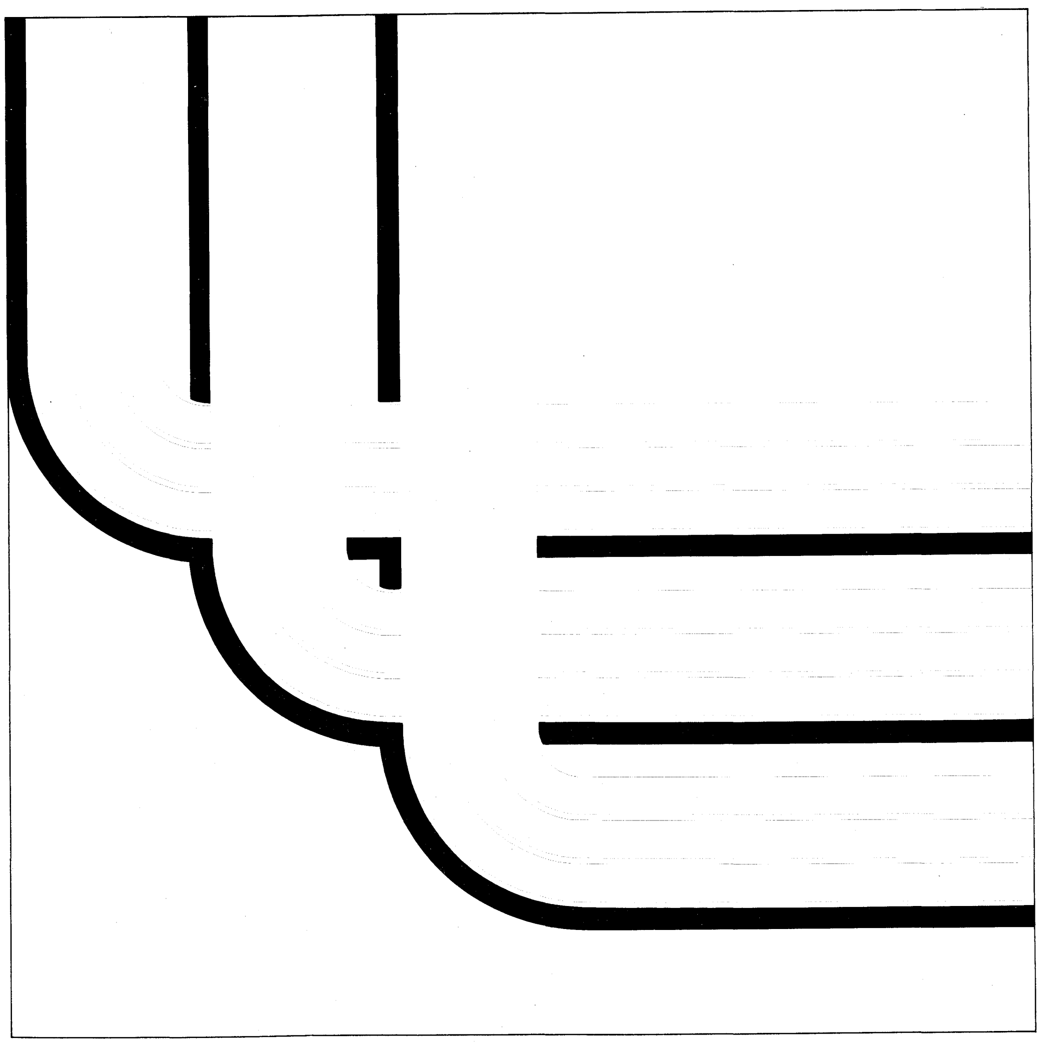
Application System/400™

SC41-8079-00

Backup and Recovery Guide

Version 2



System and Application Support

Take Note!

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

First Edition (May 1991)

This edition applies to the licensed program AS/400 Operating System/400 (Program 5738-SS1), Version 2 Release 1 Modification 0, and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the proper edition for the level of the product.

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

A form for readers' comments is provided at the back of this publication. If the form has been removed, you may address your comments to:

Attn Department 245
IBM Corporation
3605 Highway 52 N
Rochester, MN 55901-7899

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 or restricting your use of it.

Contents

Part 4. Auxiliary Storage Pools and Disk Recovery

Part 5. Checksum Protection

Part 6. Mirrored Protection

Figures

Notices

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 licensed program or other IBM product in this publication is not intended to state or imply that only IBM's program or other product may be used.

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 Commercial Relations, IBM Corporation, Purchase, NY 10577.

The following terms, denoted by an asterisk (*) in this publication, are trademarks of the IBM Corporation in the United States and/or other countries:

Application System/400	AS/400	C/400
COBOL	COBOL/400	FORTRAN/400
IBM	OfficeVision/400	Operating System/400
OS/400	PL/I	PS/2
RM/COBOL-85	RPG III	RPG/400
SQL/400	400	

This publication could contain technical inaccuracies or typographical errors.

This manual may refer to products that are announced but are not yet available.

Information that has changed since Version 1 Release 3 Modification 0 is indicated by a vertical bar (|) to the left of the change.

This publication is for planning purposes only. The information herein is subject to change before the products described become available.

This publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include 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.

This manual contains small programs that are furnished by IBM as simple examples to provide an illustration. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. All programs contained herein are provided to you "AS IS". THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.

The drawings and specifications contained herein shall not be reproduced in whole or in part without the written permission of IBM.

IBM has prepared this manual for use by hardware service representatives in the maintenance or repair of the specific machines indicated. IBM makes no representations that it is suitable for any other purpose.

Programming Interface

The *Backup and Recovery Guide* is intended to help the customer do backup and recovery planning for the system. It contains information to plan a backup and recovery strategy, procedures to implement this strategy, and procedures to recover from disk unit failures. The *Backup and Recovery Guide* contains no programming interfaces for customers.

About This Manual

This manual provides information about backup and recovery planning for the system. Backup and Recovery planning and implementation is the responsibility of the customer.

This manual provides:

- Information to plan a backup and recovery strategy.
- Procedures to implement your backup and recovery strategy
- Procedures to recover from disk unit failures

This manual is intended for someone who is assigned the responsibilities of backup and recovery planning and recovering the system after a failure.

You should be familiar with the information contained in the *System Operator's Guide*, SC41-8082, and the *New User's Guide*, SC41-8211, before using this manual.

If you know how to operate the system, you should be ready to use this manual to plan for, and implement, a backup and recovery strategy.

This manual does not describe application recovery.

Some of the information in this manual applies to Version 2 Release 1 Modification 1, which will be available at a later date. Version 2 Release 1 Modification 1 is also referred to as Version 2 Release 1.1 or V2R1.1.

You may need to refer to other IBM manuals for more specific information about a particular topic. The *Publications Guide*, GC41-9678, provides information on all the manuals in the AS/400 library.

For a list of related publications, see the "Bibliography."

Part 1. Developing a Backup and Recovery Strategy

Chapter 1. Introduction to Backup and Recovery Strategy Planning

The continued growth in dependence on computers, and on the information from them, has increased the importance of the plans to prevent loss of their availability. It is becoming very difficult, costly, and confusing to revert back to manual systems for any length of time. A backup and recovery strategy is necessary to minimize the damage that might be caused by unexpected and undesirable occurrences affecting information services or other parts of the organization's information systems.

The objective of a backup and recovery strategy for information services is to make sufficient preparations, and to establish a sufficient set of agreed upon procedures, for responding to a disaster or emergency. This will minimize the effect upon the operation of the business.

The purpose of this chapter is to provide an overview of the backup and recovery options available for the AS/400* system.

Backup and Recovery Options Available for the AS/400 System

This topic provides a high-level view of the backup and recovery options available for the AS/400 system. This information provides a base for understanding the backup and recovery options discussed later in Chapter 2, "Planning a Save and Restore Strategy." These methods include:

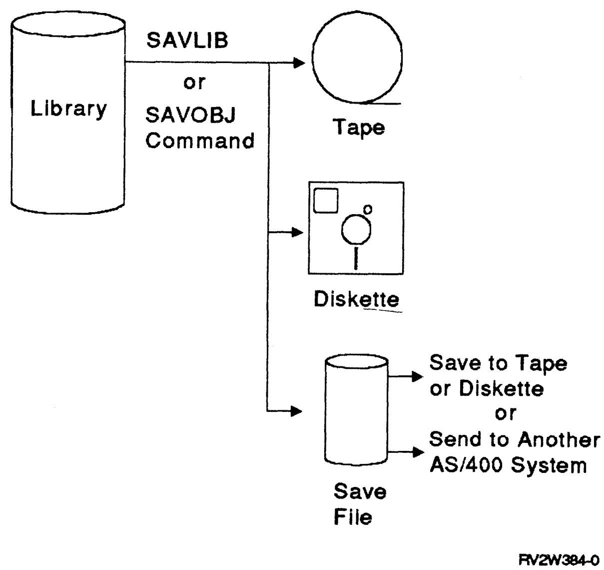
- Save and restore operations of the system, storage, and objects
- Journal management
- Access path journaling
- Commitment control
- Auxiliary storage pools
- Checksum protection
- Mirrored protection
- Uninterruptible power supply
- Battery power unit
- Dual systems

Detailed descriptions of these methods are discussed later in this guide.

Save and Restore Operations

The save and restore operations provide a means to recover from a program or system failure, save the system storage, exchange information between systems, and store infrequently used objects offline. Normally, objects are frequently saved and infrequently restored. Objects can be saved and restored using diskette, magnetic tape, or a save file.

The following shows an overview of objects being saved to a diskette, a magnetic tape, and a save file.



Consider the following when using the save and restore operations:

- When information is saved, a copy of the information in a special format is written onto one or more diskettes, reels of magnetic tape, tape cartridges, or to a save file. The save and restore format does not follow standards for normal data interchange on systems other than the AS/400 system. However, diskettes and tape can be removed and stored for future use on the same system or distributed to another AS/400* system. New tapes and diskettes must be initialized.

- A **save file** is allocated in auxiliary storage to store data in a file (without an operator present to change the media). It also allows input and output (I/O) operations from a high-level program, or receives objects sent through the network.

  Save files can be used in two ways:

  - Transmission to another AS/400 system over communications lines.
  - Unattended save operations because an operator does not need to load diskettes or tapes when information is being written to the save file.

- When information is restored, the information is read from diskette, tape, or a save file into storage where it can be accessed by system users. The information restored can be either old information previously saved from the system or new information saved from another AS/400 system.

Journal Management

Two objects unique to journal management are the journal and the journal receiver:

- The **journal receiver** (object type *JRNRCV) is an object that contains entries (called journal entries) written when a change is made to a file or access path that is being journaled.
- The **journal** (object type *JRN) identifies the journaled database files, the current journal receiver, and all journal receivers that are on the system for the journal. Only database files can be journaled.

Journal entries identify activity for a specific record (added, changed, or deleted) and for save operations for a file or file member, such as an open, close, or save operation. Each entry includes additional control information identifying the source of the activity, including the user, job, program, time, and date. The entire image of the database record is journaled (not just the changed information). Journal receiver entries can contain before-images as well as after-images for changes that affect a single record.

Using journal management is a way to recover database files. When a change is made to a journaled file, the changes are first recorded in the journal receiver. If the system ends abnormally or the file becomes damaged before the change is actually written to the record, the change may have been saved in the journal receiver. If the database file is damaged, you can restore the file from the save media and then apply the journaled changes.

Figure 1-1 on page 1-3 shows journal processing. Files A and B are being journaled, file C is not. Programs PGMX and PGMY use file B. When you make a change to a record in file A or B, the following occurs:

1. The change is written in the active journal receiver.
2. The journal receiver is written to auxiliary storage.
3. The record is written to the file.

File C record changes are written directly to the main storage copy of the file because it is not being journaled. Only the changes made to the journal receiver are written immediately to auxiliary storage (disk). Changes against the physical file may stay in main storage until the file is closed.

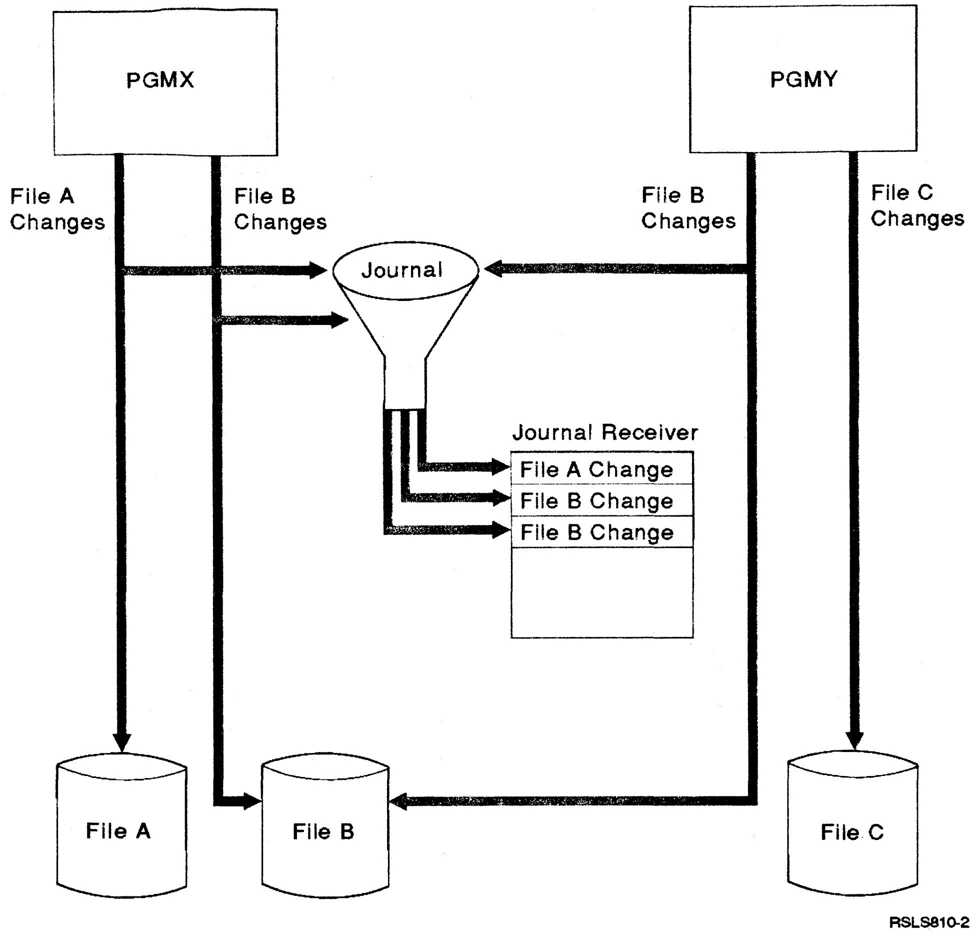


Figure 1-1. Journaling Overview

The system journals some file-level changes, including moving a file and renaming a file. The system also journals member-level changes, such as initializing a physical file member, and system-level changes, such as initial program load (IPL). Users can also add their own entries to a journal receiver to identify significant programming events or to help in the recovery of their programs.

Access Path Journaling

An **access path** describes to the system the order in which records are read. If access path changes are not journaled in the journal receivers, and a file has been changed and not closed properly, the system can spend a significant amount of time rebuilding the access paths during the IPL following an abnormal system end. If the decision is made to use access path journaling, and the system fails, the system may recover (rather than rebuild) the access paths automatically, using the information in the journal. This can greatly decrease the time required to recover. On the AS/400 system, journaling access paths is the primary method of reducing the time to rebuild access paths.

Commitment Control

Commitment control is an extension of the journal function on the AS/400 system. It allows the defining and processing of a number of changes to database files as a single transaction.

Without commitment control, recovering data for a complicated program requires detailed program knowledge and programs cannot easily be started again. To restore the data to the last completed transaction, the data may need to be adjusted by a user program or data file utility (DFU) that reverses the transactions that are not complete. This becomes more difficult when many users are accessing the files. For example, when the system or a job ends abnormally, journaling assures that all records will exist in the database. Because end-user applications can require multiple changes to files in a transaction, journaling may reflect only a partially completed transaction.

You can use commitment control to ensure:

- All changes within a transaction are completed for all files affected.
- All changes within a transaction that are not complete are removed if processing is interrupted.
- Changes made during a transaction can be removed when the user program determines that it is necessary to do so. This is called a rollback operation.

The commit (COMMIT) and rollback (ROLLBACK) operations are available in several AS/400 programming languages including RPG III*, COBOL*, PL/I*, C/400*, control language (CL), and Structured Query Language (SQL).

Auxiliary Storage Pools (ASPs)

An **auxiliary storage pool (ASP)** is a group of units defined from all the disk units that make up auxiliary storage. ASPs provide the means of isolating objects on a specific disk unit to prevent the loss of data due to a disk media failure on other disk units not included in the ASP.

The system ASP (ASP 1) is created by the system and is always configured. It contains the licensed internal code, licensed programs, and system libraries. The system ASP also contains all other configured disk units that are not assigned to a user ASP.

A user ASP is created by grouping together a physical set of disk units and assigning them a number 2 though 16. ASP 1 is always reserved as the system ASP.

User ASPs can be used to isolate libraries and objects within these libraries from the system ASP. If a library exists in a user ASP, all objects in the library must be in the same ASP as the library. "Object Types Not Allowed in a User ASP" on page 14-10 shows a list of object types that are not allowed in a user ASP.

The exceptions to this rule are journals, journal receivers, and save files. These object types can be placed in user ASPs when their libraries are in the system ASP and the selected user ASP contains no libraries. However, this type of ASP is not recommended because of the complex recovery steps. These object types can also be placed in libraries in user ASPs.

In addition to the recovery advantage, placing libraries and objects in a separate user ASP can improve performance. In a heavy journaling environment, isolating journal receivers to a user ASP can reduce disk arm contention between the files and journal receivers, and improve journaling performance.

Note: If journal receivers and save files are not put in a separate user ASP, consider saving them regularly on tape or diskette to protect them from loss due to disk failure.

Checksum Protection

Checksum protection is a function that protects data stored in an auxiliary storage pool from being lost because of damage or a disk unit media failure. When checksum protection is in effect and a disk unit media failure occurs on a protected unit, the system automatically reconstructs the data after the disk unit is repaired.

Damage to objects can occur because of slight imperfections on a disk surface. If this occurs to objects in the ASP when checksum protection is in effect, the data is automatically re-created. This avoids having the system mark the object as damaged.

When checksum protection is started, the system automatically groups the disk units in the ASP into checksum sets. Space equivalent to approximately one disk unit in each set is used to store checksum data that provides protection for the user data stored on the other units in the set.

The data residing on several disk units (checksum set) is combined onto other units in such a way that, if any one of the units fails, its contents may be recovered by recombining the data on remaining units in the checksum set. The reconstructed data reflects the most up-to-date information that was on the disk at the time of the failure.

When a disk unit fails, checksum protection does not prevent the system from ending abnormally. Rather, its main advantage is that no data is lost and it helps avoid installing the entire system and loading information to disk again if the failed unit must be replaced and the data on it is lost. In contrast, normal save and restore methods allow you to recover only to the point of the last save operation and result in entering many transactions again.

Any changes made to permanent objects residing in the system ASP are automatically updated and maintained in the checksum data (associated with the space allocated to the objects).

When checksum protection is started for a user ASP, only the ASP is loaded again. A Save Library (SAVLIB) command is run to save all the libraries in the user ASP. Libraries in the system ASP should not have objects in the user ASPs. If the libraries have objects in user ASPs, then the ASP is of the old type that contains only journals, journal receivers, and save files. Checksum protection is not recommended for user ASPs that contain only journals, journal receivers, and save files.

Figure 1-2 illustrates a system ASP configuration with checksum protection. In this example, assume that two of the three devices contain user application data, and the other device contains checksum data.

The assignment of units to checksum sets is shown for example purposes only. The actual assignment of checksum sets can vary because the system uses an algorithm to determine the checksum sets.

If the user data on unit 6 is lost, the system automatically reconstructs that data from unit 5 and the checksum data on unit 7 after unit 6 is replaced.

The actual implementation of checksum protection is more complex because there are multiple checksum areas spread across all units in a checksum set to distribute the disk activity more evenly over all units in a checksum set.

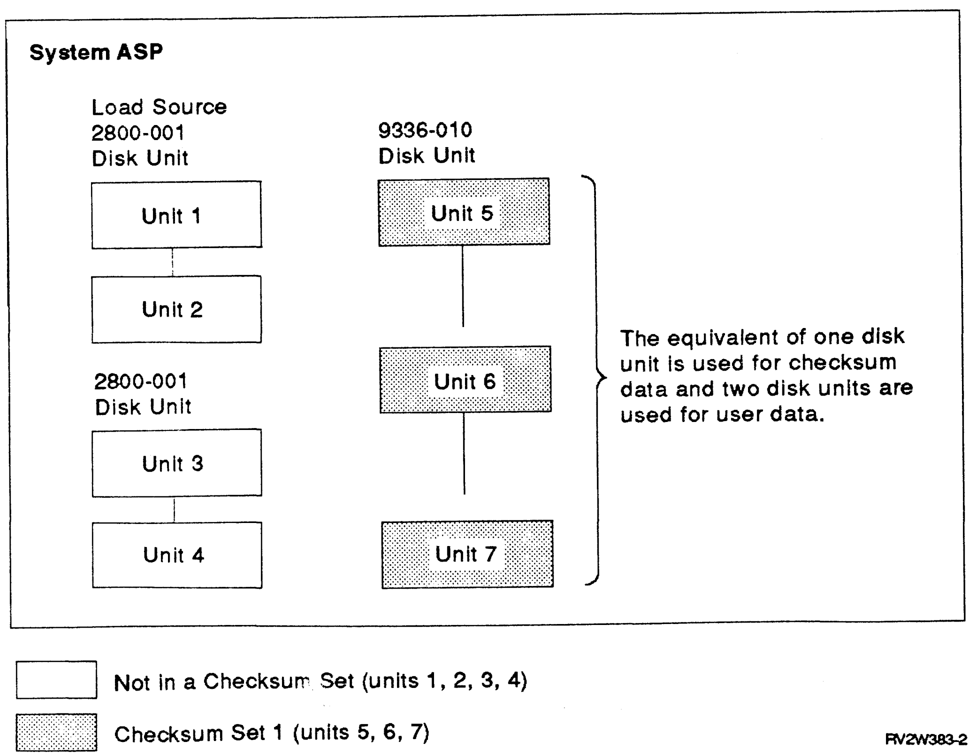


Figure 1-2. Example of Checksum Protection for the System ASP

Checksum protection should not be used as a replacement for system backup procedures. Although checksum protection can fully recover from most disk failures, it is important to be aware that there are situations where checksum protection will not be able to recover some or all of the data. For example, if more than one disk unit within the same checksum set fails and is lost or is damaged by fire, checksum protection will not be able to reconstruct any data. Even though these occurrences are rare, checksum protection should not be used as a substitute for saving the entire system on a regular basis.

Mirrored Protection

Mirrored protection is a function that increases the availability of the AS/400* system in the event of a failure of a disk-related hardware component. It can be used on any model of the AS/400* system and is a part of the licensed internal code. Different levels of mirrored protection are possible, depending on what hardware is duplicated. The system remains available during a failure of a disk-related hardware component such as a disk unit, a disk controller, a disk I/O processor, or a bus, if the failing hardware component and hardware components attached to it are duplicated. For the 9406 system unit, some failed hardware components can be serviced while the system remains available.

Note: It is not possible to have bus-level protection for unit 1 on a 9406 system unit because both units of the mirrored pair must be on the same bus.

Figure 1-3 shows a system that has bus-level protection.

The disk units in an ASP are automatically paired by the system when mirrored protection is started. The system pairs the disk units to provide the maximum level of protection for the current hardware configuration. Because all disk units have at least disk unit level protection, the system is protected against the failure of a single disk unit. In this example, if a controller or I/O processor failure occurs, the system continues to run.

