  0:04 /usr/vice/etc/afsd
    root  6263     1   0   Nov 17      -  7:02 /usr/vice/etc/afsd
    root  6520     1   0   Nov 17      -  0:08 /usr/vice/etc/afsd
    root  6777     1   0   Nov 17      -  0:07 /usr/vice/etc/afsd
    root  7034     1   0   Nov 17      -  0:04 /usr/vice/etc/afsd
    root  7291     1   0   Nov 17      -  0:04 /usr/vice/etc/afsd
    root  7548     1   0   Nov 17      -  0:03 /usr/vice/etc/afsd
    root  7805     1   0   Nov 17      -  0:03 /usr/vice/etc/afsd
    root  8062     1   0   Nov 17      -  0:03 /usr/vice/etc/afsd
    root 28482 31321   5 12:27:03  pts/4  0:00 grep afs
```

  The AFS daemons should be running.

8.1.2 Network File System

Three implementations of NFS are covered:

- AIX NFS
- VM NFS
- MVS NFS

8.1.2.1 AIX NFS

Using AIX NFS requires that certain NFS subsystems be active on the client and server machines. These can be started, using SMIT, as follows

1. Enter `smitty nfs` from a command line
2. Select the option: **Network File System (NFS)**
3. Select the option: **Configure NFS on This System**
4. Select the option: **Start NFS**

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: yes

Before command completion, additional instructions may appear below.

0513-059 The portmap Subsystem has been started. Subsystem PID is 11731.
starting nfs services:
0513-059 The biod Subsystem has been started. Subsystem PID is 10460.
0513-059 The nfsd Subsystem has been started. Subsystem PID is 10981.
0513-059 The rpc.mountd Subsystem has been started. Subsystem PID is 9966
0513-059 The rpc.statd Subsystem has been started. Subsystem PID is 5616.
0513-059 The rpc.lockd Subsystem has been started. Subsystem PID is 20978
0513-095 The request for subsystem refresh was completed successfully.









F1=Help             F2=Refresh          F3=Cancel           F6=Command
F8=Image            F9=Shell            F10=Exit
```

In order to check whether NFS is active, the following command can be used:

```
[root@solo] / > lssrc -g nfs
Subsystem         Group            PID     Status
 biod             nfs              10460   active
 nfsd             nfs              10981   active
 rpc.mountd       nfs              9966    active
 rpc.statd        nfs              5616    active
 rpc.lockd        nfs              20978   active
[root@solo] / >
```

The directory that is to be used remotely must then be exported at the NFS server. This can also be done using SMIT:

1. Enter `smitty nfs`

2. Select **Network File System (NFS)**
3. Select **Add a Directory to Exports List**

```
                        Add a Directory to Exports List

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                      [Entry Fields]
* PATHNAME of directory to export                    []                        /
  MODE to export directory                            read-write               +
  HOSTNAME list. If exported read-mostly              []
  Anonymous UID                                       [-2]
  HOSTS allowed root access                           []
  HOSTS & NETGROUPS allowed client access             []
  Use SECURE option?                                  no                       +
* EXPORT directory now, system restart or both        both                     +
  PATHNAME of Exports file if using HA-NFS            []



F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Where

PATHNAME Specifies the path of the directory that will be accessed remotely

MODE Specifies how the directory will be able to be accessed remotely. Most functions require read/write

The rest of the options can be allowed to default unless special requirements exist. More information can be found in the online InfoExplorer hypertext information library.

At the client machine, the exported directory must be mounted into the local file system for access:

```
[root@solo] / > mount ori:/nfssrvdisk /mnt
```

Where:

ori The hostname of the server machine

/nfssrvdisk The directory that was exported at the server

/mnt The mount point in the local file system at which the remote file system is to be accessed from

There are more options to the `mount` command that can be used, if required, and more information on these options can be found in the online InfoExplorer hypertext information library.

8.1.2.2 VM NFS

In order to utilize the VM NFS server, some configuration is required at the VM host to which access is required. The three steps which need to be taken are:

1. Ensure TCP/IP is running on the VM host where the VM NFS server is running.

   Access will be required to the TCP/IP disk, TCPMAINT. This can normally be linked to (if not already done) with the command:

   ```
   Ready; T=0.02/0.02 17:30:03
   LINK TCPMAINT 592 592
   ```

2. Give the VMNFS ID permission to perform NFS transactions

   The NFS server needs to be given permission to access the files that are to be shared. This is done by executing the `NFSPERM ADD` command:

   ```
   Ready; T=0.72/0.82 17:12:50
   NFSPERM ADD
   PERMIT NICKH CLASS(VMBATCH) ACCESS(CONTROL) ID(VMNFS)
   END
   ```

   In order to revoke permission when no longer required, the `NFSPERM DELETE` command is used.

3. Send a password to the server from the AIX platform

   In order to send the password to the VM NFS server to verify that the client is authorized to mount, a utility called `mountpw` is required. The mount password can be specified in the `mount` command, but this can be a security exposure, hence the use of the utility. The utility should be available for download from the VM system where the VM NFS server is located. It is a file called `MOUNTPW 6000_BIN`, found on the TCPMAINT disk. Using FTP, download this file to the client machine and make it executable:

   ```
   root@solo> chmod +x mountpw
   ```

   The `mountpw` command can now be used:

   ```
   root@solo> mountpw ausvmr:nickh.191,password=mypasswd,userid=nickh
   ```

   Where:

   ausvmr Name of the VM host machine

   nickh.191 The userid of the disk owner, a dot, and then the virtual address of the disk to be mounted. This is the disk for which authorization was given to RACF in the previous step

   mypasswd VM password for the userid owning the disk

   nickh Userid owning the disk

4. Perform the mount

Now that authority has been obtained, the mount can be performed. The command is executed in the same way as for any AIX NFS mount though additional VM NFS options can be added to the host:resource as follows:

```
root@solo> mount ausvmr:nickh.191,mode,record=nl,names=mix /mountpoint
```

Where:

ausvmr The VM host name

nick.191 The userid of the disk owner, a dot, and then the virtual address of the disk to be mounted

mode Optional. This can be `rw` for read/write, or `ro` for read only. If no value is entered, the default is read only

record=nl Optional. The parameter can be `nl`, signifying text mode, meaning that data will be translated ASCII to EBCDIC and vice versa, or it can be `binary` signifying no translation for binary data such as executables

names=mix Optional. This parameter can be `mix`, signifying that any files created will be stored exactly as is on the VM minidisk. This means that if lowercase or mixed-case file names are used, they will not be accessible from VM. It can also be `fold`, signifying that the VM NFS server will assume all VM files to be uppercase and all workstation files to be lower case. This is the recommended setting for most situations. The only time the `mix` option is likely to be required is when mixed-case VM files, such as notelogs, need to be accessed

/mountpoint The point in the AIX file system where the VM minidisk is to be mounted.

When access to the VM minidisk is no longer required, the standard AIX `umount` command can be used:

```
root@solo> umount /mntpoint
```

If a minidisk is mounted read-only in an AIX file system, any changes made to files on that minidisk at the VM host will not be reflected at the AIX system until the VM NFS server has refreshed its link to the minidisk. This can be accomplished by issuing the following command at the VM host:

```
Ready; T=0.14/0.16 18:28:52
SMSG VMNFS M REFRESH nickh.191
```

Where:

nickh.191 is the userid of the disk owner, a dot, and then the virtual address of the disk that is mounted

If it should be necessary to get the VM NFS server to detach its link to the minidisk (this should happen when the `umount` command is used, but may not in the event of a crash), the following command can be used:

```
Ready; T=0.14/0.16 18:34:15
SMSG VMNFS M DETACH nickh.191
```

Where:

nickh.191 The userid of the disk owner, a dot, and then the virtual address of the disk that was mounted

8.1.2.3 MVS NFS

There are a number of different ways in which data is stored at an MVS host, and the mechanism used governs the type of access available via NFS. In addition, some setup is required at the MVS host and AIX system prior to mounting.

MVS NFS Setup: Setup of the MVS NFS server must be performed by an MVS system administrator and is described in the manual *DFSMS/MVS Network File System Customization and Operation*. The steps basically involve:

1. Installing the NFS server
2. Protecting server control data sets
3. Defining the server to RACF
4. Defining the level of user data access protection
5. Defining the level of client authentication

Once the setup is complete, NFS can be customized for use. This involves the following steps:

1. Allocating and modifying the attributes data set

   The attributes data set defines how data sets will be created and processed. Under MVS 4.2 this should be named prefix.NFSCNTL(NFSATTR). The prefix.NFSSAMP(GFSAPATT) dataset can be used as a sample. There are three sets:

   a. Data set creation attributes

      These attributes can be altered from the AIX host and include options such as:

      - Disk block size
      - Directory size for partitioned data sets (PDS)
      - Data set type to be created when `mkdir` is used
      - Data set organization (physical sequential, VSAM)
      - Logical record length

      Essentially, these govern how data created at the AIX host will be processed and stored by MVS.

   b. Processing attributes

These attributes can be overridden at the AIX system and include options such as:

- Binary or text data
- Execute bit on or off for created files
- Map lowercase files to uppercase
- Synchronize written data immediately or not

These attributes govern how data is interpreted as well as the actual access to information at the MVS host.

c. Site attributes

These attributes are fixed and cannot be altered at the AIX system. They include options such as:

- Buffer sizes
- Inactivity time limits
- Command timeout limits
- Readahead values

These attributes are essentially performance related.

The choice of data set type, and hence the setting of the related parameters, is based on a number of criteria. This will be discussed more fully later in the chapter in the sections describing the use of remotely mounted file systems with the products included in this redbook.

2. Allocating and modifying the exports data set

   The exports data set contains entries for those data sets that can be mounted by clients. This is analogous to the /etc/exports file in AIX. Under MVS 4.2, this should be called prefix.NFSCNTL(EXPORTS). The prefix.NFSSAMP(GFSAPEXP) dataset can be used as a sample.

3. Allocating the mount handle data sets

   This data set is used to record those currently mounted data sets. It is used to allow client mounts to remain active across server shutdown and restart.

4. Customizing the ASCII to EBCDIC translation tables

   The ASCII to EBCDIC translation table is used when the processing attribute `text` is enabled. A customized version can be created and specified in the `xlat` processing attribute if there are special translation requirements.

5. Allocating log data sets

   Primary and secondary log data sets must be allocated for use by the NFS server.

6. Defining which elements of NFS will be operational

   The file tcpip.ETC.RPC contains a list of the services that are required to be operational from the NFS server. This is analogous to the rc.nfs file in AIX. The services are:

   - NFS daemon
   - mount daemon
   - MVS mount daemon (for mvslogin and mvslogout)

- showattr daemon
- pcnfs daemon

7. Downloading the MVS NFS client commands to the AIX host

   There are a number of client commands that can be used from the AIX system. The file prefix.NFSTARB(GFSAWAIX) must be downloaded to the client AIX system, using FTP, as client.tarbin. The file then needs to be unpacked and the required executables made as follows:

```
root@solo> tar -xvf client.tarbin
tar: record size = 20 blocks
x ./makefile, 8234 bytes, 17 tape blocks
x ./gfsawaxd.c, 13474 bytes, 27 tape blocks
       .
       .
root@solo> make aix_rs
```

   This will compile the required executable for AIX. Once this has been done, it should now be possible to mount an MVS data set. These procedures are described in more detail in *DFSMS/MVS Network File System User's Guide*:

   a. `mvslogin`

      This command is used to authenticate the AIX client for the data sets to which access is required. The user profile submitted is used by MVS to determine which access permissions to grant. The command is executed as follows:

```
root@solo> mvslogin -p stlmvs1.stl.ibm.com tang
GFSA973A Enter MVS password:
GFSA955I tang logged in ok.
root@solo>
```

      Where:

-p	Indicates the password should be prompted for. The `-P` password option can also be used to supply the password on the command line
ststlmvs1.stl.ibm.com	The MVS host name
tang	The userid on the MVS host that owns the data sets that are to be accessed
Enter MVS password:	The password for the host userid must be supplied

      The `mvslogin` command is not required if security has not been implemented at the MVS host.

   b. `mount`

      The `mount` command is the standard AIX command though additional MVS NFS options can be added to the command:

```
root@solo> mount -o intr stlmvs1.stl.ibm.com:"tang,binary" /mnt
root@solo> mount
  node       mounted        mounted over    vfs       date        options
-------- ---------------  ---------------  ------ ------------ ---------------
         /dev/hd4         /                jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd9var      /var             jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd2         /usr             jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd3         /tmp             jfs    Nov 17 12:44 rw,log=/dev/hd8
         /dev/hd1         /home            jfs    Nov 17 12:45 rw,log=/dev/hd8
stlmvs1.stl.ibm.com tang,binary      /mnt             nfs    Dec 07 13:14 rw,intr
```

As can be seen from the `mount` command, the MVS data set has been successfully mounted. The parameters are as follows:

-o — The standard AIX `mount` options flag. The interruptible option has been selected here to allow mount attempts to be discontinued if the MVS host should become unavailable.

stlmvs1.stl.ibm.com — The name of the MVS host.

"tang,binary" — Where `tang` is the high-level qualifier that specifies which MVS data set to mount. A comma-separated list of NFS data set and/or processing attribute overrides can be appended. In this case the processing option binary has been specified

/mnt — The point in the AIX file system where the MVS data set is to be mounted

There are several ways in which to specify the high-level qualifier, depending on the data set type that is to be mounted. The possibilities are covered in the next section on data access mechanisms.

c. `showattr`

Once a data set has been mounted, the `showattr` command can be used to view the default attributes or the attributes set for a particular mount point. The command is used as follows:

```
root@solo> showattr stlmvs1.stl.ibm.com

DFSMS/MVS 1.2.0 Network File System Server Data Set Creation Attributes:

lrecl(8196)             recfm(vb)               blksize(0)
space(100,10)           blks                    dsorg(ps)
dir(27)                 unit()                  volume()
recordsize(512,4K)      keys(64,0)              nonspanned
shareoptions(1,3)       model()
mgmtclas()              dsntype(pds)            norlse
dataclas()              storclas()

DFSMS/MVS 1.2.0 Network File System Server Processing Attributes:

binary                  lf                      blankstrip
nofastfilesize          noretrieve              maplower
mapleaddot              executebitoff           setownerroot
attrtimeout(120)        readtimeout(90)         writetimeout(30)
sync                    xlat()

DFSMS/MVS 1.2.0 Network File System Server Site Attributes (not modifiable):

mintimeout(1)           nomaxtimeout            logout(1800)
nfstasks(8,8)           restimeout(48,0)        hfs(/hfs)
bufhigh(2M)             readaheadmax(16K)       cachewindow(16)
percentsteal(20)        maxrdforszleft(32)      logicalcache(1M)
security(saf)           smf(none)       nopcnfsd        leadswitch
```

This shows the attributes globally in effect for the MVS host specified. The attributes relating to a specific mount point can also be viewed by appending the mountpoint:

```
root@solo> showattr stlmvs1.stl.ibm.com /mnt
```

With similar results to the previous invocation.

d. `umount`

When the MVS data set is no longer required, it can be unmounted using the standard AIX `umount` command:

```
root@solo> umount /mntpoint
```

Where `/mntpoint` is the point in the AIX file system where the data set is mounted.

e. `mvslogout`

When access to MVS data sets is no longer required, the `mvslogout` command can be used to disconnect from the MVS host. The command is used as follows:

```
root@solo> mvslogout stlmvs1.stl.ibm.com
GFSA958I uid 0 logged out ok.
```

MVS Data Access Mechanisms: Having looked at the process required to access MVS data sets, this section now briefly examines the data sets themselves in order to highlight some of the differences between them from an AIX perspective.

File naming must follow the MVS conventions for all data sets except OE HFS. That is to say, each file name is composed of a number of qualifiers, each of which can be from one to eight alphanumeric characters in length. Qualifiers are separated by periods, and the totally qualified name cannot exceed 44 characters in length. The specifics, in terms of numbers of qualifiers, are discussed for each data set type.

- Physical Sequential (PS)

  To create physical sequential files, the `dsorg(ps)` data set attribute must be specified. Using the mount syntax specified previously and editing a file called test will result in a file organized sequentially with a fully qualified name on MVS of TANG.TEST. From AIX, a file called test will exist in /mntpoint.

  All files created will have AIX permission set as follows:

```
root@solo> li -l
Frw-rw-rw-    1 root     system         12  Nov 17 03:56 test
```

  If the processing attribute `executebiton` is set first, the permissions will be as follows:

```
root@solo> li -l
Frwxrwxrwx    1 root     system         12  Nov 17 03:56 test
```

  With PS files, the attributes cannot otherwise be changed.

- Direct Access (DA)

  To create direct access files, the `dsorg(da)` data set attribute must be specified. Using the mount syntax specified previously and editing a file called test will result in a direct access file with a fully qualified MVS name of TANG.TEST. From AIX, a file called test will exist in /mntpoint.

  For DA files, permissions are the same as for PS files.

- Partitioned Data Set (PDS)

  To create partitioned data sets, the `dsntype(pds)` data set attribute must be specified. Using the mount syntax specified previously and the `mkdir tmp` command will result in the creation of a partitioned data set named tmp. From AIX, this will appear as a directory named tmp. It is now possible to `cd` to the directory and create PS or DA files within it. Creating a file named test will result in a fully qualified MVS name of TANG.TMP(TEST), where TEST is a member of the PDS TMP.

  Permissions are set as specified previously for DA and PS files.

  Note that it is not possible to create a PDS or PDSE within another PDS or PDSE. This is a single-level hierarchy only. All of the usual AIX commands, such as `rmdir`, operate on a PDS.

  The PDS does not support concurrent writing to members within it. Thus, if one member is being edited, an attempt to access another member simultaneously will result in a permission denied message.

- Partitioned Data Set Extended (PDSE)

To create partitioned data sets extended, the `dsntype(library)` data set attribute must be specified. Using the mount syntax specified previously and the `mkdir tmp` command will result in the creation of a partitioned data set extended named tmp. It is now possible to `cd` to the directory and create PS or DA files within it. Creating a file named test will result in a fully qualified MVS name of TANG.TMP(TEST), where TEST is a member of the PDSE TMP.

Permissions are set as specified previously for DA and PS files.

Note that it is not possible to create a PDS or PDSE within another PDS or PDSE. This is a single-level hierarchy only. All of the usual AIX commands, such as `rmdir`, operate on a PDSE.

The PDSE does support concurrent member access.

- VSAM

  Three types of VSAM file are supported:

  – Key sequenced (KSDS)

  – Entry sequenced (ESDS)

  – Relative record (RRDS)

  Creation of a VSAM file is possible only by copying the attributes of an existing VSAM file. For more information, see the Users guide: *DFSMS/MVS Network File System User's Guide*.

- SAM striped data sets

  Data striping is supported vis NFS-mounted data sets through the use of data class and storage class attributes at the MVS host. See the Users Guide for more details: *DFSMS/MVS Network File System User's Guide*.

- OE HFS

  The OpenEdition MVS support provides for a hierarchical file system. The OE HFS is organized into a directory tree structure very much like an AIX file system, with a root directory and subdirectories. Files in an OE HFS are byte oriented rather than record oriented though still stored on MVS in EBCDIC. The MVS NFS server will perform EBCDIC to ASCII translation, if required. The OE HFS file system is POSIX compliant and, as a result, supports the following:

  – Hierarchical directories

  – File names up to 255 characters in length

  – Path names up to 1023 characters in length

  – File names of mixed case and including special characters

  – UNIX-style file access permissions

  – User and group IDs at file level

  – Full NFS protocol including external links

  – Linkage of external MVS data sets into HFS

  When the HFS file system is set up on the MVS host, an HFS prefix is created that acts as a qualifier for the file system. This prefix can be seen in the site attributes and defaults to /hfs. A root directory is also created from which subdirectories can be set up as required. Mounting can, as would be expected, be performed for any exported subdirectory of the HFS (including the root directory).

Mounting an HFS file system is accomplished as follows:

```
root@solo> mount -o intr stlmvs1.stl.ibm.com:"/hfs/,binary" /mntpoint
```

The command is used identically to the invocations for the other data sets. The only difference being that the high-level qualifier is replaced by the HFS prefix, followed by the directory that is to be mounted, which in this case, is the root directory.

As for the other data sets, data set and processing attributes can be appended to the qualifier.

Once mounted, the file system can be used in exactly the same way as a normal AIX file system. Directories and subdirectories can be created hierarchically and files created within the directories.

8.2 ADSTAR Distributed Storage Manager/6000

Remote file systems interact with two distinct parts of ADSM:

- Space management
- Backup/archive services

8.2.1 Space Management

ADSM space management set up and usage has been covered in 7.1, “ADSTAR Distributed Storage Manager/6000” on page 173. This product does not currently support file systems that are not local and therefore will not allow space management of the remote file systems discussed in this chapter.

8.2.2 Backup/Archive

ADSM backup and archive services have been discussed in 5.2, “ADSTAR Distributed Storage Manager/6000” on page 138 and 6.1, “ADSTAR Distributed Storage Manager/6000” on page 165. This section looks at using ADSM to backup and archive files from the remote file systems discussed in the first part of this chapter.

8.2.2.1 AFS

ADSM can be used to backup or archive files stored in an AFS file system. The user GUI will list any AFS file systems in the main window. They can be selected for backup/restore or archive/retrieve operations.

AFS file systems are normally large; so it is not generally advisable to select the **Backup by directory tree** option from the Backup or Restore menus. Instead the option to **Backup by file specification** should be used. ADSM Version 2 allows subdirectories to be included from this option anyway; so complete directory trees can be backed up or archived from any point within the AFS file system.

8.2.2.2 AIX NFS

ADSM can also be used to backup or archive files from an AIX NFS-mounted file system. Once again, the file system will be identifiable from within the main window of the user GUI, having a type NFS.

8.2.2.3 VM NFS

In order to allow ADSM to backup or archive files from a VM NFS-mounted file system, a parameter needs to be added to the dsm.opt file. VM file systems are in fact minidisks and therefore do not have the UNIX-type directory structure; specifically, the **.** and **..** files are not present. ADSM checks for the existence of the **.** and **..** files in order to verify that the file system is a valid one. If they are not present, by default, any requested operations will fail.

Thus, the following parameter needs to be added:

```
[root@solo] /usr/lpp/adsm/bin > vi dsm.opt
************************************************************************
* ADSTAR Distributed Storage Manager                                   *
*                                                                      *
* Sample Client User Options file for AIX and SunOS (dsm.opt.smp)      *
************************************************************************

*  This file contains an option you can use to specify the ADSM
*  server to contact if more than one is defined in your client
*  system options file (dsm.sys).  Copy dsm.opt.smp to dsm.opt.
*  If you enter a server name for the option below, remove the
*  leading asterisk (*).

*  For information about additional options you can set in this file,
*  see the options.doc file in the directory where ADSM was installed.