Mirrored Busses

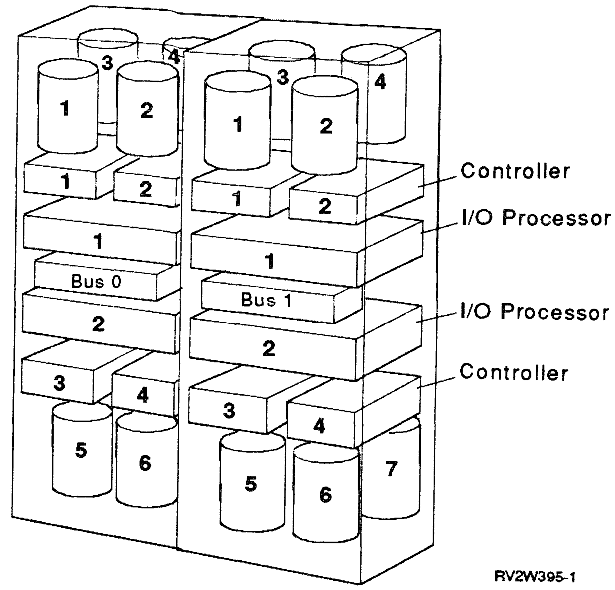


Figure 1-3. Example of a Mirrored Buses

Uninterruptible Power Supply

An uninterruptible power supply provides auxiliary power to the processing unit, disk units, and as many other devices on the system as possible.

Normally, an uninterruptible power supply does not provide power to all work stations. Assuming an uninterruptible power supply is not provided to the work stations if utility power is lost, work station jobs end abnormally but the system remains stable. After utility power is restored, the users can sign on to the work stations. An uninterruptible power supply that provides limited support provides power to the processing unit, unit 1, and all storage controllers. The system continues to run for a specified number of minutes.

Uninterruptible power supplies vary, but Figure 1-4 on page 1-7 shows a logical view of a typical uninterruptible power supply:

With the AS/400 system, the uninterruptible power supply provides the system with the ability to:

- Continue operations during brief power interruptions.
- Provide normal ending of operations so that the next time the system performs an IPL, there is minimal recovery time. If the system

ends abnormally before completing a normal ending of operations, the recovery time can be significant.

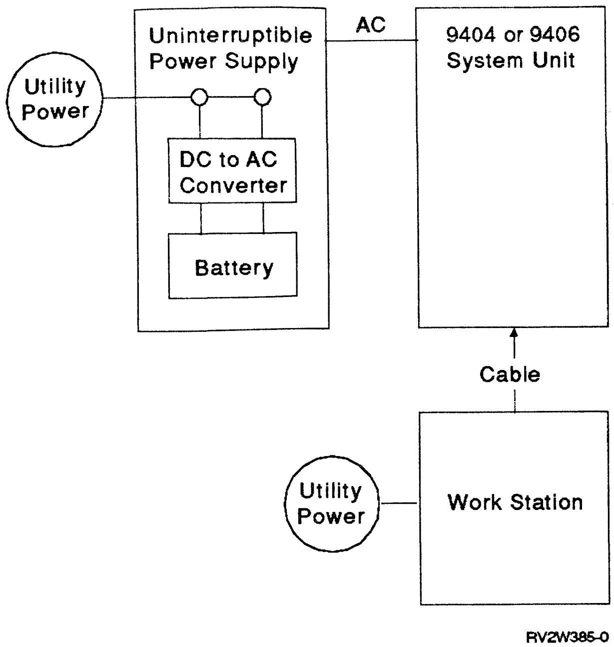


Figure 1-4. Logical View of a Typical Uninterruptible Power Supply

Battery Power Unit for the 9402, 9404, and 9406 Model D System Units

A battery power unit exists as an optional feature for the 9402 and 9404 system units and as a standard feature for the 9406 Model D system units. In order for the system unit to be protected against temporary power loss, each system unit must have a battery power unit. The Battery feature on the the 9402, 9404, or 9406 system unit supports all disk units in the system unit. The system unit has a battery with sufficient power to keep the processing unit and the disk unit powered for a minimum of five minutes in the event of a loss of utility power. For an illustration of the built-in Battery, see Figure 1-5.

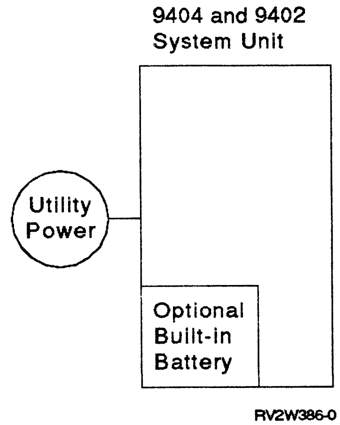


Figure 1-5. Built-in Battery Feature

Dual Systems

A dual system approach can be used to record changes on a secondary system as they occur on the primary system. The secondary system can take over critical application programs if the primary system fails. The OS/400 licensed program does not directly support this concept. However, certain functions of the AS/400 system can be used to achieve this.

Battery Power Unit Overview for the 9402, 9404, and 9406 Model D System Units

Chapter 2. Planning a Save and Restore Strategy

The purpose of this chapter is to describe the planning cycle: how to decide which recovery methods are worth using now, which should be considered later, and which do not apply at all.

The objective is to provide you with the knowledge that will help you develop or revise a backup and recovery plan that will balance the cost of a failure with the cost of protection against the failure.

This chapter does not describe or explain in detail how the backup and recovery options work or how they can be implemented. Other parts of the guide describe and explain the options in detail.

Backup and Recovery Guidelines

A comprehensive set of backup procedures is the foundation for any recovery strategy. These backup procedures should be a required part of your normal AS/400 operations. The goal of your strategy is to ensure that, following any type of failure, information exists that will allow you to recover your system.

Include backing up database files, programs, source code, operating and utility programs, office applications and documents, and end-user applications in your procedures.

Consider the following guidelines when planning your procedures:

- You should be prepared to restore or re-create any object that becomes damaged.
- You should be able to restore the entire system in the event a disaster occurs.
- You should have a plan for how you will recover any changes that have occurred since the time of the last save operation.
- Additional backup procedures can be useful for quarterly or yearly audits to help with business audits.
- A rotation method provides you with additional backup copies should your most recent set become damaged or unusable. Consider using at least a three-set rotational method for backup copies. For example, use your first set of tapes on Monday, your second set of tapes on Tuesday, and your third set of tapes on Wednesday. On Thursday, start the cycle again by using your first set of tapes.
- Protect your backup media from a site disaster. For example, rotate one of your backup sets to a different location and keep it in a fireproof and waterproof safe. Consider what would happen if your site were destroyed. Would you have a recent copy of your data to load on another system? If not, how long would it take you to manually recover your data and programs?
- Design backup and recovery into your applications. Your design should include as many automatic procedures as possible. Develop error handling routines in your programs for work stations, communications, and database files.
- Provide documentation and test your procedures to make sure they work correctly before you put them into production.
- As a minimum goal, save changes at least once a day. Save the journal receiver or transaction file that contains the latest changes to the production database files. Changes to source files, the document library, and other objects need to be saved at least once a day.
- Periodically save your system environment, including the operating system, device configuration objects, user profiles, all user objects, private authorities, and system and network values. Without this data, you would have to manually re-create your entire device configuration and security environment when building your system again.

System Availability

In any business, there are times the system is not available. The availability of the system is impacted by planned outages, such as your system maintenance, save and restore schedules, and the availability of power.

The shaded portion of Figure 2-1 represents normal operations. Historically, the shaded area continues to grow.

The following describes some terms used when discussing availability and the types of outages that can occur on a system:

- **Scheduled**

  System is down from a planned or predicted condition

  - System not needed.
  - System is dedicated to backup process.
  - New equipment or software is being installed or maintained.
  - Program temporary fix (PTF) package is being installed.

- **Unscheduled**

  System is down for some failure or error.

  Most of the time is spent waiting or getting the system operational again.

- **High Availability**

  No unscheduled outages

- **Continuous Operations**

  No scheduled outages

- **Continuous Availability**

  No outages

Scheduled and unscheduled outages may often be interrelated. Changes made to decrease unscheduled outage time may increase planned outage time, and vice versa. For example, starting checksum protection requires reloading the system. However, recovery from a disk unit failure will be much faster.

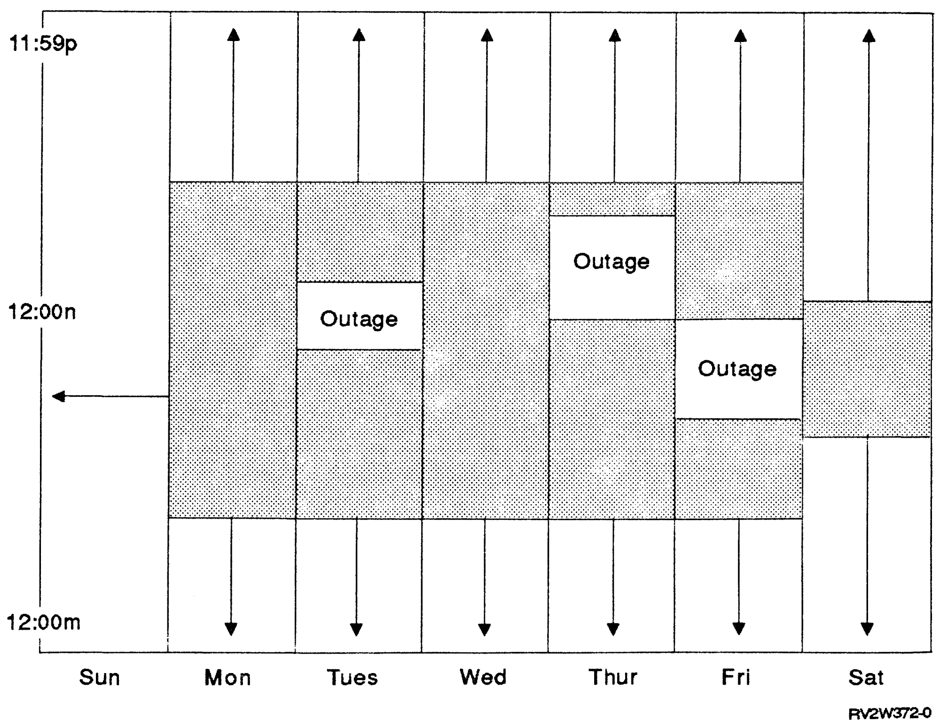


Figure 2-1. Outage Windows

Recovery Design Considerations

This topic discusses two processes: recovery and backup. It makes sense to look at the recovery requirements first because the recovery time generally has the most impact. Establishing what must be recovered and in what period of time usually dictates what must be saved and with what frequency. So the questions to be asked are:

1. What factors make the recovery process longer?
2. What factors or methods can help shorten the recovery?
3. Is it worth the cost of using the appropriate backup or recovery method in order to reduce the recovery time?

Once the recovery process and contingencies have been defined, then the backup procedures necessary to support the recovery requirements can be designed. Since many of the backup or save methods require some level of exclusive use of the system, it becomes appropriate to ask the same questions for backup as for recovery.

Cost of Being Down

The first essential step is to analyze the cost of being down. It is not easy to determine the cost of outages. The availability requirements for each application or program can have widely varying effects on the productivity of its users.

Analyze by Major Application: You must start with a reasonable estimate of what each critical application is worth to the business. Some applications are critical throughout major portions of the day, while others can be run any time or on demand.

- The **major cost** of an outage is the cumulative total of not having the applications available to continue necessary operations.
- **Direct costs** are the time and revenue lost directly because the system is down. **Indirect costs** are costs incurred by another department or function as a result of an outage. For example, the production department may absorb the cost of a manufacturing line being shut down because the system is unavailable. This is an indirect cost of the outage, but nonetheless a real cost to the company.
- **Tangible costs** are direct and indirect costs that can be measured in dollars and cents. **Intangible costs** are those for which cash never changes hands, such as lost opportunity, good will, and market share.
- **Fixed costs** are direct, indirect, tangible, or intangible costs that would result from the failure regardless of the length of the outage. **Variable costs** are those that vary with the duration of the down time, but they are not necessarily directly proportional.

Cost of Doing Backup and Recovery

The second essential step is to know how much it will cost to use the various recovery functions that are available on the AS/400 system. You can then evaluate whether the cost of the function is more expensive than the cost of being down.

The following topics evaluate the backup and recovery functions and what additional costs (if any) the functions can incur.

Save and Restore Operations: The following evaluates the costs of save and restore operations.

- Tape unit with adequate speed and capacity
- Additional disk units for on-line backup
- Operator time (or overtime)
- Programming effort to automate the save and restore processes
- Documentation of save and restore procedures

As the recovery requirements become more critical, the amount of data and the frequency that it must be saved increases. To help reduce the amount of time spent during the save process, review the following factors that affect save-time performance.

Number and Size of Objects: In most cases, saving many small objects takes more time than saving larger objects that are equal in total size.

System Activity during the Save Operation: Save performance is also affected by whether the system is in a restricted or non-restricted state. A **restricted state** is when all subsystems are ended, including the controlling subsystem (also referred to as a dedicated system). The system is considered in a **non-restricted state** if any or all subsystems are active, regardless of whether there are any active jobs or users at the time. A restricted system can offer the following benefits:

- Object lock checking can be bypassed.
- Disk contention with other jobs is reduced.
- Sufficient processing unit and main storage resources are available.

In many circumstances, it is not necessary to place the system in a restricted state to obtain good performance from these commands. However, if you determine that other active jobs have a noticeable affect on save performance, or that the save operation is having noticeable affect on active jobs, consider running the save operation in a restricted environment.

Type of Save Operation: When saving large amounts of data, the type and number of save commands used can affect performance. For example, saving multiple libraries using a single SAVLIB command is faster than using separate SAVLIB commands for each library. Saving only the objects that have changed (SAVCHGOBJ command) is much faster than saving the entire library.

User Profile Authority: If the system operator or the user performing the save operation has *SAVSYS special authority specified in their user profile, then security checking of the objects is bypassed. If *SAVSYS special authority is not specified in the user profile, then the system must check each object to make sure the user is authorized to save the object. This adds time to the save process.

Data Compression and Data Compaction Methods Being Used: The AS/400 system offers three types of data compression and data compaction to allow you to save more data to the media:

- Software data compression is performed by the OS/400 licensed program.
- Hardware data compression is performed by the AS/400 adapter (attachment feature).
- Hardware data compaction is performed by the tape unit. It is supported on only the 3480 and 3490 tape units if the tape unit has the compaction feature.

Software data compression requires central processing unit cycles to perform the compression during the save operation, thus the save time may be increased. The exception is when the central processing unit is capable of processing and then sending data to the save device faster than the device can accept it.

Hardware data compression does not require extra central processing unit cycles. Instead, compression occurs within the hardware of the AS/400 tape adapter. In most cases hardware data compression will decrease save time.

Save Device Characteristics: Whether you are in the process of designing, implementing, or reviewing your backup and recovery strategy, it is important that you understand where save performance problems may occur. Typically, faster or multiple save devices increase save performance. However, if during the save operation you observe the tape drive constantly stopping and starting, then installing a faster tape drive may not significantly increase save performance. Instead this may be indicating that the system is unable to supply data to the tape device fast enough.

Using Save Files: Saving to a save file typically offers good performance; however, there are several considerations. First, saving to a save file is a process of reading from disk and then writing to disk. This can cause disk contention, decreasing save file performance. The contention can be reduced by placing the save file in a user ASP. Contention can be further reduced by placing the disk unit that contains the user ASP on a separate input and output processor (IOP). In addition, optimum save file performance is obtained when the save file is not sharing a user ASP with other objects, or activity in that ASP is limited when saving to the save file.

Unattended Saves: By using save files, certain save commands can be run in an unattended environment. The next day, the save files can then be saved to tape at close to maximum speed. Save files can exist in either the system ASP or a user ASP. However, a user ASP can offer better performance.

The SAVSYS and the SAVLIB LIB(*NONSYS) commands do not allow saving to a save file.

Note: Keep in mind that save files may require extra disk space on the system.

Journaling of Database Files: Journaling of database files is perhaps the cheapest and easiest to use of all the recovery assistance methods. It can be started or stopped for one file at a time. Journaling can be done on all critical or large files, or until the overhead becomes too much for the existing configuration.

Journaling offers a good solution for reducing the save time if you have a large file with few changes. By saving the journal receiver daily, and the entire system less frequently, you can significantly reduce the daily save time in most environments. Journaling incurs the following costs: disk space, disk write operations, central processing unit, overhead, IPL time, and journal and receiver management.

Disk Space: The volume of database changes that are recorded by a journal directly affects the amount of space the journal receivers use. If the disk space utilization is already high, additional disk units may be necessary to contain the journaled information. However, frequently saving and deleting receivers minimizes the amount of space required.

Disk Write Operations: Journaling is quite efficient in grouping multiple record images when writing to disk, but it adds to the total number of input/output operations required of the disk configuration. One recommended technique to minimize the impact of the additional write operations is to dedicate an ASP to each journal receiver. This has three advantages:

1. The read/write head dedicated to the receiver can remain positioned over the disk track where the next write operation must take place. Therefore, no time is lost waiting for a read or write operation to the same storage unit.
2. The additional write operations caused by journaling do not have to be absorbed by the disk unit doing the application work.
3. If the system ASP is destroyed, the journal receiver in the user ASP remains intact and can be used to recover the journaled files to a very recent state.

Central Processing Unit Overhead: The amount of processing unit resource required depends on the number of files being journaled and the rate of transactions causing journal entries.

IPL Time: At IPL time all journaled files are synchronized with the latest journal entries for those files. This additional activity is required only if the IPL follows an abnormal termination.

Journal and Receiver Management: Additional steps are needed for the backup and recovery procedures to utilize journaling. It is necessary to include procedures to change, save, and delete the journal receivers as needed.

Commitment Control: Commitment Control provides for automatic data integrity for transactions involving multiple updates, and it provides for user control of transaction completion or rollback. However, there is additional overhead associated with commitment control.

Commitment control requires both before- and after-images to be journaled. If you specified only after-images when you started journaling the files, the system automatically journals the before-images when the files are under commitment control. This implies all of the overhead associated with journaling, in addition to:

- Additional journal write operations for each commit and ROLLBACK operation.
- More system overhead for ROLLBACK operations because it can involve rolling back multiple transactions and writing additional journal entries to the record that was rolled back.
- A commitment control environment established by the STRCMTCTL command involves setting up internal control information.
- Locking all changed records within a commit cycle concurrently; the number of records locked within a job increases the overall system resources used for the job. One job holding multiple record locks can have a negative effect on other jobs that need the same resources.
- Changes to application program logic are required in order to establish the commitment control environment and to establish the commitment cycle boundaries.

Access Path Journaling: Access path journaling is probably the most powerful tool available for the prevention of lengthy access path rebuilding during an IPL. One of its strengths is that it can be activated or deactivated one file at a time.

Access path journaling requires that the underlying physical files be already journaled, so that the costs are the same as for journaling, with the additional cost of disk space used by the journal receiver. Before-images of access path pages and records whose keys are changed are written to the receiver.

Very little additional I/O or processing unit overhead is required because the index pages and before-images are grouped with the other journal entries as they are written to disk.

User Auxiliary Storage Pools (ASPs): This reserved space provides a media for holding temporary online backup copies using save files and a way of isolating the read and write activity necessary for journaling. When journal receivers are placed in a user ASP, the separation provides protection from failures in the system ASP.

User ASPs can be created out of newly added disk units with very little disruption of the existing configuration.

If user ASPs are to be created out of an existing disk configuration (for example, no new disk is being added), a complete save and restore of the source ASP must be done in order to move objects off the disk units that will make up the user ASP.

Recovery procedures must ensure that journals, journal receivers, and save files are loaded or created again in the correct sequence, to the correct ASP.

Checksum Protection: Checksum protection provides a means of reconstructing information on a disk unit if the unit fails and is replaced.

The system divides all of the disk space in the ASP into checksum sets, each of which contains from two to eight storage units. Space equivalent to the size of one storage unit in each set is used to store checksum data. In a checksum set of two units, half of the disk space is used for checksum data and half for user data. In a checksum set of eight units, one eighth of the disk space contains checksum (or redundancy) data and seven eighths contains user data.

Any time a piece of data on any disk unit in a checksum set is updated, the corresponding sector in the checksum data for that location must also be updated. This requires an extra read of the original page that is about to change, a read of the checksum page, then a write of the checksum page. So a typical update operation grows from one read and one write operation to three read and two write operations. This additional workload on the disk configuration may require additional disk units, controllers, or adaptors to handle the increase.

The machine pool size should be increased by approximately 5% to hold the extra disk pages in memory while the checksum calculation is performed.

Performing the actual checksum algorithm can increase the central processing unit utilization by approximately 10%, depending on the number of updates.

Part of each disk unit in a checksum set is used to store the redundancy (or checksum) data. Therefore, it is necessary to clear all data in the system in order to start checksum protection. For the system ASP, the entire system must be restored. For a user ASP, all data in the user ASP must be saved with the Save Library (SAVLIB) command, checksum protection must be started, and all data in the ASP must be restored.

During an abnormal end of the system it is possible that the checksum calculation and update process could be interrupted, so checksum information may not be synchronized. Therefore, the IPL process following an abnormal system end validates the checksum data. If a copy of main storage was not saved to unit 1 during the abnormal system end, this validation may lengthen the IPL time considerably.

Mirrored Protection: Mirrored protection provides a way of writing data to two disk units at the same time. The information on one disk unit is duplicated on the other. The information can be read from either disk unit. You start mirrored protection by specifying an auxiliary storage pool (ASP) number. You can mirror one or more ASPs on the system. However, all ASPs should have mirrored protection to provide for maximum protection.

The main cost of using mirrored protection is in additional hardware. To achieve high availability and prevent data loss when a disk unit fails, you need mirrored protection for all the ASPs. This normally requires twice as many disk units.

If you want continuous operation and prevention of data loss when a disk unit, controller, or I/O processor fails, you need duplicate disk controllers and I/O processors. A model upgrade can be done to get nearly continuous operation and to prevent data loss when any of these failures or a failure of bus 1 occurs. If bus 0 fails, the system cannot continue to operate. Because bus failures are rare, and bus-level protection is not significantly greater than I/O processor-level protection, you may not find a model upgrade to be cost-effective for your protection needs.

Note: Bus-level protection is not possible for unit 1 on a 9406 system unit because both units in the mirrored pair must be on Bus 0.

Mirrored protection has a minimal effect on performance. If the busses, I/O processors, and controllers are no more heavily loaded on a system with mirrored protection than they are on an equivalent system without mirrored protection, then the performance of the two systems should be approximately the same.

In deciding whether or not to use mirrored protection on your system, you must evaluate the cost of potential downtime against the cost of additional hardware over the life of the system. The additional cost in performance or system complexity is usually negligible.

Uninterruptible Power Supply: If power outages occur often, it is wise to consider installing an uninterruptible power supply in order to increase the chances that a copy of main storage is saved. If a copy of main storage is saved, rebuilding access paths and checksum recovery time can be reduced.

Installing an uninterruptible power supply involves two major investments:

1. The initial cost of the uninterruptible power supply and the space to install it. An option for the B10, B20, Dxx, and Cxx models is an internal battery backup that provides power for a minimum of 5 minutes.
2. The initial cost to automate an orderly shutdown without operator intervention. This requires some thought and some CL programming effort to accomplish.

When using an uninterruptible power supply, there are a variety of ways to control what happens when the utility power fails. The options vary depending on:

- How long the uninterruptible power supply can provide power.
- Whether the intent is to remain fully operational throughout the power outage.
- If operations can be suspended until utility power is restored.

The system values QDEVRCYACN and QDSCJOBITV can be used to prevent ending interactive jobs when power to the system is maintained with an uninterruptible power supply. However, the workstation power is lost. This will also handle the situation where only the line or connection has ended unexpectedly.

Battery Feature on 9402, 9404, and 9406 Model D System Units: An internal battery is supplied as an optional feature with the 9402 and the 9404 system units and as a standard feature for the 9406 Model D system unit. This feature allows the system to perform a normal power down sequence if a power failure should occur.

The system unit has sufficient power to keep the processing unit and the disk unit powered for a minimum of five minutes.