************************************************************************

 SErvername       DSM_ITSC

 dotdircheck      no



"dsm.opt" 20 lines, 909 characters
```

The `dotdircheck` parameter will allow ADSM to successfully process file systems not using the **.** and **..** files.

8.2.2.4 MVS NFS

As was described in 8.1.2.3, “MVS NFS” on page 196, there are a number of different mechanisms for file organization under MVS. The actual file organization methods are:

- Partitioned Data Sets
- Partitioned Data Sets Extended
- OpenEdition Hierarchical File System

Within these organizational mechanisms, files can themselves be organized in several different ways:

- Physical Sequential

- Direct Access
- Virtual Sequential Access Method
- Sequential Access Method

The internal organization of the files is not important for backup or archive operations, nor is the permission or ownership, outside of the fact that the user running the client must have authority to read the files and write to the directories. Thus, it is the file organization method that is important.

Partitioned Data Sets: ADSM can backup and archive files from within a PDS. The first level of the hierarchy is not accessible to ADSM; instead, the data sets within must be specified. Thus, attempts to Backup by directory tree or Archive by directory tree will fail. Instead, use the **Backup by file specification** or equivalent archive option and specify the name of a PDS.

Partitioned Data Sets Extended: ADSM will backup and archive a PDSE in the same way as for a PDS.

OpenEdition HFS: With OE HFS, a UNIX-like file system is presented. The major issue with OE HFS file systems is one of permissions. Given that the local AIX user has sufficient permission to the directories and files that need to be backed up or archived, ADSM can be used normally to backup and archive files from directories within the HFS.

Permissions are granted, based on the userid used, when the `mvslogin` command is entered, prior to mounting the HFS file system. The MVS host systems programmer allocates authority to MVS users when the OE HFS is created.

8.3 File Storage Facility/6000

The setup and usage of FSF has already been covered in 7.2, “File Storage Facility/6000” on page 185. This section looks at using FSF with the remote file systems discussed in the first part of this chapter.

8.3.1 AFS

FSF is unable to mount its cache file system over an AFS directory. Even if this were possible, it would only be visible from the machine it was mounted from. It is possible that a volume server could use FSF space management to increase the size of the AFS directory tree that it makes available to AFS clients though this has not been tested.

8.3.2 AIX NFS

The scenario is to create a space managed file system, export it, and then mount it on other client machines in the network. In this way, a single FSF client machine can provide a space managed directory to many other clients in a network.

The steps need to be performed in the following order:

1. Create an FSF file system

   Allow FSF to create the cache. If an existing file system is used, problems can arise when remotely mounting the file system through NFS, due to permission problems.

2. Mount the FSF file system
3. Export the FSF file system
4. Mount the file system at the client machines

It is advisable to segregate client usage of the space managed, exported directory. This is most simply accomplished by creating subdirectories within the exported directory for each client. FSF will not manage concurrent access issues, nor does NFS. By exporting these subdirectories, and mounting one on each client, any such problems can be avoided.

8.3.3 VM NFS

NFS-mounted VM minidisks cannot be used as the server store by FSF. A FSF file system can be created for a NFS-mounted VM minidisk, but errors occur when FSF attempts to mount the MFS file system over the existing NFS mount. FSF would, in any case, be unable to trap file system activity to the VM minidisk, and so space management would be impossible.

Currently, the VM NFS server only provides server support; so an FSF file system cannot be mounted at the VM host.

8.3.4 MVS NFS

There are varying levels of support available from MVS with respect to FSF, depending upon the scenario.

8.3.4.1 Partitioned Data Sets

FSF cannot use an NFS-mounted PDS for the server store, and in common with VM, attempting to mount an FSF MFS file system over the MVS NFS mount results in errors. Furthermore, FSF would again not see activity within the data set; so space management would not be possible.

8.3.4.2 Partitioned Data Sets Extended

FSF interactions with a PDSE are the same as for a PDS.

8.3.4.3 OpenEdition HFS

FSF is able to use a MVS NFS-mounted HFS directory as its server store. This is highly dependent upon the permissions defined, for the directory mounted and userid used, with the `mvslogin` command. Given that this is all set up correctly (see 8.1.2.3, “MVS NFS” on page 196 for more information), FSF will copy files to and from the MVS HFS.

8.4 UniTree

The operational characteristics of UniTree preclude the usage of remote file systems. UniTree provides access to a file system at the server via either NFS or FTP for hierarchical file management and archive services. The set up and usage of UniTree is covered in 6.3, “UniTree” on page 168.

8.5 Legato NetWorker

Legato NetWorker provides backup services to client machines, and the set up and usage has already been covered in 5.5, “Legato NetWorker” on page 149. This section looks at backing up the remote file systems discussed in the first part of this chapter with NetWorker.

8.5.1 AFS

AFS file systems are visible from the NetWorker backup window. Selecting the icon for an AFS file system will show the files and directories within, at which point they can be selected for backup. Since AFS file systems are usually large, operations such as opening a subdirectory tree can take a long time.

8.5.2 AIX NFS

AIX NFS-mounted directories are also accessible to NetWorker and can be backed up and restored like a local file system. Access is, of course, still dependent upon permissions.

8.5.3 VM NFS

Locally NFS-mounted VM minidisks are also visible from the NetWorker backup window, and files stored on a minidisk can be selected and backed up. Since VM minidisks can only be accessed read/write by one user at a time, the minidisk must be released and detached from the VM system before a restore can occur.

8.5.4 MVS NFS

The types of MVS file system available have been discussed in 8.2.2.4, “MVS NFS” on page 204:

8.5.4.1 Partitioned Data Sets

Partitioned data sets can be accessed by NetWorker and files within backed up. Once again, permissions are important for restores to work.

8.5.4.2 Partitioned Data Sets Extended

Members of a partitioned data set extended are also able to be backed up by NetWorker.

8.5.4.3 OpenEdition HFS

Once mounted, the OE HFS file system looks exactly like any other part of the existing file system. Permissions are extremely important, and without the relevant read and write permissions, files are not visible or restorable. As discussed previously, the file access permissions are determined in relation to the userid specified in the `mvslogin` command.

8.6 Product Comparison Summary

This section will compare the various products discussed in this document based upon the remote file systems they support.

Table 4. Remote File System Comparison

Remote File System	ADSM/6000	FSF/6000	UniTree	Legato NetWorker
Andrew File System	Yes	No	No	Yes
AIX Network File System	Yes	Yes	No	Yes
VM Network File System	Yes	No	No	Yes
MVS NFS - PDS	Yes	No	No	Yes
MVS NFS - PDSE	Yes	No	No	Yes
MVS NFS - OE HFS	Yes	Yes	No	Yes

Note:

- ADSM support of VM NFS requires the DOTDIRCHECK option set to no
- ADSM does not recognize the first level of a PDS or PDSE. Members within must be identified by name

Chapter 9. Server Hierarchical Storage Management

One of the primary functions that a storage management product should provide is Hierarchical Storage Management (HSM). This feature allows storage resource usage optimization by automatically locating data on the most efficient media type, as defined by the requirements. HSM applications automatically migrate infrequently accessed data from online storage media to less expensive media, such as locally attached tape and optical libraries.

Once the storage devices and media available have been evaluated and the business requirements considered in terms of:

- Cost
- Capacity
- Performance
- Portability

then storage devices can be organized into hierarchies and the storage management product set up in order to provide the most intelligent and efficient usage of the media.

For example, frequently accessed information or information that requires high performance access, like databases, should be located on fast disk devices. Information with less restrictive requirements, in terms of access and performance, can be stored on optical media, while backup, archive, or long-term information can be stored on tapes.

HSM products can automatically migrate less-used data down to the different storage levels in a way that is completely transparent to the end users, thereby providing high capacity online storage using less expensive media.

9.1 ADSTAR Distributed Storage Manager/6000

As has already been shown in 4.1.1, “Device Administration” on page 69, ADSM storage pools can be defined to form a hierarchy. When additional space in required on storage devices, ADSM migrates data to the next level in the storage hierarchy according to the migration threshold parameters specified by the system administrator. There is no limit on the number of storage pool hierarchies.

For disk storage pools, ADSM can also make use of caching to improve retrievability of files.

Collocation can be used in order to minimize the file restore time. With collocation, the server keeps all the files belonging to a client filespace on the minimum possible number of sequential storage volumes.

9.1.1 Migration Hints

When a user backs up or archives files from a client node, the server looks at the bound management class, and from the copy group, it selects the storage pool where files are to be stored. Files being migrated to the server from a client space managed cache are also directed to a storage pool based on information in a management class. The following storage pool attributes affect the server behavior:

access mode A storage pool can be updated to be accessed read/write, read-only or to be unavailable for users. In the latter two cases, the server will look to the next storage pool in the hierarchy

maximum file size The maximum file size allowed in that storage pool. If the file to be backed up or archived is greater than this value, the server will look to the next storage pool down in the hierarchy. This parameter is used to prevent inefficient use of space in a disk pool. Generally, this parameter should not be used for the last pool in the hierarchy

next storage pool The name of the next pool in the hierarchy. It will be used when:

- The storage pool has read only access
- The file is larger than the maximum file size
- The files in the pool must be migrated, as the high threshold value has been reached

collocation For sequential storage pools. If enabled, the server will try to place files belonging to the same client filespace together

9.1.2 Migration Threshold Parameters

ADSM sets the high and low migration threshold parameters to 90 percent and 70 percent respectively as the default. These parameters can be optionally modified with the command:

```
update stgpool POOL_NAME hi 80 lo 60
```

When a storage pool occupancy reaches the high migration threshold, the server looks for the client filespace that consumes the most resource in the pool and starts migrating all the files from every filespace belonging to that client. If the low migration threshold is reached or passed, it stops the process; otherwise it goes on with the client owning the next largest filespace. Details on storage pool occupancy can be displayed using the `query stgpool` command.

```
adsm> query stgpool

Storage      Device      Estimated  %Util  %Migr  High   Low  Next
Pool Name    Class Name   Capacity                Mig%  Mig%  Storage
                              (MB)                            Pool
-----------  ----------  ---------  -----  -----  ----  ----  -------
ARCHIVEPOOL  DISK            200.0    3.2    3.2    90    70  OPTICALPOOL
BACKUPPOOL   DISK          1,000.0   90.0   90.0    90    70  OPTICALPOOL
OPTICALPOOL  OPTICAL       1,000.0    0.0    0.0    90    70  TAPEPOOL
SPACEMGPOOL  DISK            100.0   14.1   14.1    90    70  ARCHIVEPOOL
TAPEPOOL     TAPE              0.0    0.0    0.0    90    70
```

In order to check on the progress of the migration process, the `query process` command can be used, as shown in Figure 107 on page 211.

```
adsm> q process

 Process Process Description  Status
  Number
-------- -------------------- ------------------------------------------------
       2 Migration            Disk Storage Pool BACKUPPOOL, Moved files: 1695,
                               Moved bytes: 73,179,136, Unreadable Files: 0,
                               Unreadable bytes: 0. Current File (bytes):
                               28,672
```

Figure 107. ADSM Migration Process

```
adsm> q stgpool BACKUPPOOL format=detailed

               Storage Pool Name: BACKUPPOOL
               Storage Pool Type: Primary
               Device Class Name: DISK
         Estimated Capacity (MB): 1,000.0
                           %Util: 98.5
                           %Migr: 64.7
                       High Mig%: 90
                        Low Mig%: 70
            Migration Processes: 1
              Next Storage Pool: OPTICALPOOL
          Maximum Size Threshold: No Limit
                          Access: Read/Write
                     Description:
           Cache Migrated Files?: Yes
                       Collocate?:
           Reclamation Threshold:
 Maximum Scratch Volumes Allowed:
   Delay Period for Volume Reuse:
           Migration in progress?: Yes
             Amount Migrated (MB): 150.79
Elapsed Migration Time (seconds): 1,726
         Reclamation in progress?:
 Volume Being Migrated/Reclaimed:
 Last Updated by (administrator): ADMIN
           Last Update Date/Time: 10/28/1994 12:45:20

adsm>
```

Figure 108. Requesting Detailed Information on a Storage Pool

For disk storage pools, the migration threshold values can be lowered when caching is enabled. Complete information on the values of parameters held for a particular storage pool can be obtained with the `format=detailed` option of the `query stgpool` command, as shown in Figure 108.

9.1.3 Caching to Disk Storage Pools

When caching is enabled, anytime a migration occurs, the server does not remove the original file from the higher-level pool; so any subsequent retrieval request can be quickly performed.

The cached copy remains in the disk pool until space is needed for new files. When the server reclaims space in the disk pool, the cached files are removed according to a priority based on the file age and space occupancy.

The system default is to have caching enabled. If caching is disabled, higher values may be set for the migration threshold parameters in order to avoid having files migrated too frequently.

9.1.4 Sequential Storage Pools

Migration from one sequential pool to another can occur when more than one sequential level has been defined in the hierarchy. This may be desirable when, for example, there are various sequential devices of differing performance capability, or a new device has just been installed, and data needs to be moved across from the old device.

There are some additional issues that may have to be considered in this case:

- The next storage pool must have read/write access
- Collocation must be set the same in both pools
- Operator presence may be required

  The server attempts to reclaim space from sequential storage before migrating files to another sequential pool; so operators may be required to handle the necessary mount operations.

9.1.4.1 Collocation

When collocation is enabled, the server attempts to use all available volumes in order to keep user files separated. When collocation is disabled, the server attempts to use all the available space on a volume before selecting a new one.

Collocation should be enabled if faster restore/retrieve operation is required because the number of volume mount operations will be reduced.

On the other hand, the operators will have to perform more mounts for backup/archive operations, and a higher number of sequential devices will be involved because volumes will not be fully used.

With collocation enabled, the server will look for a volume which has already been used for data from the requesting client. If one does not exist, the search order will be as follows:

- An empty predefined volume
- An empty scratch volume
- The emptiest volume containing other client data

With collocation disabled, the server will look for the first active volume with available space. If one does not exist, the search order is as follows:

- The volume containing the most data
- An empty volume

9.1.4.2 Reclamation Threshold

The ADSM server tries to limit file fragmentation on sequential volumes by starting the reclamation process. The amount of reclaimable space on a volume increases as files become more fragmented due to files becoming obsolete because of aging or version limits.

The server moves fragmented files off one volume to another volume in the same storage pool, thus making the original volume available. The volume becomes

reusable if it has been previously defined to a storage pool, or it's simply deleted if it was a scratch volume.

The reclamation threshold parameter can be defined by specifying a number from one to 100. The default is 60 percent. The reclamation process starts when the percentage of reclaimable space rises above the specified value. Reclamation can be prevented by setting the threshold to 100.

Refer to the *ADSM/6000 Administrator's Guide* for additional information about migration.

9.2 FSF/6000

The FSF product does not provide this function because it has been designed to manage data on the client file system and to move it automatically to a centralized server.

However, when FSF works together with ADSM, it uses ADSM storage pools as server store. In this case, the storage pool containing the client filespace can be configured to point to another storage pool in the ADSM hierarchy, allowing standard migration to occur when the filespace approaches a capacity threshold.

FSF can also be configured to work with UniTree by defining the UniTree file system as the FSF server store. In this way, the files on the client can be transparently migrated from the client disk to archival media on the server managed by UniTree.

9.3 UniTree

This product manages storage in a distributed environment using a hierarchical archiving system. Client users access the UniTree file system, located on the server machine, using UniTree NFS or FTP from their local workstations. User files are automatically migrated from the first-level storage (the disk cache) to the second-level storage (tape or optical devices).

UniTree file processing is transparent to the client users. When they need to access a file that resides only on second-level storage, the file is automatically copied back on the disk cache. However, if the installation does not provide automatic tape or optical libraries as second-level storage, this process requires operator intervention to mount the required device.

9.3.1 UniTree Services

The first-level storage for user files is the disk cache. This is not a regular AIX file system. It is located in the disk server logical volumes defined by the UniTree administrator, and it cannot be viewed with standard AIX commands. The most important property of the UniTree file system is that files can be of any length, the maximum file size being limited only by the total amount of free space in the cache.

The main operations performed by UniTree on the file system are:

- Migration
- Purging
- Caching

9.3.1.1 Migration

This is the process of copying files from the first to the second level of storage. Cache files are eligible for migration when they meet the specifications setup by the administrator. Refer to 4.3.3.1, “Tuning File Migration” on page 103 for detailed information.

When a file is migrated, its status changes from *disk* to *both* (it resides on both storage levels).

9.3.1.2 Purging

This is the process of deleting files from the first level of storage. A cache file is eligible when the following conditions are all satisfied:

- A copy of the file exists in the second level of storage
- The file has not been accessed for a long time, or its size is fairly large
- The cache free space is lower than a specified value, or the number of cache files is greater than a specified value

Refer to 4.3.3.2, “Tuning File Purging” on page 105 for details.

When a file is purged, its status changes from *both* to *archive* (it resides only on archive storage).

Users can prevent files from being purged by camping them. The camping operation enhances performance as it circumvents the usual wait for files that exist only on the archive media though it does consume disk cache space.

9.3.1.3 Caching

This is the process of copying files from the second to the first level of storage. It occurs automatically every time a user process tries to access an archived file. Caching requires operator intervention if automated libraries are not available at the site.

When a file is cached, its status changes from *archive* to *both* (it resides again in both storage levels).

9.3.2 Using UniTree

The access to the UniTree server machine is provided by UniTree versions of the standard NFS and FTP protocols. They provide the same services as the standard AIX versions plus additional capabilities. It is suggested that both the protocols be enabled at installation time in order to offer the maximum level of flexibility to the client users.

9.3.2.1 Using UniTree with FTP

To start a UniTree FTP session from the local workstation, enter:

```
$ ftp server_name 1021
Connected to server_name
220 Archive FTP server (UniTree Version 1.0) ready.
Name:
 ...
ftp>
```

The UniTree file system resides on the server. For this reason, if access to a UniTree-managed file is required, it needs to be copied to the local disk and, after any changes have been implemented, copied back to the server again.

Using the FTP `quote` command, UniTree commands can be issued from the local workstation, thereby locally managing the remote files.

FTP User Commands: The FTP `dir` command has been modified in order to also show the UniTree status of files.

```
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /usr/lpp/UniTree/bin/ddir (0 bytes).
-rw-------      1 marina    staff    archive   1607680 Nov 28 11:02 test1
-rw-------      1 marina    staff    archive   1607680 Nov 25 18:02 test2
-rw-------      1 marina    staff    both      1607680 Nov 28 11:06 test3
-rw-------      1 marina    staff    archive   1607680 Nov 28 11:06 test4
226 Trasfer complete.
ftp>
```

The following is a list of the UniTree commands that can be issued as an additional argument to the FTP `quote` command:

camp Camps a file, so that it cannot be purged from the disk cache. The file status changes to *disk+*, or *both+*

chgrp Changes the group associated with a file

chmod Changes the permissions of a file

chown Changes the owner of a file

gtrsh Displays the current settings for the trash can time-out interval. Not yet implemented in the current product level

ln Creates a symbolic link between a remote directory and the specified link

mstage Exactly like *stage* for multiple files

nmdup Displays/changes the number of duplicated copies when migrating files. Not yet implemented in the current product level

stage Copies a file back from second-level storage to the disk cache. The wait time can be specified. With a wait time of zero, other operations can be run without having to wait for the actual copy of the file from archive to disk

strsh Changes the current settings for the trash can time-out interval. Not yet implemented in the current product level

umask Set the file creation mask to the specified value. The default is 077

uncamp Uncamps a file so that it can be purged from the disk cache. The file status changes to *disk*

version Displays if versioning is on or off for the specified file and allows the current version setting to be changed

wait Can be on/off, thereby governing whether the system will wait (on) for the file to be copied from first- to second-level storage before returning control

FTP Command Examples: This example shows the usage of the `version` command, from within an FTP session, to display and change version information held for a file.