Dual Systems: Using two parallel systems will achieve the highest availability. The costs of implementing a dual system approach may include the following:

- Cost of the duplicate or secondary system. In many situations the second system may not need to be a complete duplicate of the primary system because all applications may not need to be available while running on the backup system. It may not require as much disk space or as large of a central processing unit. The cost of the second system may be partially offset if it can be used to run additional applications, as well as for emergency use.
- There must be a communications link between the two systems across which journal entries can be sent if nearly immediate transaction duplication is required. An alternative is to log the journal entries to tape as they occur. Periodically, and upon failure of the first system, the tape can be taken to the backup system and the changes applied to the duplicate files.
- Programs will need to be developed to save the journal entries on the primary system and either transmit them to the secondary system or write them to a tape. On the secondary system, programs will need to receive the transmitted or transferred journal images and update the duplicate files accordingly. There are vendor-provided packages available that address this type of program.
- An automatic or manual procedure must be used to switch the online devices from the primary to the secondary system. If all or most of the workstations are PCs or PS/2*s using PC Support on a local-area network (LAN), they can have a session attached to each system.
- Uninterruptible power supply on one or both systems further raises the total availability.
- Journaling is implied to save the transactions for transmittal to the secondary system.

NSD Business Recovery Services, Fully Operational Recovery Centers: IBM NSD Business Recovery Services offers a very flexible solution for customers who require a high degree of availability within their computer network. NSD Business Recovery Services also offers on-site and remote testing capabilities, a single point of contact, immediate response to your declared disaster, as well as an IBM recovery team on site. For more information, contact your local IBM marketing representative.

Job Recovery Considerations

Job recovery and starting again should be a basic part of application design. Applications should be designed to handle unexpected data problems (alphabetic data where numeric data is expected), operator problems (operators taking the wrong option or canceling the job), or equipment problems (work station, disk unit, and communication line failures).

Job recovery procedures should ensure the integrity of the user's data and allow for easy starting of the interrupted application. Recovery procedures should be transparent to the end users.

Interactive Job Recovery: If you are running a data entry job or one that updates a single file, it is unlikely that you need to plan an extensive recovery strategy. The operators can inquire against the file to determine which record was last updated and then continue from that point.

To recover from inquire-only jobs, the work station operators simply start where they left off. When using update transactions for many files, consider using a journal or commitment control. The system automatically recovers journaled files during the initial program load (IPL) following an abnormal end of the system. In addition, the journal can be used for user-controlled forward or backward file recovery. Commitment control, using the file changes recorded in the journal, provides automatic transaction and file synchronization. The system will automatically roll back file updates to the beginning of the transaction. In addition, the commitment control notify object can assist you in restarting the transaction.

When designing an interactive application, consider the possibility that you can experience equipment problems with your work stations and communications lines. For example, suppose your computer system loses power. If you have an uninterruptible power supply installed to maintain power to the processing unit and disk units, your system remains active. However, in this example, your work stations lost power. When your programs attempt to read or write to the work stations, an error indication is returned to the program. If the application is not designed to handle these errors, the system can spend all its time in work station error recovery.

You should design your interactive applications to look at error feedback areas and handle any errors indicated. If the application handles the errors and stops, the system resource is not used to do nonproductive error recovery. Examples of using error feedback areas and error recovery routines can be found in the programming languages reference manuals.

Batch Job Recovery: Print-only batch jobs normally do not need special recovery to start again. Running the program again may be adequate.

Batch jobs that perform file updates (add, change, or delete actions) present additional considerations for starting again and recovery. One approach to starting again is to use an update code within the record. As a record is updated, the code for that record can also be updated to show that processing for that record is complete. If the job is started again, the batch program positions itself (as a result of the update code) to the first record that it had not processed. The program then continues processing from that point in the file.

Another way to start batch processing again is to save or copy the file using either the Save Object (SAVOBJ) or the Copy File (CPYF) command before starting the job. Then, if you have to start again, restore or copy the file to its original condition and run the job. A variation of this approach is to use the journal. For example, if starting again is required, you could issue the Remove Journal Change (RMVJRNCHG) command to remove changes to the files. Then, run the job again against the files. With this approach you need to ensure that no other job is changing the files. One way to ensure this is to lock the file for exclusive update while the job is running.

If your batch job consists of a complex input stream, you probably want to design a strategy for starting again into the input stream. Then, if the batch job needs to be started again, the job determines from what point the stream continues.

Commitment control also can be used for batch job recovery. However, if you plan to use commitment control for batch jobs, consider that the maximum number of record locks allowed in a commit cycle is 32 708. Therefore, you need to divide the batch job into logical transactions. For example, if your batch program updates a master file record followed by several detail records in another file, each of those sets of updates can represent a logical transaction and can be committed separately. Because locks are held on all records changed within a commit cycle, dividing your batch job into small logical transactions makes changed data available to other jobs more quickly.

Objects That Make Up the AS/400 System

Figure 2-2 breaks the system into ten basic parts. The following section shows how different parts of the system are saved or restored.

1	Model-Unique Licensed Internal Code
2	Licensed Internal Code
3	Configuration Objects (I/O)
4	User Profiles and Security Objects
5	Distribution Objects
6	OS/400 Licensed Program (QSYS)
7	Licensed Program Libraries
8	QGPL and QUSRSYS
9	User Data
10	Folders and Documents

RV2W373-2

Figure 2-2. Objects That Make Up the AS/400 System

1. Model-Unique Licensed Internal Code

   The model-unique licensed internal code is considered a piece of the processing unit hardware and is not the same thing as licensed internal code.

   Model-unique licensed internal code is unique to each model type of the AS/400 system. The tapes are not interchangeable among model types; they each have their own part numbers. Model-unique licensed internal code is not shipped on the same tape as the OS/400 licensed program and IBM licensed programs. Instead, it is shipped on a separate tape and is located in the system service kit. This is true, whether:

   - The system was ordered as a total system package system or not.
   - The system was bought directly from IBM or a business partner.
   - The OS/400 licensed program was obtained directly from IBM or obtained by a DSLO agreement.

   Every system except the 9402 model C4 receives a model-unique licensed internal code tape, just as every system receives a central processing unit card.

   Model-unique licensed internal code is not saved by any save command. If model-unique licensed internal code ever becomes damaged or must be restored, the system will display an SRC (system reference code) on the operator panel, indicating that you need to place the model-unique licensed internal code tape in the tape unit. Additionally, if you distribute software, the model-unique licensed internal code can never be distributed. If the model-unique licensed internal code tape is ever lost or damaged, your local IBM branch office will help you obtain a new one. Since the model-unique licensed internal code may be required in a recovery situation, you should store this tape in a safe and accessible place. Figure 2-3 shows an example of the label for the model-unique licensed internal code.

```
                    P/N 1234567

(C) COPYRIGHT IBM CORP.   19XX
ALL RIGHTS RESERVED
MODEL UNIQUE LIC-Property of IBM

LICENSED FOR INDICATED SPECIFIC SERIAL NUMBER SYSTEM.
NOT LICENSED FOR ANY OTHER USE, REPRODUCTION, ADAPTATION
OR DISTRIBUTION.

TYPE 940X     S/N XX-XXXXX          MODEL BXX
```

Figure 2-3. AS/400 Model-Unique Licensed Internal Code Tape or Cartridge Sample Label

2. Licensed Internal Code

   The licensed internal code is shipped on IBM tape and is the file labeled QFILEMCD, located ahead of the OS/400 licensed program. Unlike the model-unique licensed internal code, licensed internal code is located on the IBM-supplied OS/400 tapes and is the same for all models of the AS/400 system.

3. Device Configuration Objects

   These are line, controller, and device descriptions and other device configuration objects required for communications.

4. User Profiles and Security Objects

   These are any IBM-supplied or user-created user profiles. Security objects consist of authority holders (*AUTHLR) and authorization lists (*AUTL) object types.

5. Distribution Objects

   These are used internally for object and office distribution purposes for the incoming document distributions, outgoing document distributions, and error distributions.

6. OS/400 Licensed Program (library QSYS)

   The IBM-supplied library QSYS contains the majority of OS/400 objects, such as files, programs, and internal objects.

7. Licensed Program Libraries

   These are installed licensed programs, such as RPG, OfficeVision, and PC Support. The majority of the licensed programs are stored in one or more of the program's libraries. RPG is stored in library QRPG; PC Support is stored in library QIWS. However, during the software installation process, objects related to that licensed program may be placed in libraries QSYS, QUSRSYS, and QGPL.

   In addition to licensed programs, there are software functions referred to as optional parts of a licensed program. These optional parts can be deleted if they are not needed. For example, PC Support consists of several parts, which include: the base PC DOS programs, DBCS-(DOS), OS/2 programs, and DBCS-(OS/2). If your PC Support environment will not be using OS/2 or DBCS (Double-Byte Character Set - Japanese), then the four optional parts that reference OS/2 and DBCS can be deleted.

8. QGPL and QUSRSYS Libraries

   The two IBM-supplied libraries can contain both IBM-supplied objects as well as user-created objects. QUSRSYS contains IBM-supplied journals and journal receivers, output queues, message queues, translation tables, and office objects, such as files, for office enrollments and user calendars.

9. User Data

   User data is any object that you have created and placed in either an IBM-supplied library or a library you created.

10. Folders and Documents

    Documents and folders consist of object types *DOC (document) and *FLR (folder). Documents and folders can only be stored in the IBM-supplied library QDOC.

Commands Used to Save Objects on the AS/400 System

Now that the object types that make up an AS/400 system have been identified, we can look at the save commands available for use in a backup strategy (see Figure 2-4 on page 2-13).

The intent of this information is to provide you with an overview of what the command saves and where it fits in the overall strategies.

Save Storage Command: The Save Storage (SAVSTG) command produces a tape that is a sector-by-sector copy of all permanent data stored on the disk units, including the licensed internal code. The save storage operation makes an image copy of all permanent data that resides on the AS/400 system's configured disk units.

The SAVSTG command saves all the licensed internal code, device configuration objects, distribution objects, user profile and security objects, QSYS, licensed program libraries, QGPL, QUSRSYS, and user data. It also includes all office data and all permanent and temporarily applied program temporary fixes (PTFs). The SAVSTG command also saves spool file data.

Save System Command: The Save system (SAVSYS) command is part of a complete system save operation. It saves the licensed internal code, OS/400 licensed program, configuration objects, distribution objects, user profiles and security objects, and all permanent and temporarily applied program temporary fixes (PTFs).

The SAVSYS command does not save IBM licensed programs (such as RPG, OfficeVision*, or PC Support), optional parts of the OS/400 licensed program, documents or folders, libraries QGPL and QUSRSYS, or user libraries.

Save Security Data Command: The Save Security Data (SAVSECDTA) command is a subset of the SAVSYS command. The SAVSECDTA command saves distribution objects (if MAIL(*YES) is specified), user profiles, and security objects. SAVSECDTA offers an advantage in that it can eliminate the need to perform frequent save system operations when changes are made to the security data. Since the SAVSYS command saves the same security data as SAVSECDTA, it is not necessary to run SAVSECDTA after a SAVSYS. A SAVSYS operation may only be needed for a release upgrade, after applying a cumulative PTF package, or when configuration changes have been made.

Save Library and Save Object Commands: The SAVLIB(*NONSYS), SAVLIB, and SAVOBJ commands or any combination of these commands, are used to save IBM licensed program libraries (and all permanent and temporarily applied PTFs for that program product), libraries QGPL and QUSRSYS, and any other user-created libraries. SAVLIB LIB(*NONSYS) saves all libraries on the system in alphabetic order. SAVLIB LIB(*NONSYS) requires that the system be placed in a restricted state (all subsystems ended). SAVLIB saves from 1 to 50 libraries in the order specified on the command.

Libraries QSYS, QDOC, QSPL, QTEMP, QSRV and QRECOVERY cannot be saved by the

SAVLIB command. Do not use the SAVCHGOBJ command to save IBM licensed program libraries. For backup and recovery purposes, SAVLIB LIB(*NONSYS) or SAVLIB LIB(*IBM) is the intended method for saving IBM licensed programs.

Save Library (SAVLIB) Command: Using this command, you can save:

- All user libraries to diskette or tape using SAVLIB LIB(*ALLUSR). User libraries are defined as libraries that the user creates, or IBM-supplied libraries that contain user data. IBM-supplied libraries that do not contain user data are not saved. The *ALLUSR value reduces the amount of data saved because the IBM libraries are not included.
- All IBM-supplied libraries to tape or diskette using SAVLIB LIB(*IBM). Only IBM-supplied libraries that contain IBM objects are saved. The *IBM value on the LIB parameter allows the user to save IBM supplied-libraries less frequently, such as after installing a new cumulative PTF package.

  Notes:

  1. You can use the OMITLIB parameter to exclude up to 50 libraries when *ALLUSR, *IBM, or *NONSYS is specified on the SAVLIB command. The default for the OMITLIB parameter is *NONE.
  2. The system must be in a restricted state to ensure a complete save of all libraries.
  3. Objects that are in use in a library are not saved. The library should be saved at a later time when the objects are not being used.
  4. Users cannot update objects in libraries that are being saved.

Save Changed Object Command: The Save Changed Object (SAVCHGOBJ) command allows you to save all changed objects (except for office objects) from one or more user libraries. If the value for LIB parameter is specified as *ALLUSR, changed objects from all user libraries are saved. User libraries are defined as libraries created by the user or any IBM-supplied library that contains user data. The *ALLUSR value allows you to save all of the changed objects (with the exception of office objects) in a single operation.

Note: You can use the OMITLIB parameter to exclude up to 50 libraries when the value *ALLUSR is specified on the SAVCHGOBJ command. The default for the OMITLIB parameter is *NONE.

Save Document Library Object Command: The Save Document Library Object (SAVDLO) command saves specified documents or folders located in library QDOC and unfiled documents (mail) in library QUSRSYS. The SAVDLO command does not save office user data, such as calendars and document details. These are saved when library QUSRSYS is saved.

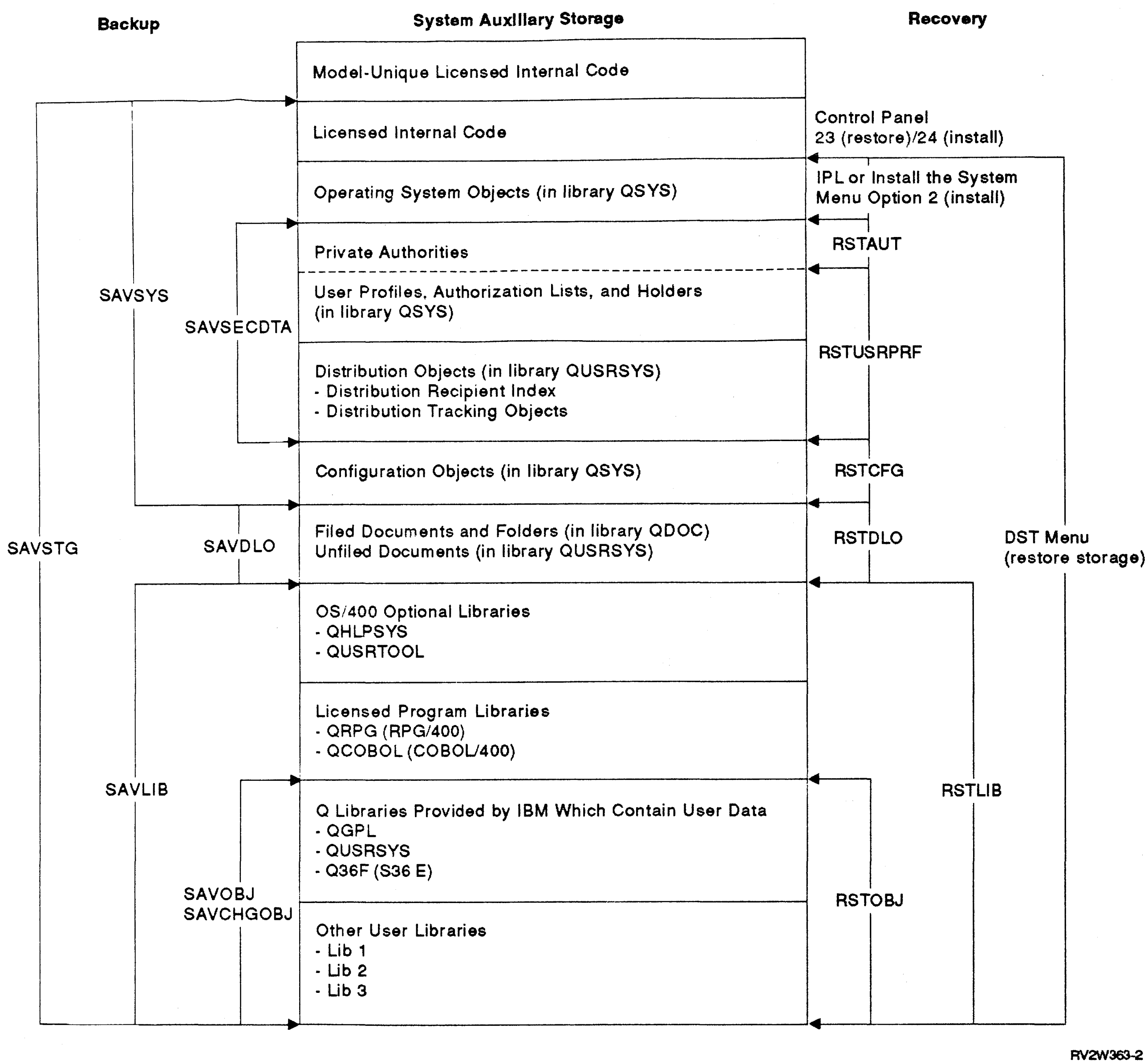


Figure 2-4. Save and Restore Commands That Can Be Used

Failure Types and Associated Data Loss

The discussion of the recovery design process will structure a walk-through of recovery events and considerations for each of the following failure types:

- Site failure (flood, tornado, fire, and so on)
- System failure due to:
  1. Power failure with data loss
  2. Disk failure with data loss
  3. Non-disk failure (such as processing unit and OS/400 program damage)
- Object failure (program bug causes logical damage to files)

Site Failure: The failure or loss of the site can be caused by fire, flood, explosions, or sabotage. Although the possibility of such disasters is remote, recovery from these events must be planned. If you are not prepared for this type of disaster and it occurs, you may not be able to recover the information. As part of your plan, it is a good idea to keep backup tapes and critical supplies at a separate location.

System Failure: The failure of the system itself can be caused by:

- *Power failure with data loss.* This causes the system to end abnormally. Normally, an initial program load (IPL) of the system can correct these errors.
- *Disk failure with data loss.* If the disk unit cannot be recovered, the source data must be restored. If a disk unit is lost, you may have to restore the operating system, licensed programs, application programs, and user data.
- *Non-disk failures.* Most failures will not cause the system to end abnormally. If a severe hardware failure causes the system to end abnormally, the hardware must normally be repaired before the system can perform an IPL.

Object Failure: The most common type of failure is the loss of an object or group of objects, such as files, libraries, or programs.

An object can be lost or damaged due to several factors, including equipment errors, programming errors, or operator errors. Any of these occurrences can cause program processing to end abnormally. For example, if an operator selects the wrong tape, data that is not current can be loaded and cause out-of-date databases. Files will need to be brought up to date.

Questions You Should Ask About Recovery

For each type of failure, there are basic questions that need to be answered during the recovery planning process:

1. What is likely to be lost during this type of failure?
2. What is the basic sequence and extent of the recovery process?
3. How long is the recovery likely to take?
4. Are there methods that can be used to avoid the failure or reduce the recovery time?

The cost of implementing the recovery methods,the cost of the recovery time, and other dependencies are determined by your business requirements.

Site Failure

The shaded portion of Figure 2-5 on page 2-15 represents what can be lost during this type of failure.

1. What is likely to be lost during this type of failure?
   - Everything except what is stored off-site or in a vault
2. What is the basic sequence and extent of the recovery process?
   - Restore the system from the last off-site backup copy.
   - If any recent on-site backups are valid, use them to update as possible.
   - Other transactions since the last good backup must be manually entered again.
3. How long is the recovery likely to take?
   - Several days
4. Are there methods that can be used to avoid the failure or reduce the recovery time?
   - The more frequently an off-site or protected backup copy is made, the less data needs to be entered again.
   - If a dual system approach is used and the secondary system is at a location not affected by the disaster, hardware replacement can take some time but very few transactions are lost. The backup system can become the primary one in a matter of minutes or seconds.
   - It may be advisable to have an arrangement with another AS/400 system user nearby, or with NSD's Business Recovery Services to provide a backup system that could be used until the necessary hardware is replaced.

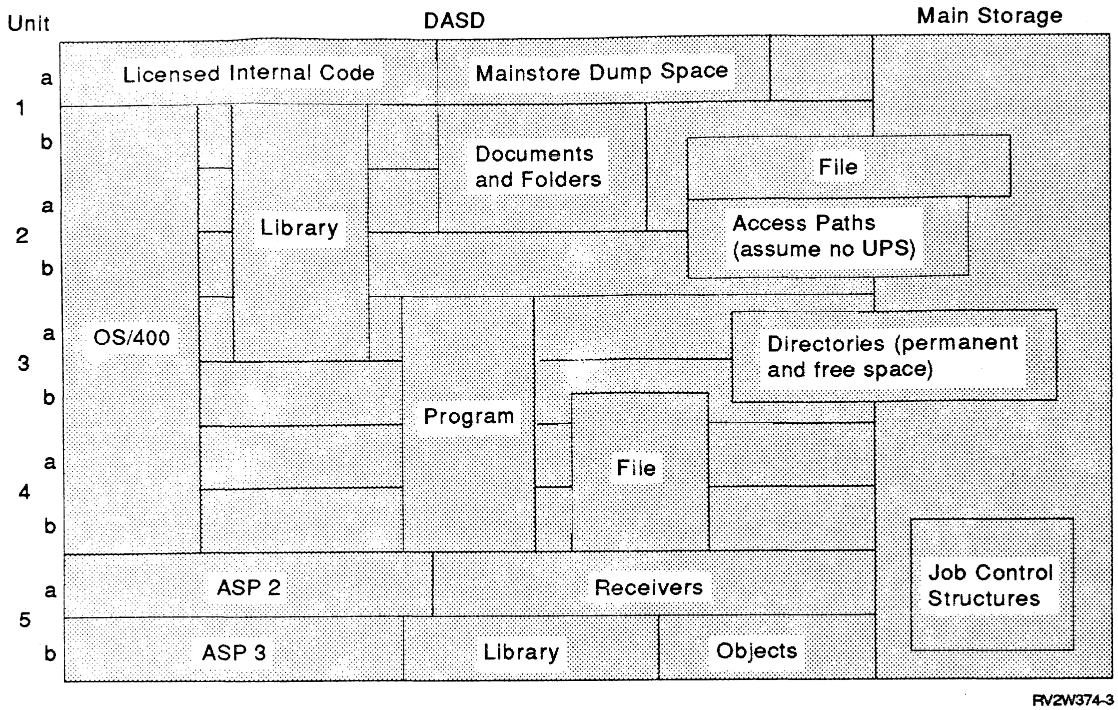


Figure 2-5. What Can Be Lost with a Site Failure

Power Failure

The shaded portion of Figure 2-6 represents what can be lost during this type of failure.

1. What is likely to be lost or damaged during this type of failure?
   - High probability of loss:
     - Access paths of files opened for update
     - Transactions in process
     - Permanent and free space directories
     - Checksum redundancy data (if used)
   - Possible damage to objects being used when the failure occurred, such as message queues and files.
2. What is the basic sequence and extent of the recovery process?
   - When power is restored, an IPL closes in-use files and initiates rebuilding of access paths as required.
   - Online users must start applications again and enter in-process transactions again.
3. How long is the recovery likely to take?
   - Depends on the size of the system and on the number and size of the access paths that must be rebuilt.
4. Are there methods that can be used to avoid the failure or reduce the recovery time?
   - Uninterruptible power supply (UPS).
   - Journaling files and access paths.
   - Commitment control.
   - If commitment control is being used, it may be used to help restart application programs at application transaction boundaries.
   - Journaling provides application restart with some user programming.