```
ftp> quote version test1
500 Version is set to on and number is 537805488
ftp> quote version test3
500 Version is set to off and number is 0
ftp> quote version test3 on
500 Version is set to on and number is 1
ftp>
```

Figure 109. UniTree FTP quote version Command

This example shows the usage of the `chmod` command, from within an FTP session, to change permissions on a file.

```
ftp> quote chmod 777 test1
250 UniTree CHMOD command successful.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /usr/lpp/UniTree/bin/ddir (0 bytes).
-rwxrwxrwx      1 marina    staff    archive   1607680 Nov 28 11:02 test1
-rw-------      1 marina    staff    archive   1607680 Nov 25 18:02 test2
-rw-------      1 marina    staff    both      1607680 Nov 28 11:06 test3
-rw-------      1 marina    staff    archive   1607680 Nov 28 11:06 test4
226 Transfer complete.
ftp>
```

Figure 110. UniTree FTP quote chmod Command

This example shows the usage of the `camp` command, used from within an FTP session, to keep a file in disk storage.

```
ftp> quote camp test3
250-Start camping...
250 UniTree CAMP command successful.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /usr/lpp/UniTree/bin/ddir (0 bytes).
-rwxrwxrwx      1 marina    staff    archive   1607680 Nov 28 11:02 test1
-rw-------      1 marina    staff    archive   1607680 Nov 25 18:02 test2
-rw-------      1 marina    staff    disk+     1607680 Nov 28 11:06 test3
-rw-------      1 marina    staff    archive   1607680 Nov 28 11:06 test4
226 Trasfer complete.
ftp>
```

Figure 111. UniTree FTP quote camp Command

This example shows the usage of the `ln` command, from within an FTP session, to create a link to a file.

```
ftp> quote ln /usr/lpp/UniTree/adm/bin /u/marina/util
250 UniTree LN command successful.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /usr/lpp/UniTree/bin/ddir (0 bytes).
-rwxrwxrwx      1 marina    staff    archive   1607680 Nov 28 11:02 test1
-rw-------      1 marina    staff    archive   1607680 Nov 25 18:02 test2
-rw-------      1 marina    staff    disk+     1607680 Nov 28 11:06 test3
-rw-------      1 marina    staff    archive   1607680 Nov 28 11:06 test4
lrwxrwxrwx      1 marina    staff    disk           24 Nov 28 17:02 util -> /usr
/lpp/UniTree/adm.bin
226 Transfer complete.
ftp>
```

Figure 112. UniTree FTP quote ln Command

9.3.2.2 Using UniTree with NFS

To start a UniTree NFS session, the following tasks need to be performed:

- Login as the root user
- Create a mount point for the UniTree file system
- Mount the UniTree remote file system, as shown in 3.3.4, “Running the Verification Test” on page 56

Users can now login as normal and change directory to their UniTree-owned remote directory. Remember that uids must be the same on both client and server machines in order to correctly access files.

Work on files is accomplished in exactly the same way as for local files; the only difference may be the speed with which files are accessed. Generally, the NFS access method is better than FTP (even if a bit slower) because the UniTree file system can simply be mounted, instead of copying files to and from workstations.

The disadvantage is that UniTree commands cannot be used locally; it is necessary to login to the server using a valid UniTree userid in order to issue UniTree commands. In this case, the FTP access method is better than NFS because the FTP `quote` command allows most of the UniTree commands to be issued from the local workstation.

9.3.2.3 Issuing Commands from the Server

From the local workstation, Telnet to the UniTree server and login with a valid UniTree userid. In the .profile, put the UniTree path /usr/lpp/UniTree/adm/bin. Verify, with the `mount` command, that the UniTree file system is mounted locally and `cd` to it.

Notice that most of the commands have already been described in 4.3.1, “Device Administration” on page 97. This is because, by default, UniTree allows any user to issue any of its commands. It is the UniTree administrator (the root user on the server) who should decide on policy as to general user access to commands.

NFS User Commands: The following is a list of the most common UniTree commands. For a complete list, refer to *UniTree System Administrator's Guide*.

camp Camps a file

filemap	Displays the archive media on which the specified file is located
forcemig	Forces the migration of files from the cache
forcepurge	Forces purging of files from the cache
mntdisp	Displays the current mount requests for archive media
newversion	If versioning is on, creates a new version of the specified file
statall	Displays the status of the disk cache
statup	Displays the status of the disk logical volumes
uls	Displays file system information
uncamp	Uncamps a previously camped file
unstc	Allows user trash can lifetimes to be displayed and changed
ustage	Caches a file, copying it from the archive media to the disk cache
vcp	If versioning is on, copies a file and creates a new version
version	Set versioning on or off for the specified file
whomnt	Displays both UniTree processes and mount request information
whosup	Displays UniTree processes
wtlog	Displays archive media currently in use

NFS Command Examples: The following example shows the usage of the `version` and `newversion` commands used on UniTree-managed files accessed via an NFS.

```
$ version new2
Version is set to on and number is 1
$ newversion new2
$ uls
-rw-r--r--      2 marina    staff    disk           35 Nov 29 10:07 .new2.941129
112351
drwxr-xr-x      2 marina    staff    disk         8192 Nov 29 11:23 .trash
-rw-r--r--      1 marina    staff    disk           35 Nov 29 11:23 new2
-rw-------      1 marina    staff    both+     1607680 Nov 28 11:06 test3
-rw-------      1 marina    staff    archive   1607680 Nov 28 11:06 test4
$
```

Figure 113. UniTree version and newversion Command Output

The following example shows the usage of the `filemap` command used on UniTree-managed files accessed via NFS.

```
$ filemap /UniRoot/u/marina/new2
"/UniRoot/u/marina/new2" is not in the tape subsystem.
$ filemap /UniRoot/u/marina/test3
/UniRoot/u/marina/test3L
  copy 1 is located in tape(s) AA0000.
  copy 2 is located in tape(s) AA0001.
$
```

Figure 114. UniTree filemap Command Output

The following example shows the usage of the `unstc` command with UniTree managed files accessed via NFS.

```
$ unstc
Could not open log file /usr/lpp/UniTree/adm/log/unstc.t
Will print log information to standard out
unstc:    getexp marina
     marina, 30 minutes
unstc:    setexp marina 60
     Setting lifetime for marina to 60 minutes.
unstc:    quit
$
```

Figure 115. UniTree unstc Command Output

The warnings are displayed only when the command is issued by a general user who does not have write access permission to the log file.

9.4 Legato NetWorker

The Legato NetWorker product does not provide this function since it is essentially a network backup product.

9.5 Product Comparison Summary

The products discussed in this redbook will be compared with regard to their hierarchical storage management support on the basis of the following criteria:

- Function
- Ease of use

9.5.1 Function

This subsection looks at the level of function provided by the products discussed in this redbook that support hierarchical storage management.

9.5.1.1 ADSM

ADSM provides the following hierarchical storage management functions:

- Hierarchies

  Within ADSM, storage pools can be linked to form as many hierarchies, of arbitrary depth, as required. The composition, in terms of pool type, is also completely flexible.

  Storage pools within a hierarchy can be defined read-only, in which case the next level is considered. They can also be limited to files under a certain size.

- Collocation

  Sequential storage pools in a hierarchy can have collocation enabled, which means that ADSM will attempt to locate all of a particular user's files on the same volume within a pool.

- Migration parameters

Based on user-definable storage pool occupancy parameters, ADSM will migrate all files from the largest filespace to the next level of the hierarchy. If the lower threshold is not reached, this continues with the next largest filespace, and so on.

- Caching

  If disk caching is enabled, when migration occurs, ADSM will not remove the original file from the higher level until occupancy exceeds the migration threshold. When this occurs, cached files are removed based on size and age. This helps improve performance.

- Reclamation

  This process operates on sequential storage pools at regular intervals in an attempt to reduce fragmentation.

9.5.1.2 FSF

FSF does not provide hierarchical storage management.

9.5.1.3 UniTree

UniTree provides the following hierarchical storage management functions:

- Hierarchies

  UniTree allows the creation of a single two-layer hierarchy where the top layer is disk and the second layer tape or optical.

- Unlimited file size

  UniTree creates a new file system independent of the regular AIX file systems on the disk cache. Within this file system, file size is limited only to the size of the cache free space.

- Migration

  This process occurs based on certain criteria applying to the disk cache. When these criteria are met (file size or file age for example), files are migrated to the second level of the hierarchy. They are not removed from the disk cache until the purge operation occurs.

- Purging

  This process occurs for files when a copy exists on the second level of storage (migration), when the file is large or has not been accessed for a long time or when space utilization reaches a predefined limit.

- Camping

  Files can be made immune to the purging process by camping them in the first level of storage.

- Caching

  This process occurs when a file on the second level of storage is accessed and requires that it be copied to the first level for access.

- Manual operations

  Users can manually camp files to improve performance. Files can also be manually staged prior to an access requirement. This initiates the copy from second- to first-level storage. Purging and migration can also be initiated manually.

- Versions

  Multiple versions of files can be maintained.

- Status commands

  Various status commands allow the location of files within the hierarchy to be determined and the hierarchy status itself to be shown.

9.5.1.4 Legato NetWorker

NetWorker does not provide hierarchical storage management.

9.5.2 Ease of Use

This subsection looks at the relative ease of use of the products discussed in this redbook that provide hierarchical storage management.

9.5.2.1 ADSM

ADSM hierarchical storage management is an automatic process. Administration of the product is via GUI or command line interface, but apart from this, the process occurs transparently to users. It particularly enhances the efficiency of the archive and space management functions.

With ADSM Version 2, it is possible to use the central scheduling facility to start the hierarchical storage management migration process by artificially lowering the migration thresholds temporarily to zero when the migration process is required to start.

9.5.2.2 FSF

FSF does not provide hierarchical storage management.

9.5.2.3 UniTree

UniTree hierarchical storage management does not provide specific user interfaces either, though in order to access the hierarchy, files must be moved into it via FTP or NFS at the client machine. A high degree of control over the location and movement of files within the hierarchy is available to the user.

9.5.2.4 Legato NetWorker

NetWorker does not provide hierarchical storage management.

Appendix A. Supported Platforms

The platforms supported for client functions by the products discussed in this redbook are shown in Table 5. For ADSM Version 2, this covers the backup/archive clients only.

Table 5. Storage Products Supported Client Platforms

Client Platform	**ADSM/6000**	**FSF/6000**	**UniTree**	**Legato NetWorker**
Apple Macintosh	Yes	No	No	No
AT&T GIS	Yes	No	Yes	No
DEC Ultrix86/Open Desktop	Yes	No	Yes	Yes
DOS	Yes	No	No	Yes
Hewlett-Packard HP-UX	Yes	No	Yes	Yes
IBM AIX	Yes	Yes	Yes	Yes
IBM OS/2	Yes	No	No	Yes
Microsoft Windows	Yes	No	No	No
Microsoft Windows/NT	Yes	No	No	No
MVS OpenEdition HFS	Yes	No	No	No
NEC EWS-UX	Yes	No	No	No
Novell NetWare	Yes	No	No	Yes
SCO UNIX 386/Open Desktop	Yes	No	No	Yes
Siemens Nixdorf SINIX	Yes	No	No	No
Silicon Graphics IRIX	Yes	No	Yes	Yes
Sony 68K and MIPS	No	No	No	Yes
Sun Microsystems SunOS/Solaris	Yes	Yes	Yes	Yes
UnixWare	No	No	No	Yes

Appendix B. Supported Devices

B.1 Tape Drives

This appendix lists the tape devices supported by the products discussed in this redbook in Table 6 and the optical devices supported in Table 7 on page 226. This list is constantly changing, so if a device is not listed, the relevant source should be contacted to check if support is now available. For ADSM, either check on the MKTTOOLS forum, or call the IBM Information Support Center on 1-800-IBM-3333 and ask for STAR 20.

Table 6. Storage Products Supported Tape Drives

Device	Device Type	Capacity	ADSM/6000	UniTree	Legato NetWorker
ADIC-1200c	4mm	12 slots	No	No	Yes
ARC-4582/4584NP	4mm	4/12 slots	No	No	Yes
IGM ATL	8mm	56 slots	No	Yes	Yes
Exabyte EXB-8200/8205	8mm	2.3GB	Yes		Yes
Exabyte EXB-8500/8500C/8505	8mm	5.0GB	Yes		Yes
Exabyte EXB-60	8mm library	50GB	Yes		Yes
Exabyte EXB-10i/EXB-10e	8mm library	50GB	Yes	Yes	Yes
Exabyte EXB-120	8mm library	580GB	Yes	Yes	Yes
Hewlett Packard HP-C1553A	4mm	6 slots	No		Yes
IBM 7206-001/005	4mm	4GB	No		Yes
IBM 7207-001/011/012	1/4-inch	1.2GB	No		Yes
IBM 7208-001/011	8mm	2.3-5GB	Yes	Yes	Yes
IBM 3480	1/2-inch cartr	200MB	Yes		No
IBM 3490/3490E	1/2-inch cartr	800MB	Yes		No
IBM 3494-L10	1/2-inch library	7.2TB	Yes		
IBM 3495	1/2-inch library	15.5TB	Yes		
IBM 9348-012	1/2-inch reel	160MB	No		Yes
LAGO DataWheel LS/380L	8mm library	270GB	Yes	Yes	Yes
STK-9709	1/2-inch cartr	288 slots	No		Yes
Storage Tek 4781/4780			No	Yes	No
Storage Tek ACS-4400	1/2-inch cartr		No	Yes	No

B.2 Optical Drives

Table 7. Storage Products Supported Optical Drives

Device	Capacity	ADSM/6000	UniTree	Legato NetWorker
Alphatronix		No	Yes	No
Docustore DISC		No	Yes	No
Hewlett-Packard (1716M based)		No	Yes	Yes
IBM 3995-A63	20GB	Yes	Yes	
IBM 3995-063	40GB	Yes	Yes	
IBM 3995-163	188GB	Yes	Yes	
IBM 7209-001	650MB	Yes		Yes
IBM 7209-002	1.19GB	Yes		No

Appendix C. Storage Products Summary

This appendix lists the functions supported by the products discussed in this redbook in Table 8.

Table 8. Storage Products Summary

Function	**ADSM/6000**	**FSF/6000**	**UniTree**	**Legato NetWorker**
Backup and Restore	Yes	No	No	Yes
Archive and Retrieve	Yes	No	Limited(FTP)	Yes(4.1) Limited(4.01)
Space Management (Client HSM)	Yes	Yes	Limited(NFS)	No
Remote File Systems	Yes	Some(NFS)	No	Yes
Migration (Server HSM)	Yes	No	Yes	No
Administration GUI	Yes	No	No	Yes
User GUI	Yes	No	No	Yes
Automation	All	Space Mgmt	Migration	Backup
System Availability	Mirroring and Backup	None	Mirroring and Limited Backup	Backup
Security	Yes	AIX	AIX	Yes

Glossary

Ablative. Ablative technology utilizes heat to remove a layer of some material In this context, it relates to writing information with a laser by using the lasers heat to burn away a layer of the recording medium, thereby representing a binary value.

Actuator. An actuator is the mechanical assembly that is responsible for moving the disk head back and forth across the disk surface.

Andrew File System. The Andrew File System combines file systems from participating machines to form a composite that can be mounted on client machines, where it is accessible from the local file system as a single subdirectory structure.

Application Programming Interface. An application programming interface is a set of function calls that provide external access to services implemented within an application. Other applications can then use the API to utilize these services, avoiding the necessity to re-implement them.

Archive. Archiving involves moving data from the location that it is usually accessed from (normally fast, expensive storage), to lower cost storage such as tape. Information is normally archived if access to it will be very infrequent. Contrast with retrieval.

Areal Density. This defines the density at which individual bits can resolved by the read head. This equates to the maximum bit density supported by the media.

Backup. Backup involves taking a copy of data, usually on some form of removable media, so that in the event that information is lost, it can be easily recovered. Contrast with restore.

Banding. Traditionally, writing of bits to a disk surface occurs in a regular fashion; thus the further in toward the center of the disk, the less information can be stored. Banding refers to a process of dividing the disk surface into a number of concentric regions. As the disk write head moves into regions closer to the center of the disk, the bit write frequency increases proportionally, thereby maintaining the bit density.

Cache. A cache is a area of extremely fast (usually expensive) memory that is used to maintain frequently accessed information, or store information temporarily. Caches are used in various parts of a computer system. In disk subsystem controllers for example, writes to disk will actually occur to the cache so that a completion return code can be quickly returned to the writing process. The actual write will occur from the cache when the subsystem has time to satisfy it. The CPU also maintains several caches where instructions and data can be pre-loaded while the current instruction is executing.

Continuous Composite Write Once. Continuous composite write describes the magneto-optical implementation of WORM. Erasure and rewriting are prevented by simply not allowing the functions to take place. Contrast with WORM.