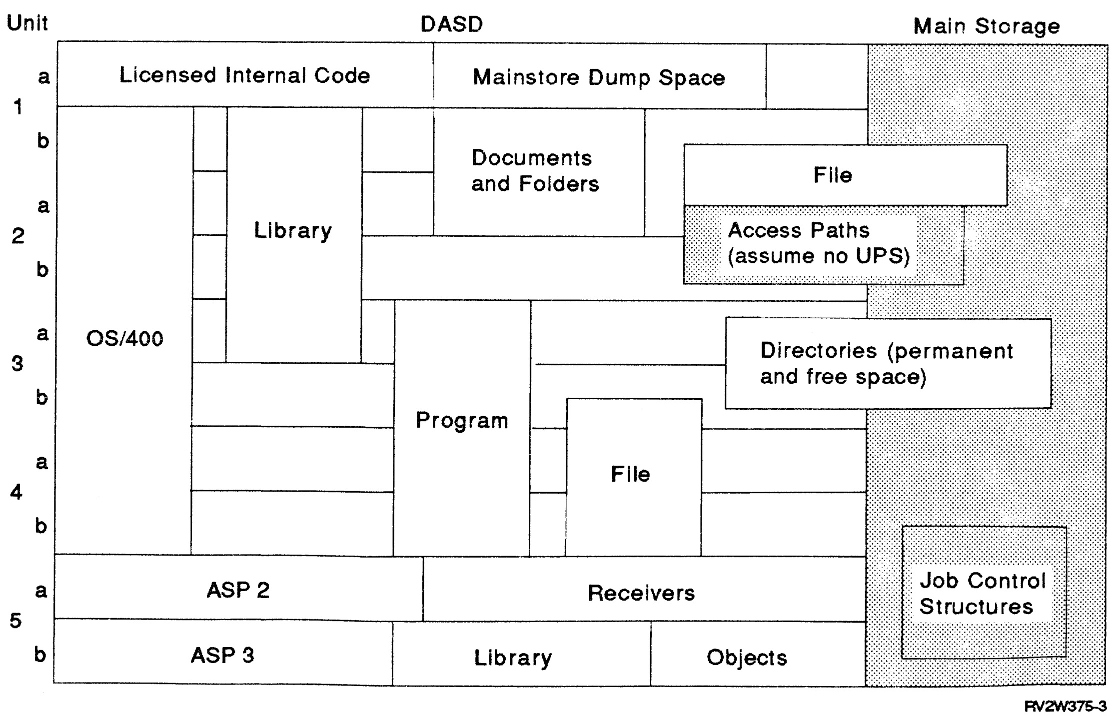


Figure 2-6. What Can Be Lost Due to a Power Failure

Disk Unit Failure (Data Loss)

The shaded portion of Figure 2-7 on page 2-18 represents what can be lost during this type of failure.

1. What is likely to be lost during this type of failure?

   One or more of the following:

   - High Probability:
     - Licensed internal code (only if unit 1 fails)
     - OS/400 licensed program
     - Security information
     - Device configuration data
     - User data (files, access paths, programs)
     - Current transactions
     - Documents
     - Journals and receivers

2. What is the basic sequence and extent of the recovery process?

   a. Replacing the damaged disk.

   b. Restoring the licensed internal code.

   c. Restoring the OS/400 licensed program.

   d. Restoring user profiles, user libraries, configuration objects, documents and folders, and authority.

   e. Capturing and entering all transactions again since the last save operation.

3. How long is the recovery likely to take?

   - Hours, possibly days on a large system

4. Are there methods that can be used to avoid the failure or reduce the recovery time?

   - Checksum protection. The entire system does not need to be reloaded. This reduces the number of transactions that must be recaptured.

     It is recommended that an uninterruptible power supply be installed when checksum protection is used so that checksum validation can be avoided following abnormal system ends caused by power failures.

   - Journal access paths. Checksum protection does not guarantee integrity of access paths. Rebuilding many large access paths can take longer than the checksum rebuild. It is recommended that access path journaling to user ASPs be considered with checksum protection to minimize recovery time.

   - Journaling to a user ASP. If the failing disk unit is in the system ASP and a user ASP contains journals, journal receivers, or save files (with their library in the system ASP), then the data in the system ASP must be restored and the Reclaim Storage (RCLSTG) command must be run.

   - Mirrored protection. The system continues to run until the disk related hardware can be repaired or replaced.

   - User ASPs. Only the data in the failed ASPs needs to be restored. Data in other ASPs does not need to be restored.

   - Dual systems. Transaction processing can continue on the backup system during the recovery. This does not eliminate the need for the recovery process on the primary system.

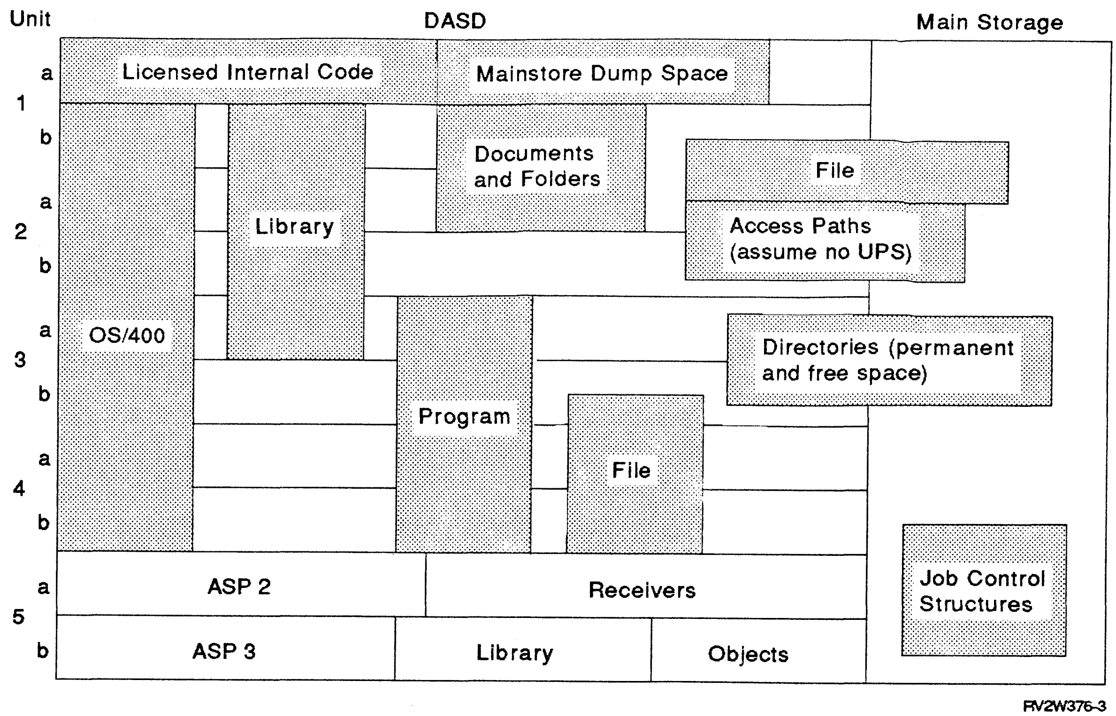


Figure 2-7. What Can Be Lost Due to a Disk Unit Failure

Non-Disk Failure

This category is somewhat miscellaneous, in that it is intended to include a variety of failures which are quite rare, such as a damaged processing unit or memory card, power supply unit, damage to the operating system, or operating system error.

The shaded portion of Figure 2-8 represents what can be lost during this type of failure.

1. What is likely to be lost or damaged during this type of failure?
   - OS/400 licensed program
   - Damage to objects in use, such as user files or documents
   - Access paths
   - Current transactions
2. What is the basic sequence and extent of the recovery process?

   Varies with the specific nature of failure, but may include the following:
   - IPL
   - Reclaim storage operation
   - Clear job queues and output queues
   - Restore the OS/400 licensed program
   - Restore the licensed internal code
3. How long is recovery likely to take?
   - A few minutes or several hours
4. Are there methods that can be used to avoid the failure or reduce the recovery time?
   - Journaling files and access paths
   - Commitment control
   - Dual systems (if damage is not concurrent to both systems)

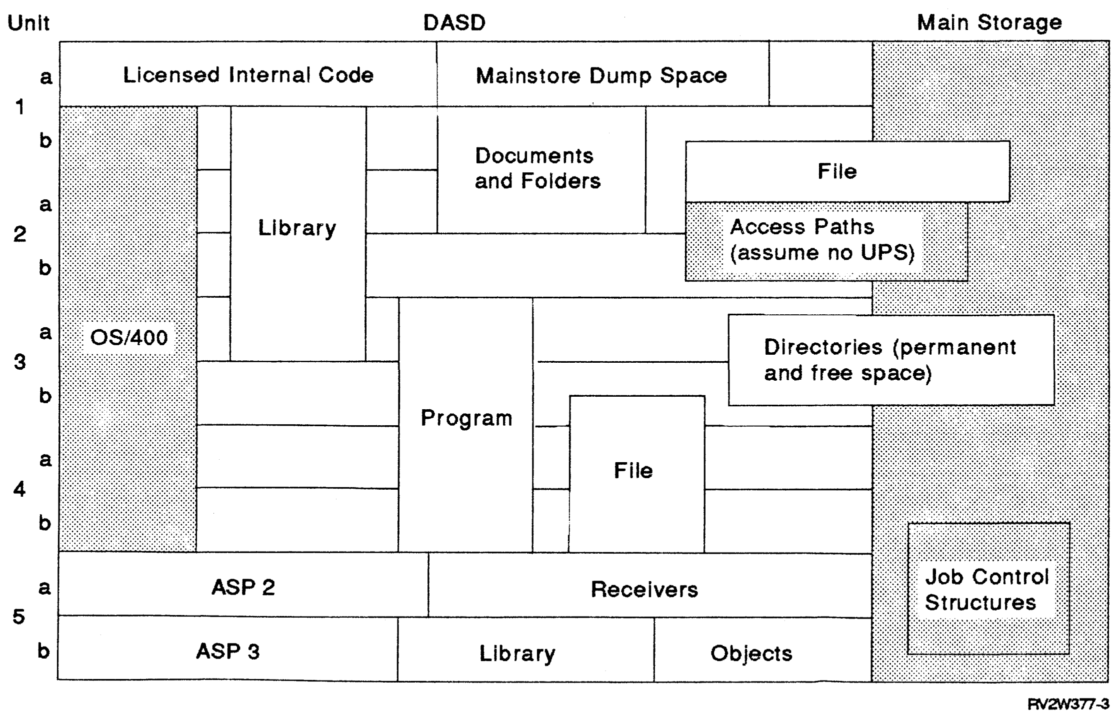


Figure 2-8. What Can Be Lost Due to a Non-Disk Related Failure

Object Failure (Program or Operator Error)

The shaded portion of Figure 2-9 represents what can be lost during this type of failure.

1. What is likely to be lost during this type of failure?
   - Data integrity
   - Access paths
   - Transactions
2. What is the basic sequence and extent of the recovery process?
   - Retract affect of program error.
   - Restore the last good copy of the object or re-create the object and re-run transactions.
3. How long is the recovery likely to take?
   - Depends on the extent of damage.
4. Are there methods that can be used to avoid the failure or reduce the recovery time?
   - Journaling
   - Commitment control
   - Multiple levels of backup
   - Dual systems

     In a dual system approach, the problem can be compounded if the defect has been introduced on both systems and files are not valid on both systems.
   - Procedures should be automated as much as possible.

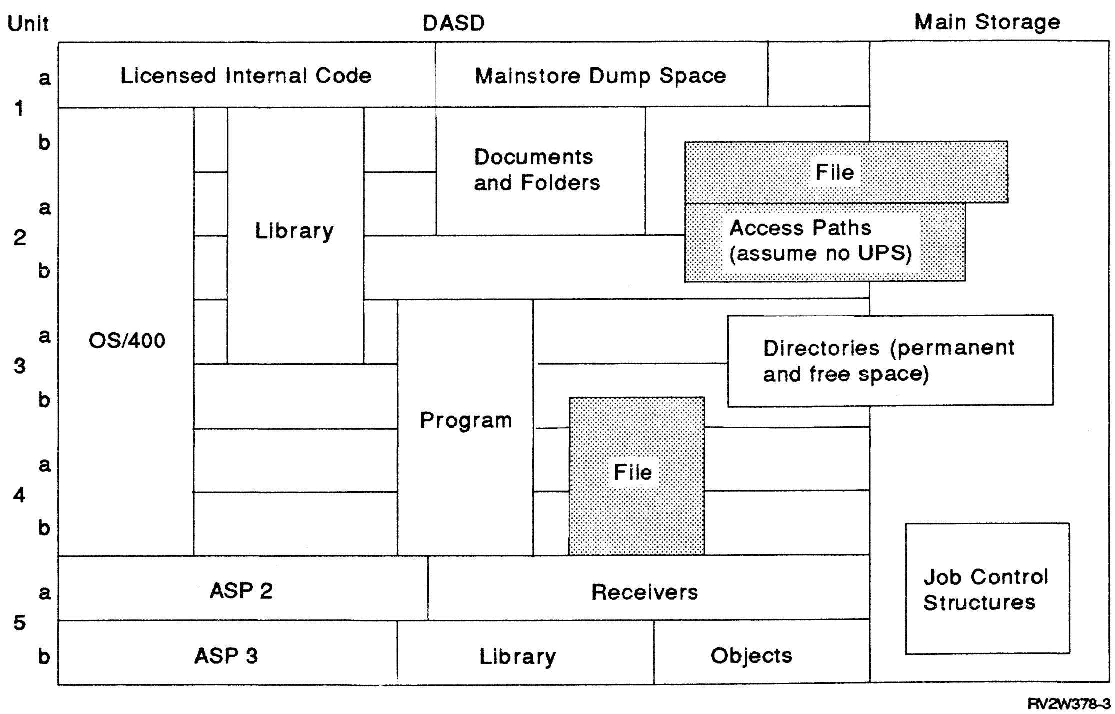


Figure 2-9. What Can Be Lost Due to Object Failure

System Availability and Recovery Strategies

System **availability** is defined as the condition where end users can access and use their applications and data. Obviously, the reliability of your equipment affects the availability of the system. The strength of your applications (how error-free they are, how they handle errors, and so on) also affects the availability of your applications. Another aspect of availability is recoverability.

Recoverability can be defined as the degree or extent to which your system can be restored to an operational condition after a system failure. It also includes ensuring that your database files are not damaged, that they can be accessed, and that they are as current as possible.

The AS/400 recovery tools help reduce the time your system is not available. Although these functions can significantly improve availability, they are not an alternative to a backup or application recovery strategy. These tools should be used in addition to backing up your system, not in place of it.

To simplify the selection of these tools, several examples are described. One may fit your specific requirements, or you may discover a different combination that fits better. Before you select any of these tools, read this manual completely.

Lowest Availability

This combination of recovery tools provides basic backup and job recoverability. The tools include:

- Using save and restore operations
- Using journals to recover physical files
- Using commitment control for transaction recovery
- Using an uninterruptible power supply (optional) to prevent loss of power

Using this combination, an intermediate to large AS/400 system can take less than a day to recover from a failure with no disk unit failure. The time to recover from a disk unit failure may exceed one day.

Because journaling and commitment control are used, the system automatically synchronizes (brings up to date) the database files to a commitment boundary. However, if you experience a disk loss you may have to retype all the data and transactions entered since the last backup operation.

If you experience frequent power outages or fluctuations, it is recommended that you install an uninterruptible power supply.

With this set of tools, consider the additional cost if performance is slower when journaling files.

Medium Availability

This combination of recovery tools provides higher system recoverability in the event of a system failure with no disk device loss. This combination also provides a good level of recoverability, in many instances, when the system fails with disk device loss. The tools include:

- Using save and restore operations
- Using journals to recover physical files
- Using commitment control for transaction recovery
- Using journals to recover access paths
- Using user ASPs to limit data loss and to speed recovery
- Using an uninterruptible power supply (optional) to prevent loss of power

Because journaling and commitment control are used, the system automatically synchronizes the database files to the last commitment boundary. Access paths are recovered rather than rebuilt. Even if you experience a disk unit failure, your database files can be brought up to date because your receivers are isolated in user ASPs. At most, only the last transaction would have to be retyped.

If you experience frequent power outages or fluctuations, it is recommended that you install an uninterruptible power supply.

Some costs to consider with this set of tools include additional disk devices for user ASPs and possibly slower performance when journaling files and access paths.

Highest Single-System Availability

This combination of recovery tools provides the highest availability for a single AS/400 system. The tools include:

- Using save and restore operations
- Using journals to recover physical files
- Using commitment control for transaction recovery
- Using journals to recover access paths
- Using user ASPs to limit data loss and to speed recovery
- Using an uninterruptible power supply (optional) to prevent loss of power
- Using one of the following:
  - Checksum protection to protect the ASPs
  - Mirrored protection to protect all ASPs

Because journaling and commitment control are used, the system automatically synchronizes the database files to the last transaction boundary. Even if you experience a disk device loss, the contents of the lost disk device is reconstructed by checksum protection. If you are using mirrored protection and experience a disk failure, no data is lost and the system continues to run.

Some costs to consider with this set of tools include the additional resources that may be necessary to run checksum or mirrored protection (disk, main storage, and processing unit speed), and the cost for an uninterruptible power supply.

Highest Dual-System Availability

This combination of recovery tools provides the highest availability for the AS/400 system and a second system. This combination assumes you have a second system backing up your primary system. The tools include:

- Using two systems
- Using save and restore operations
- Using journals to recover physical files
- Using commitment control for transaction recovery
- Using journals to recover access paths (optional)
- Using user ASPs to limit data loss and to speed recovery
- Using an uninterruptible power supply to prevent loss of power
- Using one of the following (optional):
  - Checksum protection to protect the ASPs
  - Mirrored protection to protect all ASPs

The database on the backup system may not reflect current information on the primary system. For example, if you are updating the backup system for every transaction, then the backup system data could be one transaction behind the primary system.

Some costs to consider with this set of tools include the cost of the backup system (this cost can be offset, to some extent, by using the backup system for other applications), the connection between the primary and backup systems, and the cost of an uninterruptible power supply. Additional resources may be needed for checksum or mirrored protection.

Strategies Summary

Table 2-1 summarizes these recovery tools. An X indicates the tool is used; an O indicates it is optional.

Table 2-1. Summary of Strategies

Recovery Tool	Lowest	Medium	Highest Single System	Highest Dual System
Save and restore	X	X	X	X
Uninterruptible power supply	O	O	O	X
Journal files	X	X	X	X
Commitment control	X	X	X	X
Journal access path		X	X	O
User ASPs		X	X	X
Checksum protection			X	O
Mirrored protection			X	O
Two systems				X

Table 2-2 illustrates estimated recovery time and whether or not the data is current for the specified recovery tools.

Table 2-2. Estimated Recovery Time and Current Data for Recovery Tools

Recovery Time/Current Data	Lowest	Medium	Highest Single System	Highest Dual System
Estimated recovery time for:				
No disk loss	Long	Medium	Medium	Short
Disk loss	Long	Long	Medium	Short
Is data current for:				
No disk loss	Yes	Yes	Yes	Yes
Disk loss	No	Yes	Yes	Yes

Note: With mirrored protection, there is no data loss. The system continues to run.

Backup and Recovery Summary

This topic provides recommendations for backup and recovery planning and summary charts showing the benefits and limitations for using the functions discussed in this guide.

Table 2-3 on page 2-24 illustrates the benefits and limitations of each of the backup and recovery functions. Carefully examine each of these functions to determine the best protection for you.

Note: There are generally two types of disk unit failures. The most common type involves no data loss. For example, a cable may be damaged. The second type of disk unit failure involves media failure where data loss occurs. For the latter type of disk unit failure, checksum protection, mirrored protection and user auxiliary storage pools provide you with improved protection from the failure.

Table 2-3. Benefits and Limitations of Optional Disk Recovery Functions

Function	Benefits	Limitations
Journaling Files	• Reduces the frequency and amount of data saved when used in conjunction with user ASPs.	• Increase in storage requirements. • May impact performance due to increased I/O and processing unit activity.
Access Path Journaling	• Avoids rebuilding access paths after most abnormal system ends. • Successful even if main storage cannot be copied to storage unit 1 of the system ASP during an abnormal system end. • Generally faster and more dependable than using the FRCACCPTH parameter on the CRTPF, CRTLF, CRTSRCPF, CHGPF, CHGLF, and CHGSRCPF commands.	• Normally requires a significant increase in the storage requirements for journaling. The additional processing time is normally minor.
Auxiliary Storage Pools	• Reduces amount of data loss if a disk unit failure occurs. • Libraries can be placed in user ASPs. This allows for separation of critical or highly used objects. • Performance improvement. If you do extensive journaling, you can place a journal receiver in a user ASP that can be used exclusively for journaling. • Can significantly reduce the number of transactions lost since you last saved the system. ASPs allow you to separate your files and journals receivers: your files can be placed in the system ASP or a user ASP and your journal receivers in another user ASP, reducing the chances that both will be lost.	• System cannot directly recover lost data from a disk unit media failure: requires operations by user to recover. • You must place libraries or objects in a user ASP with a parameter on the create or restore command. • Can require additional disk devices. • System must clear system ASP during the initial configuration unless you install new disk devices.
Checksum Protection	• Lost data is automatically reconstructed after a disk device media failure in the system ASP. • Reduces the number of objects that are damaged.	• Requires additional processing unit resources. • Requires additional main storage. • Can require additional disk devices to prevent slower performance. • Must clear the system ASP to configure.
Mirrored Protection	• System continues to operate with disk unit failures. • On the 9406 system unit, most failed disk units can be replaced while the system continues to run.	• Requires additional disk units.

Chapter 3. System Save Strategies

A save strategy is the plan you put in place to save your data. The methods you choose to use in your plan can vary depending on your business requirements. This topic discusses several approaches to save strategies and the methods used for each approach.

Complete System Save Strategy: A complete system save strategy is one that ends up saving all possible data on the system over a set period of time. A complete system save strategy may mean saving the entire system daily, or it may mean breaking the save up into parts, where some objects are only saved weekly, monthly, or more. There are several approaches you can use:

- Basic Strategy
  - Method 1. Save Storage
  - Method 2. SAVSYS, SAVLIB LIB(*NONSYS) and SAVDLO
- Complex Approach
  - After install--complete save
  - Quarterly--SAVSYS, SAVLIB LIB(*IBM), and SAVDLO
  - Weekly--SAVLIB LIB(*ALLUSR), SAVSECDTA, and SAVDLO
  - Daily--save changed objects

Incomplete System Saves: An incomplete system save strategy is one that does not save the licensed internal code, IBM software, configuration data, or distribution objects. It may or may not save user profiles, security objects, QUSRSYS and QGPL, office data, folders and documents, and user data.

Basic Save Strategies

The following topic discuss two basic save strategies. The two methods save all the data on the system.

Method 1. Save Storage

A basic save method on the AS/400 system would be a daily save storage (SAVSTG command) operation as shown in Figure 3-1 on page 3-2. By design, saving storage is an efficient way of saving data from disk to tape. For systems with smaller amounts of disk space, saving storage is the fastest way of backing up the entire system. In all cases, saving storage is the simplest method.

The save storage operation has a significant disadvantage because it does not allow for restoring of individual objects or libraries, such as the recovery of a data file due to programmer or system operator error. Because most data files are changed frequently, you should have copies of the files on one of the daily save operation media to allow for the restoring of individual objects.