CD-ROM. A CD-ROM is an optical disk that has information stored on it before it is distributed. The information is permanently stored and cannot be erased or rewritten.

Client. The client/server paradigm defines a client as an entity that makes use of facilities offered be a server. Contrast with a server.

Copy Group. A copy group contains parameters that control the generation and expiration times of backup and archive data for ADSM. Span of control is at file level.

Daemon. A daemon is a process that usually runs in the background providing services to other requesting processes.

File System. A file system is a high level entity that manages the storage of data. Through the file system, information can be organized within directories and files created, read, written, and erased.

Filespace. A filespace is an ADSM entity maintained on behalf of clients where client data is stored.

Hierarchical File System. This refers to an file system organizational method that involves levels that are accessed by moving from one to another starting at the top of the hierarchy. For example, the directory tree in a file system, where files are found by navigating from the top of the tree through a series of subdirectories, until the file is found.

Hierarchical Storage Management. At a higher level, treating the various storage technologies available as levels, with disk being the level for interactive access, through optical for intermediate, to tape for backup/archive, defines a storage hierarchy through which data can be migrated according to space and usage requirements. The process of managing this mechanism is known as hierarchical storage management.

JFS log. Every action that occurs within a file system is recorded into a log known as the journaled file system log. These actions include events such as

opening files, writes to files, and closing files. If the system should fail during operation, upon reboot, the file systems can be brought back to consistent states by replaying the information contained in the log.

Journaled File System. The Journaled File System is the main AIX file system implementation It defines a file system with a JFS log.

Logical Partition. A logical volume is composed of a number of logical partitions. Each logical partition maps directly to from one to three physical partitions where data is actually stored.

Logical Volume. A logical volume is an area of physical disk storage comprising a number of logical partitions. Logical volume can be written to directly, or a file system can be created within them.

Logical Volume Manager. The logical volume manager is a collection of device drivers, disk data areas, daemons, and management subroutines that collectively form a high level interface to disk storage. It provides functions for the creation, manipulation, access, and deletion of logical volumes.

Magneto Optical. Magneto optical technology utilizes a laser and an electromagnet to alter the state of the material in an optical disk in order to store information. This mechanism can provide the capability to read, write and erase information.

Management Class. The management class determines how backed up or archived data will be managed. It contains a backup copy group and/or an archive copy group. Client node data is bound to a specific management class thereby determining how its data will be managed.

Migratable File System. There are a number of implementations of migratable file systems, though each has essentially the same purpose. The mfs is a virtual file system that is mounted over a directory that is to be space managed. File system calls are then trapped by the mfs and processed accordingly. For example and open system call is trapped, the local cache checked to see if the file is actually resident, and if not, a retrieve from server store executed to bring the file back. th open can then be processed by the normal file system beneath.

Migration. Migration is the term used to describe the process of moving files from one location to another. It is used mainly in two scenarios. The first is with regard to hierarchical storage management. Files are migrated or moved from one level of the hierarchy to another, based upon certain criteria, such as file age. The second is with regard to space management where files are migrated from the local space managed cache to a server store based upon certain criteria, such as cache occupancy.

Network File System. The network file system is an implementation of a remote file system Files stored on a remote machine are made to appear as if they were being accessed from the local machine.

Occupancy. Occupancy refers to the level of usage of an area of storage. The higher the occupancy, the fuller the storage, and vice versa.

Physical Partition. A physical volume is divided up into a number of physical partitions whose size is defined when the volume group containing the disk is created. These partitions are then mapped to logical partitions when a logical volume is created.

Physical Volume. Before a physical disk can be added to a volume group, it must be defined as a physical volume. This process assigns the disk a unique number by which it will be identified, and creates some on-disk data areas which are used to store information regarding the disks usage.

Pinning. This term is used in the context of space management to refer to a file that cannot to removed from the local space managed cache. Files are usually pinned for performance reasons; when required, they will not need to be copied back first.

Policy Domain. A policy domain defines a logical collection of clients that are all working according to the same set of policy needs. It thus provides a mechanism for managing backup and archive policies for groups of ADSM client nodes. Policy domains contain policy sets.

Policy Set. A policy set defines a collection of management classes. Only one policy set within a policy domain can be active at any one time.

Pruning. This term is used in the context of space management to refer to the process of removing files from client cache to free up space. Files can only be pruned if a copy exists at the server store. Pruning can be instituted manually or on the basis of a number of criteria, such as file age, or usage.

Restore. Restoring is the process of copying information back from its safe location (usually some form of removable media) to replace the original copy that has somehow been lost. Contrast with backup.

Retrieve. Retrieval is the process of moving data back from archive storage to its original location where it can be accessed. Contrast with archive.

Rotational Latency. When a block of data is to be read/written from a disk, the actuator moves the read/write head to the track where the block is located and then waits for the platter to rotate the start of the

block underneath so reading/writing can begin. This delay before the start of the block arrives is called rotational latency.

Seek Time. The seek time is the sum of the time taken for the disk head to be moved to the required track plus the rotational latency.

Server. The client/server paradigm defines a server as an entity providing services for requesting clients. These services can be any function performed on behalf of a client. Contrast with client. One server generally serves many clients.

Space Management. Space management is the process of managing the occupancy of client disk. Normally a cache is created at a client, and the space manager ensures that there is always free space in the cache for new files by moving existing files from the cache to a server store. If an existing file is required, it can be copied back for use.

Storage Pool. A storage pool can be viewed as an area of storage. Storage pools generally consist of storage volumes. An ADSM storage pool is also associated with a device class that indicates the physical type of storage volume.

Track. This defines a single one bit wide stream of physical data written on a storage medium. Tracks are concentric circles on disk and most optical media, horizontal lines on longitudinal technology tape, and inclined lines on helical scan technology tape.

Volume Group. This is a logical volume manager entity that contains a number of physical volumes.

Volumes. A storage volume is an entity such as a physical disk, a tape cartridge, an optical disk, or a file. Storage pools consist of a number of storage volumes.

Write Once Read Many. Optical media that utilizes a destructive writing process meaning that once written, information cannot be erased. Contrast with CCW.

List of Abbreviations

ADSM	Adstar Distributed Storage Manager
AFS	Andrew File System
API	Application Programming Interface
CCW	Continuous Composite Write-Once
CD-ROM	Compact Disk-Read Only Memory
CLI	Command Line Interface
DA	Direct Access
ESDS	Entry Sequenced Data Set
FSF	File Storage Facility
GUI	Graphical User Interface
HFS	Hierarchical File System
HSM	Hierarchical Storage Management
IBM	International Business Machines Corporation
ITSO	International Technical Support Organization
JFS	Journaled File System
KB	Kilobytes
KSDS	Key Sequenced Data Set
LP	Logical Partition
LV	Logical Volume
LVM	Logical Volume Manager
MB	Megabytes
MFS	Migratable File System
NFS	Network File System
PS	Physical Sequential
PDS	Partitioned Data Set
PDSE	Partitioned Data Set Extended
PP	Physical Partition
PV	Physical Volume
RDBMS	Relational Data Base Management System
RRDS	Relative Record Data Set
SAM	Sequential Access Method
TCP/IP	Transmission Control Protocol/Internet Protocol
VG	Volume Group
VSAM	Virtual Sequential Access Method
WORM	Write Once Read Many

Index

ITSO Technical Bulletin Evaluation
RED000

AIX Storage Management Products Comparison

Publication No. GG24-4495-00

Your feedback is very important to help us maintain the quality of ITSO Bulletins. **Please fill out this questionnaire and return it using one of the following methods:**

- Mail it to the address on the back (postage paid in U.S. only)
- Give it to an IBM marketing representative for mailing
- Fax it to: Your International Access Code + 1 914 432 8246
- Send a note to REDBOOK@VNET.IBM.COM

Please rate on a scale of 1 to 5 the subjects below.
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction ____

Organization of the book	____	Grammar/punctuation/spelling	____
Accuracy of the information	____	Ease of reading and understanding	____
Relevance of the information	____	Ease of finding information	____
Completeness of the information	____	Level of technical detail	____
Value of illustrations	____	Print quality	____

Please answer the following questions:

a) If you are an employee of IBM or its subsidiaries:
   Do you provide billable services for 20% or more of your time? Yes____ No____
   Are you in a Services Organization? Yes____ No____

b) Are you working in the USA? Yes____ No____

c) Was the Bulletin published in time for your needs? Yes____ No____

d) Did this Bulletin meet your needs? Yes____ No____
   If no, please explain:

_____

_____

What other topics would you like to see in this Bulletin?

_____

_____

What other Technical Bulletins would you like to see published?

_____

Comments/Suggestions: **(THANK YOU FOR YOUR FEEDBACK!)**

Name _____ Address _____

Company or Organization _____

Phone No. _____

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM International Technical Support Organization
Department 948, Building 821
Internal Zip 2834
11400 BURNET ROAD
AUSTIN TX
USA 78758-3493

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold
Along Line

Printed in U.S.A.