- Advantages
  - Simple, one command saves all
  - Usually fast for small system
- Disadvantages
  - Cannot restore individual objects
  - Requires IPL after complete save
  - Requires dedicated system

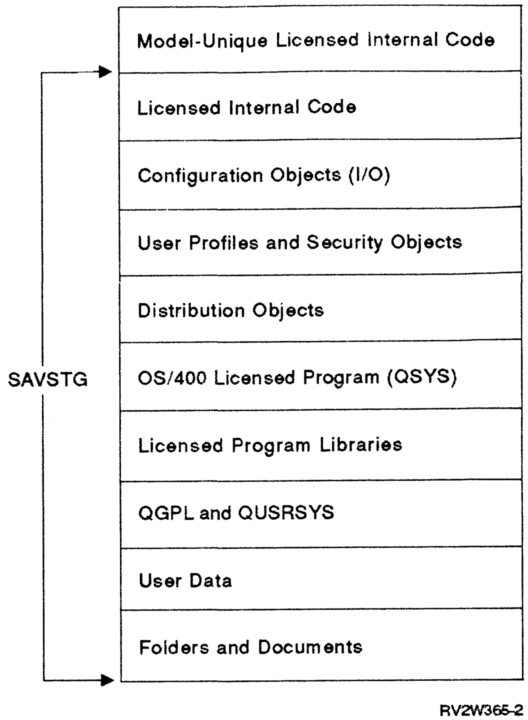


Figure 3-1. Save Storage

The basic steps used for disaster recovery using the SAVSTG media are:

1. Restore the licensed internal code from the SAVSTG media.
2. Restore the model-unique licensed internal code from the tape provided in the service kit (if prompted).
3. Restore the operating system, other licensed programs, and user libraries using the SAVSTG media.

There are some special recovery instructions if you are using journals for a user ASP with save storage recovery.

If you must recover a file, program, or document that has not changed for several months, you will probably need to use the media from the last complete save operation. Therefore, an infrequent (quarterly) complete save operation should be done using method 2 commands to allow for recovery of objects that have not changed. If you have objects that change frequently, you should use method 2 monthly or weekly, or consider saving frequently changed objects daily.

Recovery of the licensed internal code must also be considered. In some situations where damage occurs, it may be necessary to restore the licensed internal code. This can be done by using the save storage media.

The recovery of optional IBM licensed programs, such as the RPG/400* program, must also be performed in certain damage situations. These licensed programs are in libraries that have names beginning with Q, such as QRPG. A typical solution is to back up these libraries as part of the periodic save operation.

It is recommended that the latest program temporary fix (PTF) package be applied before performing the complete save operation using the other save commands.

To provide for the recovery of any object, run an accumulative daily save operation (SAVCHGOBJ) just before running the SAVSTG command. If you need to recover a specific object from the previous cycle, use the last daily save operation media.

Assume that each of the daily save operations (D1, D2, ...) is accumulative; any changes made since the last save storage operation are saved. This provides a simpler recovery technique because only the last daily save operation to tape needs to be restored in a disaster recovery situation.

	Week 1	Week 2	Week 3
Weekly SAVSTG Operation	D1 D2 . . D5	D1 D2 . . D5	D1 D2 . . D5

RV2W387-0

Figure 3-2. Weekly Save Storage with Daily Save Changed Object

To provide for adequate recovery, you should run the daily save operation just before you run the SAVSTG command. The day 5 (D5) media from each week should be kept until the next quarterly save operation occurs. This will allow you to recover any object that was changed during week 1 of the quarter and has not changed since then.

Doing daily save operations just before running the Save Storage command allows for maximum object recovery even if the complete save operation media cannot be used.

The daily save media, used along with the save storage media, allows recovery in case the current save storage media cannot be used. If you need to recover from a disaster, you would normally use the save storage media from that week and then use the daily save operation media. If the save storage media from that week cannot be used, you would:

1. Restore the save storage media from the previous week.
2. Restore the last daily save media from the previous week.
3. Restore the daily save media from the current week.

Notice that when the restore operation of the save storage media occurs, the system is brought back to the point of the save storage operation of week 1. This includes spooled files, messages, and so on, which cannot be saved with the other save commands.

If journaling is used, there are some additional considerations. It is important to note that even if you are not journaling objects, the system may be journaling for you. This is true if you are using Office functions, PC Support, or SNA distribution services. Normally you would want to change the journal (CHGJRN command) just before running the daily save operation. Changing the journal detaches the journal receiver and allows you to save the journal receiver. (Journal receivers can be saved when they are active.) After the journal receiver is saved to the media, it can be deleted from the system. Another approach is to delete the saved receivers once a week.

In the save storage approach for daily save operations, the default for the Save Changed Objects (SAVCHGOBJ) command is that the files being journaled are not saved. The files are not saved until the complete save operation is done (quarterly) using other save commands. The quarterly save operation is too infrequent, and the recovery of many journal receivers is very slow. Therefore, you want to ensure that the files being journaled are saved more frequently.

Method 2. SAVSYS, SAVLIB LIB(*NONSYS) and SAVDLO

Systems with large amounts of data to be saved may not find the basic save method 1 adequate for their recovery requirements (see Figure 3-3 on page 3-4). This method saves all the data on the system, except the model-unique licensed internal code. The commands used for this method include SAVSYS, SAVLIB LIB(*NONSYS) and SAVDLO. This method creates a recent base starting point of use for recovery.

- Advantages
  - Can restore an individual object
  - An IPL is not required
- Disadvantages
  - Requires a dedicated system
  - Slower than the save storage method

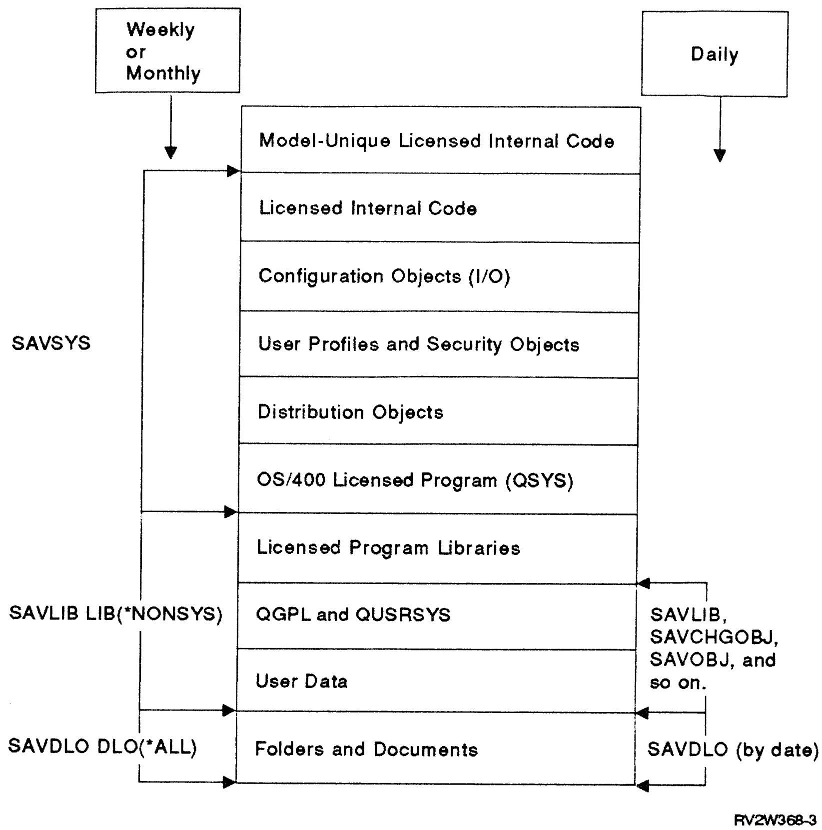


Figure 3-3. Basic Save Method 2

The basic recovery steps are as follows:

1. Restore the licensed internal code from the SAVSYS media.
2. Restore the model-unique licensed internal code from the tape provided in the service kit.
3. Restore the OS/400 licensed program from the SAVSYS media.
4. Restore the user profiles from the current SAVSYS media.
5. Restore the configuration objects (the RSTCFG command) from the SAVSYS media.
6. Restore all user libraries (RSTLIB SAVLIB(*NONSYS)) from the last SAVLIB LIB(*NONSYS) media.
7. Restore user authority (RSTAUT). No specific media is used.

Complex, Complete Save Method

A more complex complete system save method (see Figure 3-4 on page 3-5) is very similar to the previous basic methods, except it greatly lengthens the time between a complete save of the system (SAVSTG or SAVSYS, SAVLIB LIB(*NONSYS) and SAVDLO(*ALL)) and the more frequent daily and weekly saves.

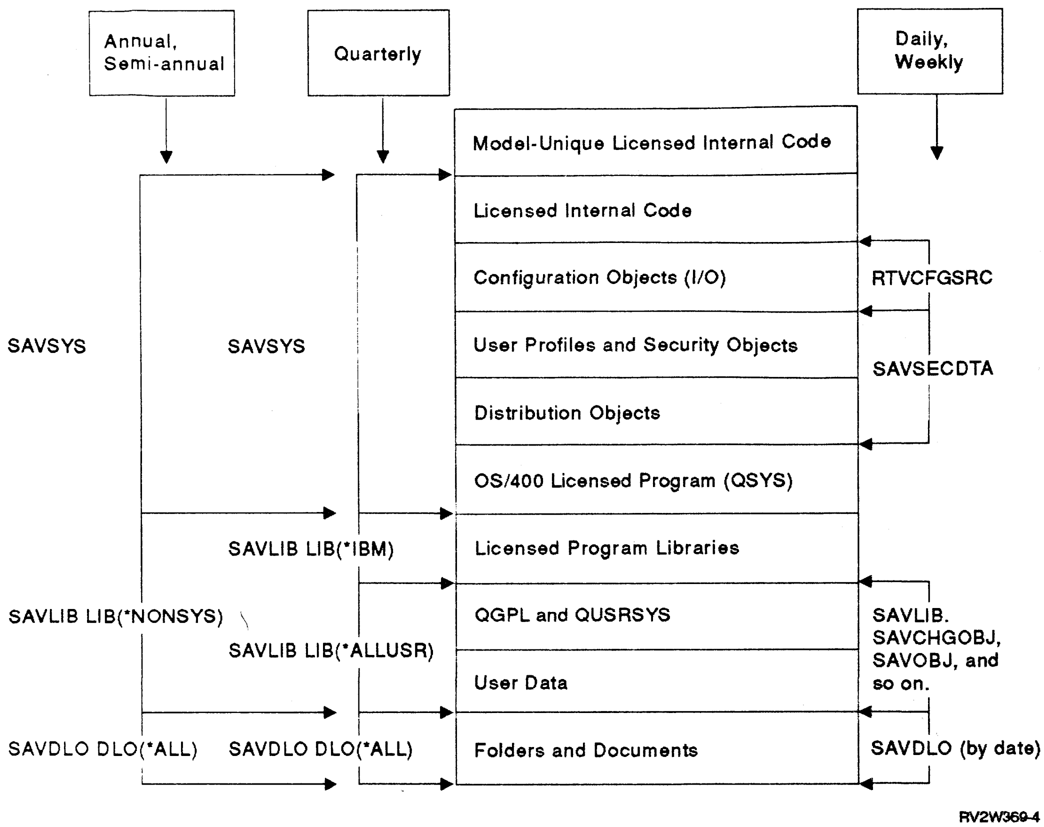


Figure 3-4. Complex Save Method

Once the starting point has been established, periodic save operations of changed and new data are done by using the Save Library (SAVLIB), Save Object (SAVOBJ), Save Change Object (SAVCHGOBJ), Save Document Library Objects (SAVDLO), and Save Security Data (SAVSECDTA) commands. In the event the system must be recovered by restoring the entire system, the base save medium can be restored onto the system, then brought up to date by restoring the more recent save medium of changed and new objects.

Notes:

1. The system must be in a restricted state to ensure a complete save of all libraries.
2. Objects that are in use in a library are not saved. This library should be saved at a later time when the objects are not being used.
3. Users cannot update objects in libraries that are being saved.

Only one of the basic save methods needs to be performed after a new AS/400 system is installed or upgraded to a new release. When IBM licensed program libraries are updated, the data can be saved using the SAVSYS and SAVLIB LIB(*IBM) commands. This is necessary after a program temporary fix (PTF) package is installed and is usually done on a quarterly basis. All user data should be saved on a weekly basis using SAVLIB LIB(*ALLUSR), SAVSECDTA, and SAVDLO DLO(*ALL). Daily changes can be saved using the SAVCHGOBJ, SAVDLO, and SAVSECDTA commands.

- Advantages
  - Shorter backup time
  - Less data is saved, allowing for unattended backup
  - A restricted system is not required for daily and weekly save operations
- Disadvantages
  - Complex recovery procedures

The basic recovery steps are as follows:

1. Restore the licensed internal code from the SAVSYS media.
2. Restore the model-unique licensed internal code from the tape provided in the service kit.
3. Restore the OS/400 licensed program from the SAVSYS media.
4. Restore the user profiles from the most current SAVSYS or SAVSECDTA media.
5. Restore the configuration objects (the RSTCFG command) from the SAVSYS media.
6. Restore all IBM libraries (RSTLIB SAVLIB LIB(*IBM)) from the last SAVLIB LIB(*IBM) media.
7. Restore all user libraries (RSTLIB SAVLIB LIB(*ALLUSR)) from the last SAVLIB LIB(*ALLUSR) media.
8. Restore all documents and folders (RSTDLO DLO(*ALL)) from the SAVDLO DLO(*ALL) media.
9. Restore the last saved changes from the SAVCHGOBJ and the daily SAVDLO media.
10. Restore user authority (RSTAUT). No specific media is used.

In addition to these steps, there are several conditions that need to be handled manually.

- Programs that adopt the owner's authority can be restored. However, the user performing the restore operation must be the owner of the program, or have all object (*ALLOBJ) special authority (for example, QSECOFR). If not, the program is restored, all private authorities are revoked, and only the owner can run the program. For this reason, it is recommended that the restore operation be performed using the QSECOFR user profile. To avoid signing on as QSECOFR, you can use the Restore Any Library (RSTANYLIB) command found in library QUSRTOOL, which allows you to adopt the QSECOFR user profile during the restore operation.
- When a logical file is saved, you can request that the access path be saved by specifying ACCPTH(*YES) on the save command; the access path is saved with the physical file. When the logical file is restored, the system will restore the access path, not rebuild it. If a logical file is restored that was saved with ACCPTH(*YES) specified, then the access path is not rebuilt unless MAINT(*REBLD) was specified.

  If the physical and dependent logical files exist in different libraries, the following occurs:

  - If the physical file is restored first, then the logical file can be restored. The access paths will be rebuilt.
  - If an attempt is made to restore the logical file first, the request fails because the logical file is dependent on the physical file that has not yet been restored.
- The journal objects must be restored before the objects being journaled. If they are not, journaling is not started for the object.

  If the journal is in the same library as the objects being journaled, then the system restores the journal first. If the journal is in a different library, you must restore it before you restore the objects being journaled.

  The Convert Journal Attributes (CVTJRNA) command in library QUSRTOOL is designed to help you recover from this situation or other instances when journaling must be started again.

Library QSYS (OS/400) and licensed program libraries do not change very often. Generally, library QSYS is updated only when changes or additions have occurred to: user profile and security information, configuration data, or when there has been a release upgrade or program temporary fix (PTF) activity. Licensed program libraries (such as QRPG and QIWS) generally change only when a release upgrade or PTF activity has occurred.

Because you have control over when a release upgrade will occur, and to a greater extent PTF activity, you can control how often a complete system save operation is required. What is important is how well you have planned. For example, a release upgrade is a fairly easy activity to plan for and work around. Installing a PTF cumulative package is a major activity. Because a PTF cumulative tape tends to affect more than the licensed programs, it is best to save the IBM libraries using SAVSYS and SAVLIB LIB(*IBM) after the package has been installed. This offers two benefits:

1. Your backup media will already be at the proper PTF level. This will eliminate the need to install the cumulative PTF package again, in the event a full system recovery is required.
2. If you have to restore an individual object from your backup media, it will be at the proper level and match other objects already on your system.

Minor PTF activity is installing individual PTFs that are available in between the cumulative packages and offers a fix or an enhancement you need. How minor PTF activity affects your save strategy depends on how well you have planned and written your save strategy. By maintaining an accurate log of save activity and PTF activity, you may decide not to save the entire system when a single PTF fix has been applied. Instead, your save medium and PTF log will tell you what PTFs are not contained in the set of save tapes and must be re-applied. Again, the important part is to fully plan for and write down your save and restore strategy.

Limited, Incomplete Save Method

A limited, incomplete system save method is one that periodically saves user profiles and security objects, QUSRSYS and QGPL, and all user data (programs, files, and so forth). It may or may not include a method of saving configuration information (RTVCFGSRC). All user data can be saved using SAVLIB LIB(*ALLUSR), SAVSECDTA, and SAVDLO.

Because the method shown in Figure 3-5 does not save IBM software, program temporary fixes (PTFs) are not saved. In the event the entire system must be restored, IBM-supplied software must be installed again, just as if it were a new system. All PTFs must be applied again. If the Save Security Data (SAVSECDTA) command is not used, user-created user profiles will have to be created and authority granted again as necessary. This save strategy is sufficient as long as recovery time is not an issue.

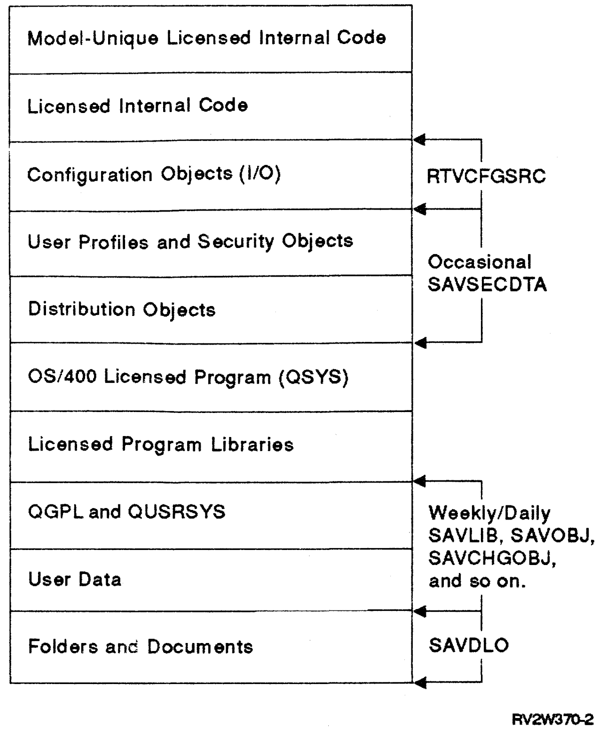


Figure 3-5. Incomplete Save Method

The basic recovery steps are:

1. Install the licensed internal code from the IBM distribution tape.
2. Install OS/400 licensed program, IBM-supplied libraries (QGPL and QUSRSYS), and IBM licensed programs from the distribution tape. Use the instructions "Installing a New Release of AS/400 Without the Operating System Installed" in the *Licensed Programs and New Release Installation Guide*, SC41-9878. This step includes installing the latest cumulative PTF package.
3. Restore the user profiles from the most current SAVSECDTA media, or re-create the user profiles if no save operation was done.
4. Restore all user libraries from the last SAVLIB LIB(*ALLUSR) media.
5. Re-create the configuration objects from the most current retrieve configuration source (RTVCFGSRC command) file, re-create the configuration objects manually, or use automatic configuration.
6. Restore private authorities using the RSTAUT command if a RSTUSRPRF command was run, or manually grant desired authorities using the EDTOBJAUT command.

Summary of Save Strategies

Table 3-1. Summary of Save Strategies

TYPE	COST	BENEFIT
BASIC Method 1	• No partial recovery. • No individual object recovery.	• Fast save and recovery. • Very simple to use
BASIC Method 2	• More complex than BASIC method 1. • More commands.	• Full individual object recovery. • Full flexibility. • Partial recovery.
COMPLEX	• Extra planning required. • Operator training. • More complex recovery. • Multiple commands. • CL Programming.	• Medium to fast save and recovery. • Full individual recovery. • Partial recovery. • High Flexibility.
INCOMPLETE	• Long recovery. • Requires manual recovery and rebuild. • Limited individual object recovery. • PTFs are lost.	• Fast save (user data only).

Backup Strategies for Office Data

The purpose of this topic is to provide backup and recovery strategies for office data. If you are not using OfficeVision/400 or documents and folders, you do not need to read this topic.

Save Strategy for Office Data

A complete save operation of office services data means all objects on the system used by office services are saved to offline storage.

A daily save operation of office services data means only those objects on the system used by office services that have changed or been created since the last complete save operation are saved to offline storage. You should use a **cumulative save changed data** strategy for daily save operations. All office services objects changed since the last complete save operation are saved during each daily save operation. This strategy makes restoring your entire system in the event of a disaster (fire, flood, power outage, and so on) much easier.

You choose the frequency for both save operations. You should do a complete save operation of office services data once a week, in addition to a daily save operation of office services data. You can adjust this schedule to fit your system activity level. Do not allow too much time between complete save operations, because outdated complete save data makes it more difficult to restore your entire system.

Complete Save Operation

The specific types of objects saved during a complete save operation of office services data include office database files, the office services journal (QAOSDIAJRN), journal receivers, distribution objects, distribution documents, and filed documents and folders. Figure 3-6 on page 3-9 illustrates a complete save operation.

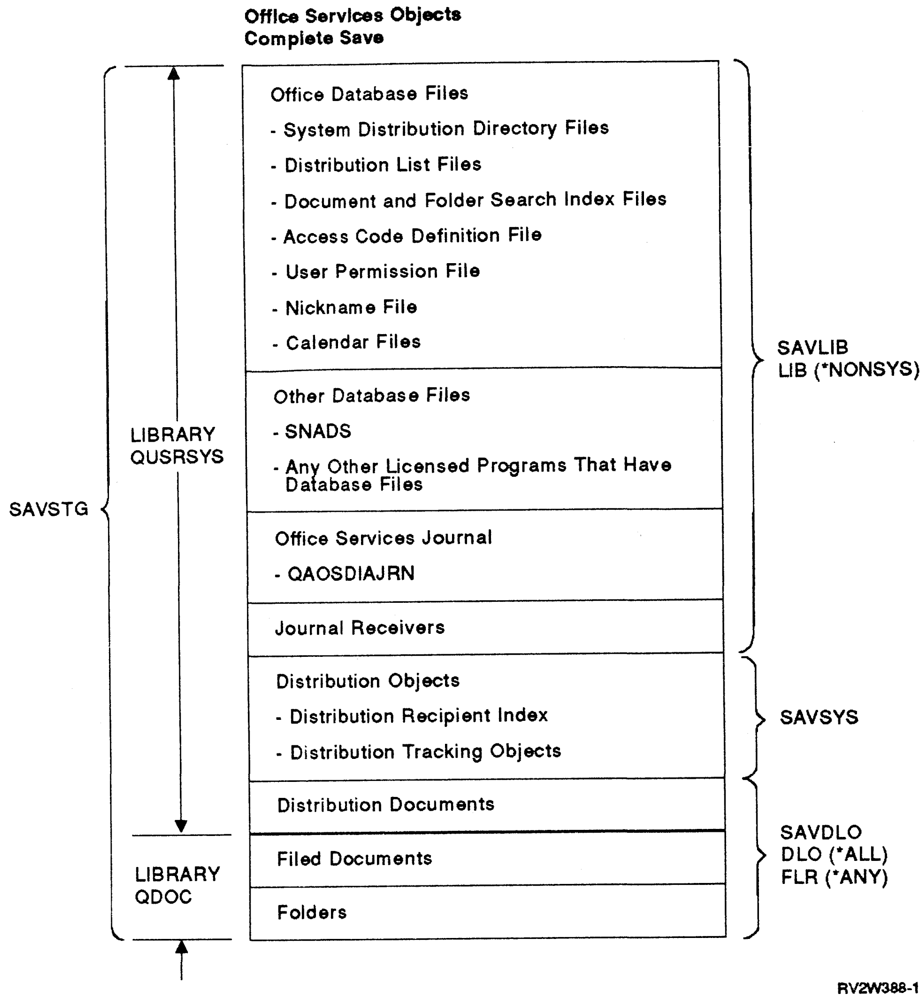


Figure 3-6. Complete Save of Office Services Objects

There are two methods that can be used for a complete save:

Method 1: The Save Storage (SAVSTG) command only. The SAVSTG command saves all licensed internal code and all disk unit data to tape. All office services data is included when saving storage is done. This command is intended for disaster recovery. Individual objects cannot be restored from the SAVSTG tapes.

Method 2: A combination of the Save System (SAVSYS), Save Library (SAVLIB), and Save Document Library Object (SAVDLO) commands. The SAVSYS and SAVLIB commands save data in addition to office services data. The following commands ensure office services data is saved:

1. SAVSYS

   The office services data saved by SAVSYS includes all the distribution objects from library QUSRSYS. These distribution objects are a part of mail. Figure 3-7 on page 3-10 illustrates the structure of saved mail.

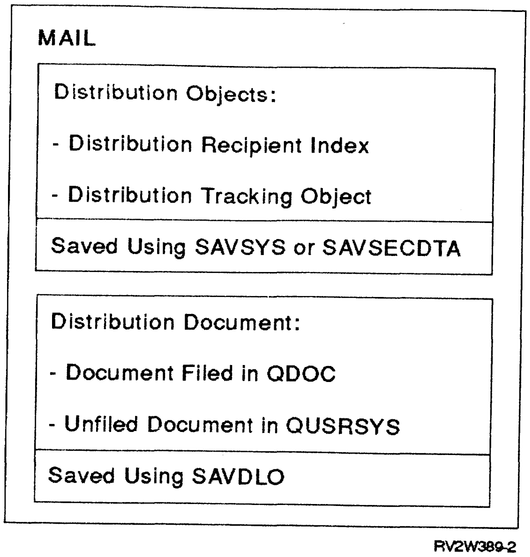


Figure 3-7. Saving Mail

2. SAVLIB LIB(*NONSYS)

   The *NONSYS option results in the save of the entire QUSRSYS library. The office services data saved by saving library QUSRSYS includes:

   - The database files for directory, distribution, document library, and calendar services.
   - The office services journal (QAOSDIAJRN)
   - Journal receivers

3. SAVDLO DLO(*ALL) FLR(*ANY)

   Saves all filed documents in QDOC library, all folders, and all distribution documents in library QUSRSYS. Some of the filed documents (those filed off the mail log) and all of the distribution documents that are saved are a part of mail. See Figure 3-7 for an illustration of the structure of saved mail.

   You do not have to stop all office activity when running this command. However, if you run this command with office activity on, the system may not save some documents and folders. Check your job log after running this command to find the names of any documents or folders not saved because they were being used at the time of the SAVDLO. Save these documents and folders separately. Do not run this command when your system is at peak activity.

Daily Save Operations

Objects saved with a daily save operation of office services data are the same as those saved with a complete save. A daily save operation updates your complete save operation with the most recent save changes for your system. It is important for disaster recovery to restore the latest version of an object.

Note: With the exception of mail, a daily save operation saves only the office services data that has changed or was created since the last complete save operation. It does not save all the office services data currently on the system.

When saving mail, all distribution objects and all documents associated with distributions are saved. You are saving a record of the state of the mail on your system at the time of the save. You can use the save medium to recover all the mail that existed at the time of the save, but you cannot use it to restore an individual user's mail or an individual distribution document.

The SAVCHGOBJ and SAVSECDTA commands save data in addition to office services data. For example, the SAVSECDTA command saves user profiles, authorization lists, and authority holders. For a description of the data (not used by office) saved by SAVCHGOBJ and SAVSECDTA, see the descriptions of these commands in this guide.

Figure 3-8 on page 3-11 shows the objects saved by a daily save.

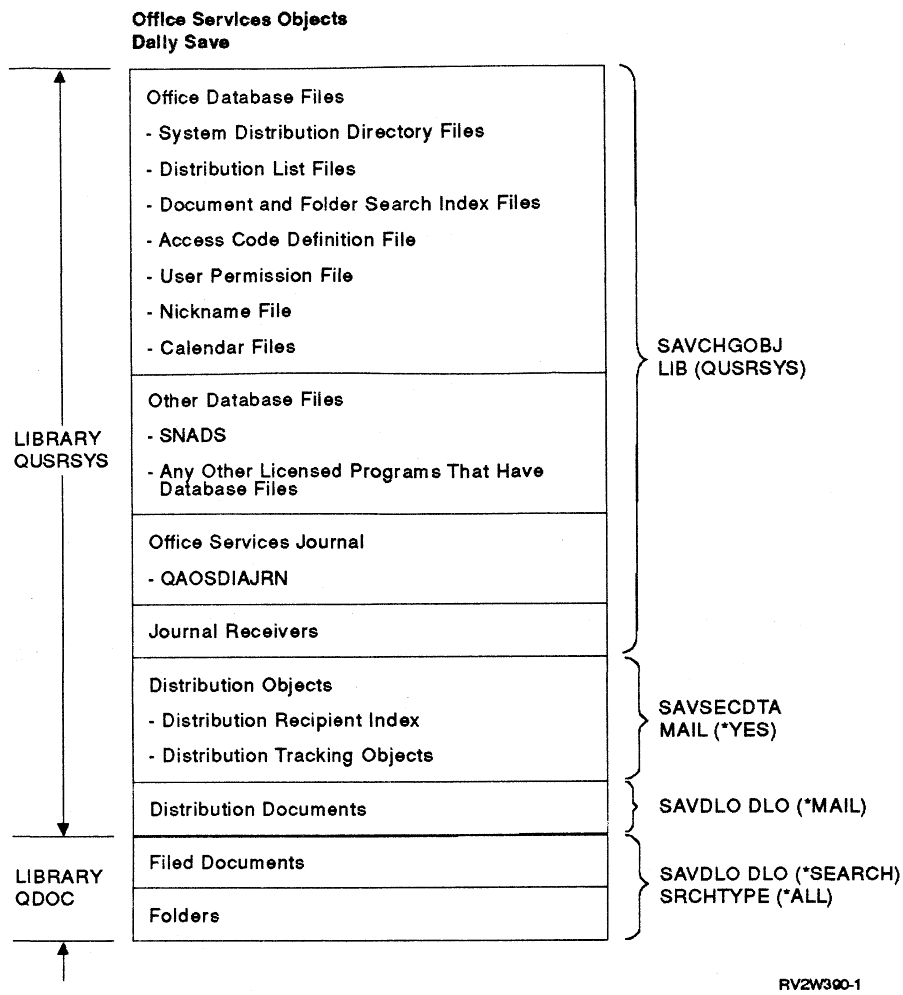


Figure 3-8. Daily Save of Office Services Objects

The set of commands used for daily save operations is as follows:

1. SAVCHGOBJ OBJ(*ALL) LIB(QUSRSYS) OBJJRN(*YES)

   The office services data saved by the SAVCHGOBJ command includes all changes that have occurred since the last complete save of library QUSRSYS.

   Notes:

   a. If you do not journal user files, specify OBJJRN(*YES). This simplifies the recovery procedures and avoids applying journal changes.

   b. The default on the SAVCHGOBJ command is OBJJRN(*NO). Database files that are journaled are not saved with the save changed object request but journal receivers are saved. Keep this in mind because you will need to apply the journal changes in the saved receivers to recover your system or an individual database file.

2. SAVSECDTA MAIL(*YES)

   The SAVSECDTA command with the MAIL option of *YES saves distribution objects from library QUSRSYS currently on the system. This command save all user profiles, authorization list, authority holders, and distribution objects, not just the new or changed objects.

   On most systems, mail changes so frequently that a daily save operation is a must if mail is saved at all. If restoring mail is an important part of your disaster recovery strategy, you should use SAVSECDTA MAIL(*YES) at least once a day.

   You can reduce the amount of mail saved daily by doing the following:

   - Encourage users not to allow their mail logs to become too large. Promptly delete mail items that are no longer needed.

- Encourage users to promptly file distribution documents that they want to keep. After the documents are filed, remove the mail log entry for the filed documents.

You can run the SAVSECDTA command when the system is not in a *restricted state*. However, at most 32 766 distribution objects can be saved by the SAVSECDTA command. Keep the object limitation in mind and manage your mail logs accordingly.

3. SAVDLO DLO(*SEARCH) SRCHTYPE(*ALL) CRTDATE(creation time/creation date) OWNER(*ALL) REFCHGDATE(reference change date)

   Use this command to save any folders created, or any filed documents created or changed, since the latest complete save operation. Depending on the rate at which new folders are created on your system, you may or may not have new folders to save daily.

   Note: Enter the time and date of the last complete save operation in both the CRTDATE and REFCHGDATE keywords. This allows you to restore only the latest complete save medium and the latest daily save medium in case disaster recovery becomes necessary.

4. SAVDLO DLO(*MAIL)

   This command saves all the distribution documents and all the filed documents referred to by a distribution (see Figure 3-8 on page 3-11).

   Note: If you use the SAVSECDTA MAIL(*YES) at least once a day to save distribution objects on your system, you must save the documents referred to by these distribution objects after the SAVSECDTA command completes by using the SAVDLO DLO(*MAIL).

   If saving mail is not important and you do not run SAVSECDTA MAIL(*YES) as part of your daily save operation, you should omit doing a SAVDLO DLO(*MAIL).

Strategies for IBM-Supplied Journals

A typical strategy is to perform the following once a week:

1. Change the journal receiver and attach a new one using the Change Journal (CHGJRN) command.
2. Save library QUSRSYS using the Save Library (SAVLIB) command.
3. Delete the detached journal receiver.

There are four journals shipped with the system. These journals are:

QAOSDIAJRN - Journal for DIA Files:

This journal provides recovery for document library and distribution services files. The journal tracks office changes and activity from OfficeVision/400 and PC Support/400 licensed programs.

QDSNX - Journal for DSNX Log: This journal found in library QUSRSYS is only used for audit trail for DSNX activity. You can access the information to determine what DSNX functions are performed by users and when they were performed. Only entries are sent to this journal. It does not journal database files.

QSNADS - Journal for SNADS Files: This journal found in library QUSRSYS is only used for audit trail for SNADS activity. You can access the information to determine what SNADS functions are performed by users and when they were performed. Only entries are sent to this journal. It does not journal database files.

QSXJRN - Journal for Problem Database: This journal is used for recording the activity in the database files that are in library QUSRSYS and contain service related problems. It is recommended that this information be kept for 30 days. You should change the journal receiver periodically by using the Change Journal (CHGJRN) command.

There are two optional journals that you must take specific steps to create and maintain. These journals are:

QAUDJRN - Journal for Logging Security-Related Events: See the manual *Security Concepts and Planning*, SC41-8083, for more information about the security auditing journal.

QACGJRN - Journal for Job Accounting: The *Programming: Work Management Guide*, SC41-8078, has more information about job accounting and the job accounting journal.

Verifying a Backup Design

Verify what objects are being saved, and when, by doing the following:

- Locate objects not being saved
- Review the save and restore job logs
- Find out when and where libraries are saved

How to Locate Objects That are Not Being Saved

Your best chance of avoiding a recovery exposure is to know exactly what you are saving (or perhaps more importantly, what you are not saving). Although you may save objects weekly, if the objects are changed daily, you may not be in a position to recover the changes.

To prevent such an exposure, you should periodically verify your backup strategy by reviewing when objects and changes made to these objects are being saved. This can be done by using the Check Save (CHKSAV) command in library QUSRTOOL.

The CHKSAV command allows you to determine whether one or more libraries have had any objects or members changed since the last save operation. CHKSAV can be used on any or all libraries except library QSYS. When library QSYS is saved using the SAVSYS command, individual object descriptions are not updated. Instead, a SAVSYS updates the object description of two data areas named QSAVSYS and QSAVUSRPRF in library QSYS.

To determine when a SAVSYS command was run, use the DSPOBJD command to display the object description of these two data areas (SAVSECDTA updates the object description of the data area QSAVUSRPRF only). You do not display the data area itself. You display the object description of the data area using the DSPOBJD or WRKOBJD command.

A second command that assists you in locating objects not being saved is the Print Library Analysis (PRTLIBANL) command in library QUSRTOOL. The PRTLIBANL command analyzes all objects by library (either a single library or all libraries) and summarizes, by days, what objects were changed, saved, or not saved. This summary helps to easily identify any exposure.

Once you determine that all objects and changes are being saved to your satisfaction, consider using the CHKSAV command on a periodic basis, perhaps monthly, semi-annually, or yearly. As new applications are placed on the system, or as existing applications are changed, use the CHKSAV command to make sure these additions or changes have not created an exposure in your backup and recovery strategy.

How to Review the Save and Restore Job Logs

It is also important to make sure that the save command completes successfully. This can be done by reading through the job log created during the save. However, this can be a tedious task for large save operations. To help automate this function, the Check Save Status (CHKSAVSTS) command in library QUSRTOOL can be used.

The CHKSAVSTS command reads the job log that includes the save or restore commands and prints a list of completion messages for that save operation, as well as any diagnostic messages found. The CHKSAVSTS command summarizes the job log used during the save or restore process and summarizes what is important from a save or restore viewpoint. CHKSAVSTS command allows you to direct the output to either a printer or a database file.

How to Find Out When and Where Libraries Are Saved

Using the Display Object Description (DSPOBJD) command against a specific library can tell when and where the library was saved (date, time, and what tape volume). The DSPOBJD command works for single libraries. However, to help verify your backup and recovery strategy, you may want to look at multiple libraries. This can be accomplished by using the Print Save Status (PRTSAVSTS) command in library QUSRTOOL.

The PRTSAVSTS command is intended for use following a save of multiple libraries. The PRTSAVSTS command creates printed output that contains a description of one or more library names, save date, save command, and the volumes the libraries are saved to. This list should then be saved to tape in the event a restore is necessary. An external description exists for the libraries that were saved and what tape volumes contain the libraries.

Verify Integrity of the Save or Restore Tapes

The time spent saving an object is wasted if the tape used for the save operation is unusable. All tapes are subject to wear over a period of time. To help monitor the quality of your tapes, use the *Work with tape or diskette statistics* option accessed using the System Service Tools functions. This shows you the number of temporary and permanent read or write errors that have occurred for a particular tape volume as well as the total amount of data read or written to the tape.

Chapter 4. Save and Restore Media Considerations

Diskettes, magnetic tapes, or save files are written to and read from during save and restore operations. Information is provided here to help you decide which medium to use.

Table 4-1 shows the commands and the media that can be used with the save command.

Table 4-1. Media Used with the Save Commands

Command	Type of Media		
	Tape	Diskette	Save File
SAVSYS	Yes	No	No
SAVSECDTA	Yes	No	Yes
SAVLIB	Yes	Yes	Yes
SAVOBJ	Yes	Yes	Yes
SAVCHGOBJ	Yes	Yes	Yes
SAVDLO	Yes	Yes	Yes
SAVSAVFDTA	Yes	Yes	No
SAVLICPGM	Yes	No	No
SAVSTG	Yes	No	No

Objects are saved on a diskette in the diskette unit or on a magnetic tape in the magnetic tape unit. The diskette or magnetic tape is removed from the device and stored offline. There is no limit to the number of tapes or diskettes you can use during a particular operation, but you must specify the devices you use on the save and restore commands.

You cannot use a save or restore command with an override command to the system tape file, a diskette file, or a save file. For example, an override to tape file QSYSTAP or diskette file QSYSDKT is ignored when followed by a SAVLIB command. An attempt to change the expiration date of a saved tape library with an override to QSYSTAP is ignored, and the tape receives the expiration date specified on the SAVLIB command. Or, if the Override Tape File (OVRTAPF) command is used to override the LABEL value for the tape file, that new value is ignored during the save or restore operation. The LABEL value for the save or restore operation is used instead.

You can specify up to 50 volume names on the save and restore commands. This limits volume ID checking to 50 volumes but does not limit the operation to 50 media volumes. A system file is maintained with a record for each media volume entered. (If more than one medium volume exists with the same name, they are assumed to be the same volume.)

The system file grows as new media volumes are used by the system. You can list the system file using the Print Error Log (PRTERRLOG) command. Specifying VOLSTAT(*DLT) on the PRTERRLOG command resets the system file.

Be certain you have enough initialized diskettes or tapes to complete the save operation. Use the Display Object Description (DSPOBJD) command for information on storage sizes. To find the size of a complete library, use the Display Library (DSPLIB LIB(XXX) OUTPUT(*PRINT)) command.

Naming Conventions

A typical approach to media handling is to use sets of media and rotate them on a regular basis. For example, assume you perform a weekly save operation of your payroll information. The save operation requires two tapes and you want a three-set rotational method (sets A, B, and C). The rotational schedule would be as follows:

```
Week 1        Set A
Week 2        Set B
Week 3        Set C
Week 4        Set A
          .
          .
          .
```

To help identify the tape, you can use a naming convention for the volume IDs using PAY and a letter identifying each set and a number to identify the sequence. For example:

```
Set A   PAYA01, PAYA02
Set B   PAYB01, PAYB02
Set C   PAYC01, PAYC02
```

If you decide to add additional sets or the save operation requires additional tapes, the naming convention will allow these changes.

Color codes can be used to further identify the different sets of save tapes for a specific day. For example, Monday save tapes could be coded with a green plus, Tuesday with blue, Wednesday with yellow, and so on.

Volume IDs: Providing a naming convention is a good practice regardless of how you verify volume IDs. There are two ways to verify volume IDs:

Manual Checking: You can use the default for the VOL parameter on the save or restore commands that tell the system to use the tape that is loaded. It is up to the operator to load the correct tape and in the correct order. Usually, this approach is used for small computer systems.

System Checking: You specify a list of volume IDs on the save or restore commands. The system makes sure that the tapes loaded by the operator are the correct volumes and in the order specified on the command. If an error occurs, a message is sent to the operator requesting the correct tape volume. The operator can either load another tape or override the request. This approach is normally used for computer systems that require tighter control of the save media and are running save procedures from CL programs.

Expiration Dates: Developing a strategy for tape expiration dates is also important. There are two general approaches:

- Make all files permanent when saving to tape or diskette. Use the default value (*PERM) on the save commands for the expiration date (EXPDATE) parameter. A date of 999999 is assigned to the files on tape. When a save command is run, the system looks at the files to determine if any have dates that have not reached the expiration date.

  This approach is normally used for small computer systems. If the default is used to make all files permanent, you will need to do one of the following the next time you save to the same save media:

  - Specify CLEAR(*ALL) on the save commands to clear all files on tape before writing to the tape. This causes the system to ignore the expiration date. This approach relies on the operator to load the correct tape.
  - Have the operator respond to each system message that occurs when a tape is read that contains files that have not reached the expiration date. Although this approach is feasible in many cases, the messages can become so routine for the operator that he may respond to them without thinking about it.

- Specify a date on the expiration date EXPDATE parameter. The save commands default to CLEAR(*NONE). The system verifies that the expiration date has passed for the files that exist on the tape. If the expiration date has not passed, a message is sent to the operator explaining the fact.

  Normally, a date that corresponds to your rotational cycle is used. For example, if you rotate your media every three weeks, then specify a date eighteen days past the current date. The EXPDATE parameter does not provide a way to specify the current date plus eighteen. However, the Add Date (ADDDAT) command in QUSRTOOL can be used to create the appropriate day if you use control language (CL) programs to perform your save operations.

The typical solution for small computer systems is to use the system defaults and rely on the system operator to load the correct tapes.

As your system becomes larger, you may want to consider the use of volume ID and expiration date to provide better protection against the operator loading the wrong tapes or loading the tapes in the wrong order.

It is recommended that you stay away from a situation where the operator must respond on every tape loaded to override either the volume ID or the expiration date. After a period of time, the operator may automatically respond to messages without reading them. It is best to let the system provide a message when a real exception occurs.

Strategies Planning

Considerations for Using Tape

Among the items to consider for using tape are:

- Tape unit compatibility
- Performance
- Storage capacity
- Preparing tapes for use

Regardless of your save and restore strategy, you must decide how you will handle your save tapes. Consider the following when using tapes.

- Tape units must be cleaned on a regular basis. The read and write head collects dust and other material that can cause errors when reading or writing to tape. In addition to your regular cleaning cycle, you should also clean the tape unit if you are going to be using it for an extended period of time or you are using new tapes. New tapes tend to collect more material on the read and write head of the tape unit. For more specific recommendations, refer to the manual for the specific tape unit.
- Tapes must be initialized using the Initialize Tape (INZTAP) command. The command allows you to control writing to tape by clearing files on the tape. You can also specify the density or bits per inch (bpi) before writing to tape by using parameters on the INZTAP command when the tape is initialized.
- When using tape as the save and restore medium, it is important that you use the tape unit on your system that is defined as the IPL device when saving the licensed internal code (SAVSYS or SAVSTG command). The IPL device was defined by your service representative when your system was installed. If you do not use the IPL device when saving the licensed internal code, you may not be able to restore the system with the tapes used to save the system.
- Before you begin any save procedure, it is recommended that you initialize enough tapes to complete the save operation. Initialize at least three tapes more than you think you will need. The extra tapes allow for situations where the save operation requires more media than planned for or for damage that may occur to the tapes.
- The INZTAP command requires that a volume identifier (ID) be specified. The volume ID is 6 characters in length. The volume ID should allow you to easily identify the tape. You should not use the same volume ID for all your tapes. It is best to use unique names.

  In many cases, you may need more than one tape to back up a specific function. Therefore, you may want to follow a naming convention. A recommended naming convention is discussed earlier in this chapter.
- In addition to the volume ID, an external label should be used on the outside of the tape to allow you to easily identify the tape. A typical approach is to use the volume ID.
- Tape devices that use 1/4-inch or 8-mm cartridge media (reference Table 4-2 on page 4-4) can only erase existing data if your save operation starts at the beginning of the tape. Therefore, when saving to 1/4-inch or 8-mm cartridge media, the sequence number (SEQNBR) parameter on the save command must be 1 to overwrite all active files. To append a file after existing data, specify *END or 1 greater than the last file on the cartridge.

Tape Unit Characteristics

Table 4-2 on page 4-4 illustrates the characteristics of the compatible AS/400 tape unit models.

Table 4-2. Tape Unit Model Characteristics

Tape Type/Model	AS/400 System Unit	Medium Type	Density[4]	Capacity (MB)	Maximum Bytes Per Second[1]
2440/A12	9406	1/2-inch reel	1600 bpi	44	120KB
			6250 bpi	156	468KB
2440/A12[2]	9406	1/2-inch reel	1600 bpi	44	235KB
			6250 bpi	156	918KB
3422 A01/B01[2]	9406	1/2-inch reel	1600 bpi	44	235KB
			6250 bpi	156	780KB
3430 A01/B01[2]	9406	1/2-inch reel	1600 bpi	44	80KB
			6250 bpi	156	312KB
3480 A22/B22	9406	1/2-inch cartridge	38000 bpi[3]	200	3.0MB
3490 D31/D32	9406	1/2-inch cartridge	38000 bpi	200	3.0MB
6366/0001	9406	1/4-inch cartridge	10000 bpi	120	90KB
6341/001	9402	1/4-inch cartridge	10000 bpi	120	90KB
6342/001	9402	1/4-inch cartridge	16000 bpi	320[6]	200KB
6346/001	9404	1/4-inch cartridge	10000 bpi	120	90KB
6347	9404	1/4-inch cartridge	16000 bpi	320[6]	200KB
7208/0002		8-mm cartridge	43200 bpi	2332	246KB
9346/0001	9406	1/4-inch cartridge	10000 bpi	120	90KB
9347/0001[5]	9406	1/2-inch reel	1600 bpi	43	160KB
			3200 bpi	82	160KB
9348/0001	9406	1/2-inch reel	1600 bpi	44	200KB
			6250 bpi	156	780KB
9348/0002	9404	1/2-inch reel	1600 bpi	44	200KB
			6250 bpi	156	780KB
9348/002	9402	1/2-inch reel	1600 bpi 6250 bpi	44 156	200KB 780KB

Notes:

1 This is the instantaneous data rate that is the maximum rate that the tape unit can transmit data to and from the AS/400 system.

2 This is the rate using hardware data compression.

3 The abbreviation *bpi* is used to represent bits per inch.

4 The values are for data that is not compressed. For programming or device data compression, the capacity normally increases and is data-dependent.

5 The 3600-foot tape is not supported on the AS/400.

6 Capacity is 530MB on a 1000-foot tape.

The rate at which the system does save and restore operations depends on many factors, including the following:

- The processing speed of the AS/400 model.
- The distribution of data on the disk units.
- The number and speed of the disk units.
- The number and speed of the storage processors.
- The operating characteristics of the tape unit (start-stop as opposed to streaming).
- The type of data (large or small objects)

If you have only one tape unit, you must wait for the tape to rewind before loading another volume. However, if you have specified more than one tape unit, processing continues in the second unit while the first is rewinding, and so on. The save storage function supports the use of only one tape unit.

Tape Storage Capacity

A save operation writes variable-length blocks of up to 24.5KB. On a 2400-foot tape reel at 1600 bits-per-inch (bpi), this allows approximately 44MB of data per reel and approximately 161MB at 6250 bpi. Because a save operation adds descriptive information along with the data, a tape with 500 saved objects holds only about 43MB of data at 1600 bpi and 155MB of data at 6250 bpi. These statistics are without data compression. Tape cartridges also have these reductions in capacity. Refer to "Data Compression and Decompression" on page 4-6 for more information.

Initializing Tapes

A save operation does not require that a tape be prepared using a special format but it does require a standard label tape. Use a tape with a standard label and a volume identifier (VOL parameter). You can prepare a tape by using the Initialize Tape (INZTAP) command or by replying to an inquiry message sent during the save operation. Specifying CLEAR(*ALL) on the save command allows unexpired files to be written over without operator intervention.

When no list of volume identifiers is specified for the file and the loaded volume has an acceptable identifier, but some attribute of the volume is incorrect (such as the code), an inquiry message is sent allowing you to label the tape with the same volume and owner identifier. Use the Initialize Tape (INZTAP) command to prepare a tape or to add or change a volume identifier. If more than one volume is needed for the save and restore operations, files may cross the volume boundaries. The volumes must be loaded in correct sequence for restore functions. You also can clear a tape with the INZTAP command.

Performance Considerations for Using Tapes

There are streaming tape drives and start-stop tape drives. For example, the 3430, 3422, 3480, and 6341 are start-stop tape drives while the 9348, 9347, 9346, and 6346 are streaming tape drives. The 2440 can be both a start-stop and a streaming tape drive.

A **start-stop tape drive** has the ability to stop and restart movement of the tape between reading or writing each block of tape data. A **streaming tape drive** cannot stop and start between blocks of data. If the next request does not reach the device fast enough, the tape drive overruns its position on the tape and must stop and backup before it can run the next command. This repositioning takes time and can cause additional stress to the device and the medium.

Save and restore operations are designed to keep the tape streaming as much as possible. However, on heavily used systems, the ability of the system to maintain streaming is reduced and this increases the time required for the save or restore operation. When possible, you should consider limiting save and restore operations on streaming tape drives to times of lower system activity.

Data Compaction and Data Compression

The AS/400 system offers three types of data compression and data compaction to allow you to save more data to the media:

- Software data compression is performed by the OS/400 licensed program.
- Hardware data compression is performed by the AS/400 adapter (attachment feature).
- Hardware data compaction is performed by the tape unit. It is supported on only the 3480 and 3490 tape units if the tape unit has the compaction feature.

Data Compaction: Data compaction is available on the 3480 and 3490 tape units if the compaction feature is installed. Data compaction provides additional media capacity that reduces the amount of media handling during a save operation and allows for unattended backup.

The operating system also allows duplicating tapes (DUPTAP command) that were written using data compaction. Data compaction is not supported when writing data interchange tapes but allows for reading them.

Most save commands allow you to specify the data compaction (COMPACT) parameter. This parameter allows you to use data compaction on the 3480 and 3490 tape units.

Device data compaction can only be used if all tape units specified on the save command support data compaction. Device data compaction can also be used when TGTRLS(*PRV) is specified on the save command.

Unlike data compression, data compaction and decompaction do not have software support available for devices that do not have the data compaction feature. Tapes saved using data compaction must be restored on tape units that support data compaction.

Using device data compaction alone (the default) will usually result in the highest capacity on tape. However, device data compaction and device data compression can be used together and may result in higher capacity in rare cases (if the COMPACT parameter is *DEV and the DTACPR parameter is specified as *YES.) Data compaction is not supported for the SAVSTG function.

Data Compression and Decompression: The AS/400 system uses programming compression or device data compression and decompression. Most tape I/O processors support device compression and decompression.

Programming compression and decompression require significant processing unit resources and have a considerable effect on other system users. Programming compression and decompression usually increase the save and restore time.

Performance may be improved and the amount of space required on the save media can be reduced using programming or device data compression and decompression. **Data compression** compresses data on the media when you perform the save operations. **Data decompression** reconstructs data when you perform a restore operation. The system ensures that information saved can be reconstructed exactly. No data is lost as a result of compression and decompression. Programming and device data compression and decompression is available only for save and restore operations.

Compression normally reduces the amount of storage media required for a save operation. Objects, such as database files that normally contain strings of consecutive characters (such as blanks or zeros), can be significantly compressed. Some objects, however, can actually expand by a small amount and require more storage media than if not compressed. This expansion can be caused by too few repeated strings.

Data Compaction and Data Compression Summary: Table 4-3 summarizes data compaction and data compression by tape unit.

Table 4-3. Data Compaction and Data Compression Summary

Tape Unit	Hardware Data Compression	Programming Data Compression	Hardware Data Compaction
2440[1]	X	X	
3430	X		
3422	X		
3480[2]	X		X
3490[2]	X		X
6341	X	X	
6342	X	X	
6346[1]	X	X	
6347	X	X	
6366	X	X	
7208	X	X	
9346		X	
9347		X	
9348[1]	X	X	

Notes:

1. Determined by the AS/400 attachment feature card
2. Can use both hardware data compression and hardware data compaction together

Data Compression Method: The compression method used compresses strings of identical characters, and adds control bytes to mark the beginning of each compressed or not compressed string. Compression occurs as follows:

- Each string of consecutive blanks (between 2 and 63 bytes long) is compressed to a single control byte.
- Each string of consecutive characters other than blanks (between 3 and 63 bytes long) is

compressed to 2 bytes: a control byte and a copy of the repeated character.

- Each string of non-repeating characters (between 1 and 63 bytes) is expanded by having a control byte added at the beginning of the non-repeating character string.

All control bytes contain the control information and the length of the adjacent equivalent string not compressed. For example, assume the following data stream is being saved (ƀ represents a blank):

ABCƀƀDEF000GHIƀJKL11MNOƀƀƀ
ƀƀƀƀƀƀƀƀƀƀ
ƀƀƀPQR22222222STUSTUSTUSTU

This compresses into the following string where * represents a control byte:

*ABC**DEF*0*GHIƀJKL11MNO**PQR*2*STUSTUSTUSTU

- A control byte appears before ABC to designate a string not compressed.
- The two blanks between ABC and DEF are compressed to a single control byte.
- A control byte appears before DEF to designate another string not compressed.
- The three consecutive zeros are compressed to a control byte and a single zero.
- The blank between the GHI and the JKL is not compressed because there is only one.
- The 11 string between JKL and MNO is not compressed because there are only two consecutive non-blank characters.
- The string of STUSTUSTUSTU is not compressed even though the string repeats itself because individual characters are not consecutive.

This compression method provides significant compression and helps ensure integrity of the data. For example, this method determines where each control byte should be, and identifies an inconsistency.

This compression method is a form of the Systems Network Architecture (SNA) compression method.

Errors That Occur When Using Tape

When reading from or writing to tape, it is normal for some errors to occur. The system has recovery functions that reposition the tape automatically and try the operation again. If the system is writing to tape and an unrecoverable error occurs, the system may request you load another tape and the media can be used for recovery. Processing continues on the next tape. However, there are some situations when writing to tape where the recovery fails and the save operation is ended (processing does not continue on the next tape). The tape units that do not support automatic recovery are the 6341, 6346, 6366, and 9346.

Tapes will physically wear out after extended use. You can determine if a tape is wearing out by periodically printing the error log by using the Print Error Log (PRTERRLOG) command and specifying TYPE(*VOLSTAT). The printed output provides statistics about the volume data, which is kept by the system. If unique volume IDs are used, the information provides a good indication of which tapes have excessive read or write errors and should be removed from the media library.

Considerations for Using Diskettes

Among the items to consider for using diskettes are:

- Performance
- Storage capacity
- Preparing diskettes for use

Performance Using Diskettes: Table 4-4 illustrates the characteristics of the compatible AS/400 diskette models.

Table 4-4. Diskette Model Characteristics

Diskette Type/Model	Maximum KB per Second	Maximum MB per Hour for Save/ Restore Rate	Maximum MB per Volume
9331/0002	19	68/60	1.2
9331/0001	19	68/60	1.2
6133/0001	19	68/60	1.2
6332/0001	19	68/60	1.2
6331/0001	19	68/60	1.2

Diskette Storage Capacity: The AS/400 system supports two types of diskettes for data exchange. The save and restore operation requires the use of the IBM diskette 2D for 8-inch diskettes with a sector size of 1024 bytes or a double-sided high-density 5.25-inch diskette with a sector size of 1024 bytes. This two-sided, double-density diskette is prepared to save and restore type-E format allowing a total capacity of 1 212 416 bytes per diskette.

Preparing Diskettes for Use: You can prepare a diskette using a command (initialize diskette), or by replying to an inquiry message reported during the save operation. Diskettes have a special format used for a save and restore operation. Use the Initialize Diskette (INZDKT) command and specify FMT(*SAVRST). If active files are on the diskette, you can clear them at this time. If you specified CHECK(*YES) (which is the default), an inquiry message is sent to the operator before active files are cleared.

If the diskette was already initialized to the correct format, you can use the Clear Diskette (CLRDKT) command, or specify the CLEAR(*ALL) on the save command. The Clear Diskette (CLRDKT) command takes less time than the Initialize Diskette (INZDKT) command because the CLRDKT does not have to reformat the diskette.

Considerations for Using Save Files

Using a save file allows you to save and restore objects without first placing a tape or diskette in a tape or diskette unit, and to send objects from one AS/400 system to another over communications lines. You can use the save file as an online container to save the contents of a single library to run overnight. The next day, save the contents of the save file to tape or diskette using the Save Save File Data (SAVSAVFDTA) command. Objects saved to a save file can be restored directly from tape or diskette, using the RSTLIB or RSTOBJ command.

Among the items to consider for using save files are:

- Performance
- Storage capacity
- Preparing save files for use

Save files that are larger than that supported for the previous release (3 997 574 records) cannot be saved to a previous release using SAVOBJ SAVSAVFDTA(*YES) TGTRLS(*PRV), or sent to a previous release system using the Send Net File (SNDNETF) and the Receive Net File (RCVNETF) commands.

Performance When Using Save Files: Performance can vary, depending on other disk activity. Save files can be created on or moved to an ASP for improved performance and additional protection from system disk device failures. For information on user ASPs, refer to "General Information about Auxiliary Storage Pools" on page 14-7.

Save File Storage Capacity: The maximum capacity of a save file is 5 865 293 records, or approximately 3 000 300 000 bytes. You can specify the maximum size of the save file on the Create Save File (CRTSAVF) command.

Data compression can be specified on the save commands to reduce the space required for the save file, and the amount of media needed for the SAVSAVFDTA command. (Data compression is not an option on the SAVSAVFDTA command.) For more information on data compression, refer to "Data Compression and Decompression" on page 4-6.

Preparing Save Files for Use: When saving to a save file that already contains data, use the Clear Save File (CLRSAVF) command, specify CLEAR(*ALL) on the save command, or by reply to an inquiry message reported during the save operation.

Saving the Save File Data: There are two ways to save the save file data:

- With the SAVSAVFDTA command, only the data is saved. The description of the save file object is not saved. The save date and time of the save file are not updated.
- When you use the Save Object (SAVOBJ) or the Save Library (SAVLIB) command with SAVFDTA(*YES) specified, both the object description and the data are saved. The save date and time are updated for the save file.

Using Control Language (CL) Commands for Save Files: Use the following CL commands with save files:

- The Create Save File (CRTSAVF) command creates a save file that can be used with save and restore commands to store data otherwise written to tape or diskette. A save file can also be used as a container to send objects to another AS/400 user on the systems network architecture distribution services (SNADS) network.
- The Change Save File (CHGSAVF) command changes one or more of the attributes of a save file, such as the maximum number of records.
- The Override Save File (OVRSAVF) command overrides or replaces certain attributes of a save file, or overrides any file with a save file.
- The Display File Description (DSPFD) command displays the attributes of the save file.
- The Clear Save File (CLRSAVF) command clears the contents of a save file.
- The Display Save File (DSPSAVF) command displays the save and restore information in a save file, or the contents of the save file.
- The Save Object (SAVOBJ) or the Save Library (SAVLIB) command can be used to save the description of the save file and to optionally save the data to tape, diskette, or another save file in a different library.
- The Save Save File Data (SAVSAVFDTA) command writes the contents of a save file to either tape or diskette.

Several commands used for save and restore operations also apply to save files.

Save File Security: The authority you grant for a save file is the same as for any file. Be careful when granting authority for save files. The authority you grant to the save file allows access to objects in the save file. For example, the same file can be read from and written to by a high-level language program. The authority you grant for a particular save file should depend on what objects are in the file.

Consider the following factors when granting authorities to save files:

- A user with operational and read authority can read records and restore objects from the save file, and save the contents of the save file to tape or diskette.
- A user with operational, read, and add authority can write records and save objects in a save file.
- A user with operational and object management authority can clear the contents of a save file using the CLRSAVF command. The clear operation is required first when replacing existing records in a save file.
- A user with either save system (*SAVSYS) special authority or object existence (*OBJEXIST) authority for the file can save the description.

Opening a Save File Table 4-5 on page 4-10 lists the parameters that apply to save files, and indicates where each parameter is specified.

Table 4-5. Parameters Used by Save Files

Description	Specified in CL Parameter	Specified in CRTSAVF Command	Specified in RPG Program	Specified in COBOL Program	Specified in BASIC Program	Specified in PL/I Program	Specified in OVRSAVF Command
Record length			X	X	X	X	
Extend	EXTEND			X			*NO or *YES
Position	POSITION						*RRN with relative record number
Maximum records	MAXRCDS	*NOMAX or number of records					
File wait time	WAITFILE	*IMMED, *CLS, or number of seconds					*IMMED, *CLS, or number of seconds
Shared file	SHARE	*NO or *YES					*NO or *YES
Secure	SECURE						*NO or *YES
Authority	AUT	*EXCLUDE, *CHANGE, or *ALL					
Text	TEXT	*BLANK or text					
Auxiliary Storage Pool	ASP	1–16 *LIBASP					

The following considerations apply when opening a save file:

- Parameters specified in the file are overridden with parameters specified in the program. Program-specified parameters are overridden with parameters on the OVRSAVF command.
- The fewer parameters specified in the program and on the OVRSAVF command, the faster the file is opened.
- If no record length is specified in the high-level language program that opens the file, a length of 528 bytes is assumed. If the program specifies a record length value, it must be 528.
- For an input file, if the relative record number specified for the parameter does not exist in the file, an error message is sent and the file is not opened.
- If you do not specify EXTEND(*YES) for a save file opened for output, the save file must be empty. If the save file is not empty, you receive an inquiry message to clear the file or cancel the operation. The file can only be cleared if the job trying to open the file has operational and object management authority. You can use the CLRSAVF command before the file is opened to ensure that it is empty and to avoid the inquiry message.
- A save file cannot be opened by another program or job for input or output if it is already open for output, or if it is in use for a save or restore operation.
- A save file can be opened by more than one program or job for input. While it is in use, the save file cannot be opened for output.

Input and Output Operations on a Save File: The following considerations apply to input and output operations on a save file:

- Records are always read and written sequentially. The records read from a save file contain sequence and parity information that is validated when the records are written into another save file. This information ensures that the records are processed in sequence and have not been changed.

  If you attempt to write a record that changed since it was retrieved from another save file, or if you attempt to write a record that is not the next record in sequence, an escape message is sent to report the error.
- If you attempt to read a record from a save file but the record is not available because of an auxiliary storage error, the notify message CPF5030 is sent to your program. Your program can look for this error and do another read operation to get the next available record in the file. You will not be able to write any of the records retrieved after an

auxiliary storage error into another save file, because the sequence of records was broken by the unreadable records. You may be able to use the data to manually reconstruct the data in the saved object.

- The force-end-of-data (FEOD) function is valid for both input and output.

  For an input file, FEOD signals end-of-file to the program that does the operation.

  For an output file, FEOD ensures that buffered output records are written to the file so they are not lost after the FEOD operation completes, even if the job or system fails.

File-Dependent Attributes for a Save File: Following are the file-dependent attributes for a save file:

- The following file-dependent attributes apply when the save file is open:
  - For input operations, the first record returned for a read operation is the one specified by the parameter POSITION when the file is opened. After the first record is read, all remaining records are returned sequentially to the end of the file.
  - For output operations, new records can be added to the end of records already in the file (specified using the EXTEND parameter). Each save file record contains sequencing information used by the system to ensure that a record is not skipped or written more than once.
  - If no record length is specified in the high-level language program that opens the file, a length of 528 bytes is assumed. If the program specifies a record length value, it must be 528 bytes.
- No file-dependent parameters (such as format name) can be specified for read or write operations with a save file. Any file-dependent parameters specified are ignored.

Damage to a Save File: A save file is marked partially damaged if an attempt to read a record or restore an object from the file encounters an auxiliary storage error. When a save file is partially damaged, use file read operations to retrieve records that are not on the damaged part of auxiliary storage. You can restore objects from a partially damaged save file other than the objects on the damaged part of auxiliary storage. The objects on the damaged portion of the auxiliary storage within the save file cannot be restored. When a file is marked partially damaged, you cannot add more records to it until it is cleared.

Clearing a Save File: Clearing a save file releases the auxiliary storage allocation to reduce the size of the object, and resets the partial damage mark on the file if it was previously marked damaged. A save file is cleared in one of two ways:

- By using the CLRSAVF command prior to the save operation.
- By specifying CLEAR(*ALL) as the first part of a save operation to the save file. If CLEAR(*ALL) is not specified and emptied, a message is sent asking whether or not you want to clear the file.

Sending Network Files: The only objects you can send with the Send Network (SNDNETF) command are database file members or save files. The SNDNETF command creates a save file and copies the information into it. Once the file has been received using the Receive Network File (RCVNETF) command, the copy on the source system is not saved. Consider backing up the information on the remote system.

Other objects (such as programs or commands) must be saved in a save file before they can be sent using the SNDNETF command.

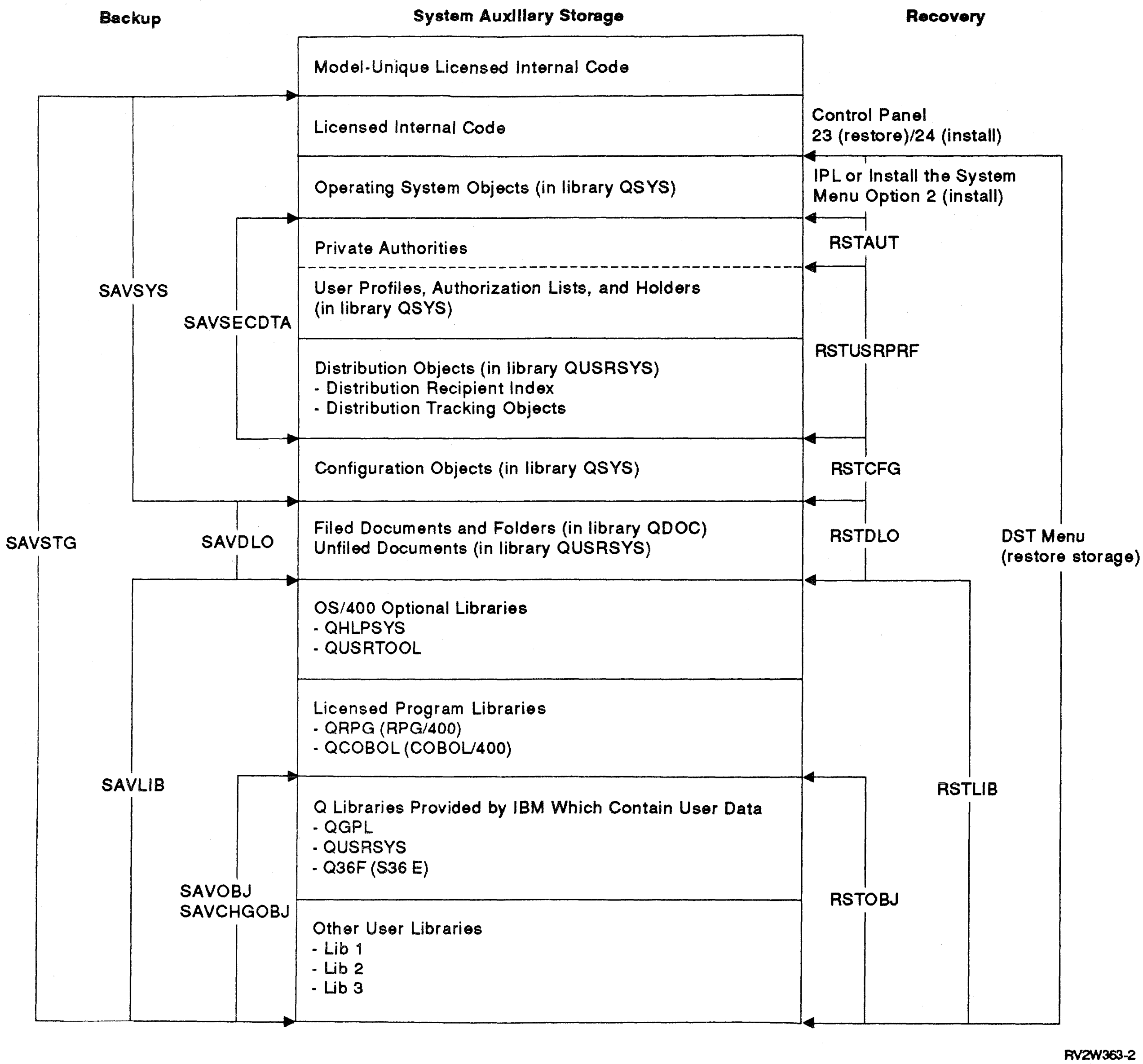
Note: Do not use save files to save objects on a system at the current release to distribute them to a system at a previous release unless TGTRLS(*PRV) is specified on the save command. The current release to previous release rules still apply.

Part 2. System and Object Recovery

Chapter 5. Description of the Save Processes

This chapter describes saving the different types of objects and considerations for the save processes.

To understand the save processes for different types of objects, the following illustration provides an overview of the system ASP and the commands used to save different types of objects.



Objects That Can Be Saved and Restored

You can save and restore an entire system or individual objects in the system. The objects that can be saved or restored are as follows:

- The system, which includes the licensed internal code, OS/400 licensed program library (QSYS), device configurations, and security objects.
- All security objects, including user profiles, authorization lists, authority holders, and distribution objects associated with the mail.
- All objects in all user libraries (including the general purpose library (QGPL) and licensed program libraries). You can restore either one or all of the libraries.
- All objects in one or more named libraries can be saved.
- Only the changed objects in one or more named libraries.
- A group of objects by generic name or by object type in one or more named libraries.
- Several objects by name in a single library.
- A single object in a library.
- A list of members by generic or specific name for specific database files.

Saving User Libraries and IBM-Supplied Libraries

You can save one or more libraries by using the Save Library (SAVLIB) command.

Using one SAVLIB command, you can save:

- Only one library when saving to a save file.
- From 1 to 50 libraries. Specifying multiple libraries allows overlapping processing, which can improve performance.
- All libraries (including the IBM-supplied libraries QGPL, QUSRSYS, and licensed program libraries) to diskette or tape using SAVLIB LIB(*NONSYS).
- All user libraries to diskette or tape using SAVLIB LIB(*ALLUSR). User libraries are defined as libraries that the user creates or IBM-supplied libraries that contain user data. IBM-supplied libraries that do not contain user data are not saved.
- All IBM-supplied libraries to tape or diskette using SAVLIB LIB(*IBM). IBM-supplied libraries that contain only IBM objects are saved.

Table 5-1 on page 5-3 shows which libraries support the *ALLUSR and *IBM values:

Table 5-1. Comparison of Special Values for the LIB Parameter

Library Name	SAVLIB LIB(xx) SAVLIB LIB(*NONSYS)		SAVLIB LIB(*IBM)		SAVLIB LIB(*ALLUSR) SAVCHGOBJ LIB(*ALLUSR)	
	Saved	**Not Saved**	**Saved**	**Not Saved**	**Saved**	**Not Saved**
QDOC		X		X		X
QGPL	X			X	X	
QGPLTEMP	X		X			X
QGPL38	X			X	X	
QPFRDATA	X			X	X	
QRCL	X			X	X	
QRECOVERY		X		X		X
QRPLOBJ		X		X		X
QSPL		X		X		X
QSRV		X		X		X
QSYS		X		X		X
QSYS2	X		X			X
QS36F	X			X	X	
QTEMP		X		X		X
QUSER38	X			X	X	
QUSRSYS	X			X	X	
QUSRTEMP	X		X			X
QDSNX	X			X	X	
#LIBRARY	X			X	X	
#CGULIB	X		X			X
#COBLIB	X		X			X
#DFULIB	X		X			X
#RPGLIB	X		X			X
#SDALIB	X		X			X
#SEULIB	X		X			X
#DSULIB	X		X			X
Qxxxxxxxxx[1]	X		X			X

Notes:

1. Qxxxxxxxxx refers to any other library that starts with the letter Q. These libraries are processed as IBM libraries that do not contain any customer data.
2. Libraries QDOC, QRECOVERY, QRPLOBJ, QSPL, QSRV, QSYS, and QTEMP are not saved with the SAVLIB command.
3. The system must be in a restricted state to ensure a complete save of all libraries.
4. Objects that are in use in a library are not saved. This library should be saved at a later time when the objects are not being used.
5. Users cannot update objects in libraries that are being saved.

There are three data areas that store the save information when you specify one of the special values for the LIB parameter. These data areas are:

- QSAVLIBALL contains the information for the SAVLIB LIB(*NONSYS) operation.
- QSAVALLUSR contains the information for the SAVLIB LIB(*ALLUSR) operation.
- QSAVIBM contains the information for the SAVLIB LIB(*IBM) operation.

You can use the Display Object Description (DSPOBJD) command to display the description of the QSAVLIBALL, QSAVALLUSR, or QSAVIBM data areas to determine the date and time of the last save operation.

If you specify one of the three special values (*ALLUSR, *IBM, or *NONSYS) on the LIB parameter, you can exclude from 1 to 50 libraries from the save operation. You can specify the list of libraries to be excluded on the OMITLIB parameter.

Saving Objects

You use the Save Object (SAVOBJ) command to save one or more objects. Objects are not affected by this command unless it specifies that storage is to be freed.

You can specify up to 50 different libraries on the Save Object (SAVOBJ) and Save Changed Object (SAVCHGOBJ) commands, although you must save all objects (OBJ(*ALL)) when you are saving from more than one library. Libraries are processed in the order specified in the library (LIB) parameter. Specifying more than one library allows overlapping processing, which can improve performance.

You can specify only one library when using the SAVOBJ and SAVCHGOBJ commands to save objects to a save file.

Single Object or More Than One Object: You can save a single object or more than one object (not a generic name) by specifying:

- A list of object names
- A list of object types
- *ALL for object name
- *ALL for object type

Group of Objects Using a Generic or Specific Name: You can save a group of objects by a generic name or by generic name and type. However, all the objects must be in the same library.

To specify a generic name, you use the common characters of object names that identify the group of objects. For example, OBJ(ORD*) would save objects whose names begin with ORD, such as ORDPART, ORDREQ, and ORDSTK.

In addition to specifying generic names, you can also save objects by specific name. Both generic and specific names can be mixed in lists of up to 50 names. For instance, you can save the file CUSTNO as well as all other files in that library beginning with ORD and MST by specifying OBJ(CUSTNO ORD* MST*).

Saving Changed Objects: The SAVCHGOBJ command is like the SAVOBJ command except that the SAVCHGOBJ command saves only *changed* objects or members. You can save changed objects using the following information:

- Save all changed objects from all user libraries. The *ALLUSR value allows you to save all of the changed objects (with the exception of office objects) in a single operation. (See Table 5-1 on page 5-3 to determine which IBM-supplied libraries are included when *ALLUSR is specified.)
- If you specify the special value *ALLUSR on the LIB parameter, you can exclude from 1 to 50 libraries from the save operation. You can specify the list of libraries to be excluded on the OMITLIB parameter.
- The changed object information kept by the system is a date and time stamp. When an object is created, the time stamp is placed in the changed field and any change to the object causes the date and time stamp to be updated.
- As with the SAVOBJ command, you can save all objects that meet a generic selection.
- Because the SAVCHGOBJ command can save only those objects or members that have changed, you can reduce the time needed to save objects and may be able to alter the frequency of your save operations.
- Use the SAVCHGOBJ command to specify the date and time, in the job date and time format, from which you want your changed objects saved. If you do not use the REFDATE and REFTIME parameters, the system uses the date and time of the last SAVLIB operation for that library. These parameters improve the performance of the SAVCHGOBJ command.

  Use the Display Object Description (DSPOBJD) command and specify *FULL to display the date and time of the last change for a specific object. Use the Display File Description (DSPFD) command to display the last change date for a database member.

- In a typical environment, you might be using the SAVLIB command once a week and the SAVCHGOBJ command every day. Because the default for SAVCHGOBJ is from the last SAVLIB operation, the media produced by SAVCHGOBJ tends to grow during the week.

  For example, if you use SAVLIB on Sunday, the Monday SAVCHGOBJ saves any changes from Monday, the Tuesday SAVCHGOBJ saves both the Monday and Tuesday changes, and so on. The default is to produce a cumulative result that is reset by the next SAVLIB. This causes more media to be produced, but simplifies recovery in that only the last SAVCHGOBJ media is needed.

- The default value of *NO for the object journal (OBJJRN) parameter indicates that objects associated with the journal and journal receivers are not included in the SAVCHGOBJ operation. If you save the journal receivers for physical files, you do not need to save large physical files as frequently.

- Different functions cause the system to update the time stamp. For example, if you issued the Change Job Description (CHGJOBD) command, the system time stamps the job description as changed. If a file is updated, the system time stamps the file as changed. This function exists for both source and data files.

Some common operations that result in a change of the date and time are:

- Create commands
- Change commands
- Restore commands
- Add and remove commands
- Journal commands
- Generic commands (for example, Grant Object Authority (GRTOBJAUT), Move Object (MOVOBJ), and Revoke Object Authority (RVKOBJAUT))

Operations that affect only the content or attributes of a member change only the member's date and time (for example, clearing a physical file member using the Clear Physical File Member (CLRPFM) command, or updating the member through a utility (for example, source entry utility (SEU) or a user program).

The SAVCHGOBJ command can be used for backing up typical source files. Normally, a source file has many members, and only a small percentage of members is changed every day. SAVCHGOBJ saves the file description and any members that have changed.

Some operations change the change date and time of the file and all of its members (for example, the CHGOBJOWN, RNMOBJ, and MOVOBJ commands).

An IPL causes all job queues and output queues to be changed. The system does not record the change date and time for the following object types:

- *Message queue.* When a message is sent or received.
- *Data queue.* When an entry is sent or received.
- *Save file.* When a save command writes data into the object.

Changes to the following types of objects are not recorded by the system:

- Libraries
- User profiles
- Configuration objects (for example, line descriptions and controller descriptions)

Saving Database Files

You can save individual database members only with the SAVOBJ command using the FILEMBR parameter. Changed database members can be saved with the SAVCHGOBJ command.

When saving a database file, consider the following:

- The save operation saves all records in the file, but does not save unused storage at the end of the file, regardless of how much was allocated.
- A save operation of a physical file, TYPE(*DATA), with a keyed access path cannot contain more than 7999 members. If the file is a source file, or if it is a data file that does not have a key, the limit is 15 999 members.

- When you save a keyed physical file with TYPE(*DATA), the keyed access path is also saved. If the keyed physical file is TYPE(*SRC), the access path is not saved.
- Journaled database files that started journaling after the save process reference date and time are saved even if you specified OBJJRN(*NO). Use the DSPFD command to display the journaling start date. This date is updated each time a Start Journaling Physical Files

  (STRJRNPF) command is issued, or whenever a member is added with a create or restore operation. For a further discussion of journal management, see Chapter 12, "Database Recovery."
- Only the changed members of a database file are saved by SAVCHGOBJ.
- If you save a database file and free its storage, you cannot save it again if it still exists with its storage freed.

Saving Access Paths: The access path (ACCPTH) parameter on the SAVCHGOBJ command or the SAVOBJ command allows you to save access paths. Normally, only the descriptions of logical files are saved. The logical file access paths are saved under the following conditions:

- ACCPTH(*YES) is specified.

  When you specify ACCPTH(*YES), a message is issued for each physical file saved indicating how many logical file access paths were saved with that physical file.
- All physical files under the logical file are being saved and are in the same library.

  If the logical file is not in the same request, the access path is still saved. The logical file does not need to be in the same library as the underlying physical files. However, using different libraries is not desirable when restoring is needed and neither object exists on the system. In this case, the system rebuilds the access path.
- The logical file is MAINT(*IMMED) or MAINT(*DLY).

Saving a List of Members: You can save a list of members for a database file using the FILEMBR parameter of the SAVOBJ command. This list may consist of specifically named members, generically named members or a combination of both specifically and generically named members.

The FILEMBR parameter is used to specify:

- A list of file members (specific or generic) for a specific database file
- The same group of members from more than one file

The following restrictions apply to the FILEMBR parameter:

- Each database file specified in the FILEMBR parameter must also be specified in the OBJ parameter by its complete name, a generic name, or *ALL.
- Generic names are not valid for the database file name.
- Generic names are valid for the member name.

  If a generic name is used, the file is not saved if it does not have members that fit the generic name. If all files specified by the FILEMBR parameter are not saved, a diagnostic message is sent and the save operation ends with an escape message giving the number of files not saved.

  If a name that is not generic is used, the specific members must exist in the file for any part of the file to be saved.
- The OBJTYPE parameter must be *ALL or include *FILE.
- The save must be a single-library save operation.

For examples of saving a list of members, see the topic "Examples of Saving a List of Members" on page 9-18.

Saving Source Files: When source files are backed up by either the SAVLIB, SAVOBJ, or SAVCHGOBJ command, you can display the exceptions if all the source cannot be saved. For example, a message can be sent that gives the status of the save source file process. An example of this program is found in "Using a

Status Program for Saving Source Files" on page B-5.

Consider having a separate library or libraries restricted to production objects. Then, if the entire system must be restored, you can restore production objects first.

Saving Files Being Journaled: If you save files while they are being journaled or while their access paths are being journaled and the journal exists on the system at the time of the restore operation, then the files are journaled again provided the library for the journal and the libraries for the files are in the same ASP.

When you save a file that is being journaled, an entry is written to the journal to indicate that the file was saved.

When you save access paths that are being journaled, no entry is written to the journal to indicate it was saved.

Saving Journals and Journal Receivers

You can use the SAVOBJ, SAVCHGOBJ, SAVLIB, and SAVSYS commands to save journals and journal receivers. It is best to save the old journal receiver when you change the journal to a new receiver (CHGJRN command). This is the most efficient method and provides for a backup copy.

However, you can save a journal, or journal receiver, even when files are being journaled to it, and when it is not at a commitment boundary. (For more information about commitment boundaries, see the topic "Starting Commitment Control" on page 12-29.) The save operation always starts at the beginning of the journal receiver. An informational message is sent when you save a journal receiver currently attached to a journal.

Storage for a journal receiver can be freed if the journal receiver is not currently attached and all previous journal receivers are deleted or have their storage freed. If you are using two journal receivers, one of the two journal receivers can be saved with storage freed at any time after it is detached provided the other member of the pair keeps its storage.

Saving Save File Data Using the SAVSAVFDTA Command

A save file can contain objects from only one library. When objects are saved to a save file, they can be saved to tape or diskette (using the Save Save File Data (SAVSAVFDTA) command). When the data in a save file is saved to tape or diskette, the data can be restored as if it was originally written to tape or diskette. However, the save file cannot be restored.

Saving Save File Data Using the SAVFDTA Parameter

The save file data (SAVFDTA) parameter can be specified on the Save Library (SAVLIB), Save Object (SAVOBJ), or the Save Changed Object (SAVCHGOBJ) command. This parameter allows you to specify, for save file objects, whether the description of the file is saved, or if both the description and the contents of the file are saved. The default SAVFDTA(*YES) saves both the description and the contents.

If the contents of the save file are saved, the restore operation restores the entire save file. The OPTION parameter allows you to control whether the objects are restored over existing objects.

If you are saving the save file for a system at a previous release (TGTRLS(*PRV)) and specify SAVFDTA(*YES), the system does not verify the objects saved into the save file. Therefore, an attempt to restore objects from the save file to a previous release system may not be successful.

The following restrictions apply when specifying SAVFDTA(*YES):

- When the contents of the save file are saved (SAVFDTA(*YES)), and the save media is the same save file in the same library (SAVF parameter), only the description of the save file is saved. Message CPI374B *SAVFDTA ignored for file &1 in library &2* is sent and the save operation continues.
- The SAVFDTA(*YES) parameter cannot be specified on the Save System (SAVSYS) or the Save Licensed Program (SAVLICPGM) command. However, both of these commands save the contents of the save files (by default).

- When the contents of a save file are saved using SAVFDTA(*YES), the save file must be restored before the objects can be restored from the save file.

Saving Spooled Output Files

The system does not directly support saving spooled files on an output queue. However, you can accomplish the same thing by copying the spooled files to a database file using the Copy Spooled File (CPYSPLF) command and then saving the database file with a save command (such as the Save Object (SAVOBJ), Save Library (SAVLIB), or Save Changed Object (SAVCHGOBJ) command).

This approach saves the textual data in the file. However, advanced function attributes, such as graphics and variable fonts, are not saved. For an example of this approach, see the Spool Control (SPLCTL) tool in library QUSRTOOL.

Saving Licensed Programs

Use the Save Licensed Program (SAVLICPGM) command to save licensed programs on your system. The SAVLICPGM command is not for backup purposes, but can be useful for distributing IBM licensed programs to other systems. Refer to the *Central Site Distribution Guide*, SC41-9993, for more information about saving licensed programs to distribute to other systems.

Saving Security Information

Security information can be saved to tape or to a save file without doing a complete save system (SAVSYS) command and without requiring the system to be in a restricted state. All subsystems, batch jobs, and interactive jobs must be ended to place the system in a restricted state.

The Save Security Data (SAVSECDTA) command saves the same security information as the SAVSYS command. The advantage is that the command can be run at any time. The system does not need to be in a restricted state. A user must have save system (*SAVSYS) special authority to use the SAVSECDTA command. The following information is saved:

- User profiles
- Authorization lists
- Authority holders
- Distribution objects (only if MAIL(*YES) is specified on the SAVSECDTA command)

If the Save Security Data (SAVSECDTA) command used a save file, the SAVSAVFDTA command must be used to save the information to tape if it is to be restored at a later date. After the information is saved to tape, it can be restored only by using the Restore User Profile (RSTUSRPRF) command. A restricted system is required when restoring the security data that was saved to tape.

Consider the following when using the SAVSECDTA command:

- If the system is not in a restricted state, changing or creating user profiles and granting authority is allowed, but deleting a user profile is prevented.
- The number of distribution (mail) objects being saved.
- Changes made to user profiles while the save security data operation is running may not be reflected in the save media depending on when the changes occurred during the save.
- The time required to run the SAVSECDTA command is determined by the number of user profiles and individual authorities on the system.

Considerations for the Save Processes

The following considerations and restrictions apply when using the save processes.

Limitations When Saving Objects

The maximum number of 512 byte records that can be stored in a save file is:

- 5 865 293 for Version 2 Release 1
- 3 997 574 for Version 1 Release 3 and earlier

Save operations create one or more directories on the save media for each library saved. Each directory contains a list and description of objects to be saved. Each directory can contain up to 32 766 objects and 16MB of description data. Because multiple directories can be created for each library, the chances of exceeding the limits is rare. However, the following objects are always grouped together in a single directory:

- All objects in a library, when the save device is a diskette unit
- All objects in a library, when the save command specifies TGTRLS(*PRV)
- All database file objects in a library that are related to each other
- All distribution (mail) objects

In some cases, the save operation counts one OS/400 object as several objects, and these must be considered separately in the 32 766 limit, as shown in the following list (for each object, count one, with the considerations and exceptions indicated):

- For each saved subsystem description, count eight objects as saved.
- For each saved physical file of any type that has an arrival sequence access path, count one object as saved for each physical file member.
- For each saved physical file of type *DATA that has a keyed sequence access path, count two objects as saved for each physical file member. For source files with keyed sequence access paths, count one for each member.
- If a physical file with no members is saved, or a logical file description or a description of a job queue, output queue, message queue, data queue, or save file is saved, count no object as saved.
- If you specify ACCPTH(*YES), access paths for all logical files based on physical files in the library are also saved. See also "Examples of Saving Access Paths" on page 9-18. for more information. Count one object as saved for each keyed logical file member that owns an access path over the data being saved. Also count one object as saved for each secondary in each join logical file.
- If a programming change has been made to a library specified on the Save Library (SAVLIB) command, count one object as saved.
- Count one object as saved for the save/restore directory that contains the descriptions of the saved objects.

Saving Object Descriptions

Table 5-2 shows object descriptions that are saved although the contents are not saved.

Table 5-2. Object Descriptions That Are Saved

Object Description Saved	Contents Not Saved
Message queues (*MSGQ)	Messages
Job queues (*JOBQ)	Jobs
Output queues (*OUTQ)	Spooled files
Data queues (*DTAQ)	Data queue entries
Logical files	Physical files making up logical files
Journals	Journals (journal receiver contents are saved). Access paths can be saved with the physical file

Freeing Storage

When you save an object, it is not removed from the system. Rather, a copy is made on the save media. If you do not want to keep the object on the system after saving it, you can release some of the area for other uses by specifying STG(*FREE). Using STG(*FREE) deletes the contents of the object but saves the object description and search values. This process is called freeing storage and is supported for files (except save files), journal receivers, programs, and documents. When a database file is freed, the member information is lost. If you save a data base file or document and free its storage, the object cannot be saved again if it still exists with its storage freed.

After storage has been freed, the object only exists offline and must be restored to be used. The storage occupied by logical file access paths is not freed. Operations such as moving or renaming the object can continue to be performed because the description of the object remains on the system with the exception of document objects. The object description consists of the following types of information.

- Object name
- Object type
- Date and time of the save operation

- Owner name
- Object size at the time of the save operation
- Storage required on the system
- Text description
- Any public authority
- For database files, attributes of the files and members

You can also specify STG(*DELETE) on the Save Document Library Object (SAVDLO) command to delete any filed documents after they were saved. No object description remains on the system if STG(*DELETE) is used. For documents, STG(*DELETE) deletes the document description, search values, and document contents.

Chapter 6. Description of the Restore Processes

Objects that were saved by separate commands must be restored by separate commands. If you try to restore them with a single command, not all of them are restored. When you use the Save Library (SAVLIB) command, objects can be restored using either the Restore Library (RSTLIB) or the Restore Object (RSTOBJ) command. You cannot use the RSTOBJ command on a data dictionary that has been saved by using the SAVLIB command. If you saved an object using the SAVOBJ or SAVCHGOBJ command, it can only be restored with the RSTOBJ command.

You can restore objects to a system whether or not those objects (or their definitions) exist on that system. If the object had its storage freed or deleted, or if it previously did not exist on the system, it can be restored. When you restore an object, the original object existing on the system is replaced by the copy on the save media. If a media error occurs during the replacement, the original object may be unusable because it is partially overlaid. If you suspect media errors exist, restore to a different library.

Generally, the system to which you are restoring objects must be the same or at a higher release level than the system from which the objects were saved. The target release (TGTRLS(*PRV)) parameter allows you to save objects in a previous release format so they can be restored on a previous release system. Data interchange can also be used to send data to a system at a previous release level. For more information on release levels, see Chapter 8, "Release to Release Support."

Table 6-1 shows which restore commands can be used, depending on the save command you used.

Table 6-1. Restore Commands to Use

Save Command You Used	RSTOBJ	RSTLIB	RSTUSRPRF RSTAUT	RSTCFG	RST-LICPGM	RSTDLO
SAVOBJ	X					
SAVLIB	X	X				
SAVCHGOBJ	X					
SAVSYS	X[1]		X	X		
SAVSAVFDTA	X	X	X			X
SAVLICPGM					X	
SAVDLO						X
SAVSECDTA			X			

Note: You can only run the RSTOBJ command against files with labels beginning with QSS1.

You can restore documents and folders only with the Restore Document Library Objects (RSTDLO) command.

You can restore a library that was saved on your AS/400 system or another AS/400 system, even though that library did not previously exist on the system. Restoring an object requires that the library exists, although you can restore an object to a library different than the one from which it was saved to create a duplicate object. All objects restored by a single command must be in the same library, except when *ALLUSR, *IBM, or *NONSYS is specified on the RSTLIB command. If only selected objects from a library were saved, only those objects, not the entire library, can be restored.

When you restore from diskette and multiple copies of a library were saved to the same diskette, specify a save date and time, or a diskette file label to distinguish the save operations. Doing so allows you to specify which copy you want to restore. If you do not specify a save date and time, or a file label, the first copy encountered on the volume is restored.

When restoring from tape, and multiple copies of the library were saved to the same tape, the save date and time or SEQNBR allows you to distinguish between the files. If you do not know the save date and time, file label, or the sequence number, use the Display Diskette (DSPDKT) or the Display Tape (DSPTAP) command with DATA(*SAVRST) specified to view the contents of the media.

When you restore a library, you do not have to specify the volume on which the library was saved. The default (*MOUNTED) is to use the volume currently placed on a tape or diskette unit. If you are restoring over an existing object and want the volume containing the most current saved copy, specify VOL (*SAVVOL).

You can restore only those objects that were previously saved on your system, or on another AS/400 system or System/38. For more information about restoring System/38 and System/36 objects, see the *System/38 to AS/400 Migration Aid User's Guide and Reference* and *System/36* *to AS/400 Migration Aid User's Guide and Reference.*

Restoring Libraries

Use the Restore Library (RSTLIB) command to restore a single saved library, all libraries, all saved user libraries, or all IBM libraries. Library QDOC is not restored with the RSTLIB command. Objects in library QDOC can be restored using the Restore Document Library Object (RSTDLO) command.

Using the RSTLIB command, you can restore:

- All libraries (including the IBM-supplied libraries QGPL, QUSRSYS, and licensed program libraries) that were saved with SAVLIB LIB(*NONSYS) from diskette or tape using RSTLIB SAVLIB(*NONSYS).
- All user libraries that were saved with SAVLIB LIB(*ALLUSR) from diskette or tape using RSTLIB SAVLIB(*ALLUSR).
- All IBM-supplied libraries that were saved with SAVLIB LIB(*IBM) from tape or diskette using RSTLIB SAVLIB(*IBM). Only IBM-supplied libraries that contain IBM objects are restored.

The RSTLIB command restores the entire library, including the library description, object descriptions (for logical files, job queues, message queues, output queues, and data queues), and the contents of other objects. This command also restores status information for programming temporary fixes (PTFs) that were in the library at the time the library was saved. Restoring a library destroys the PTF status information that currently exists in the library and replaces it with the PTF status information contained on the save media, if any.

The following options are used when restoring old, new, and freed objects:

- OPTION(*OLD). Only old objects that already exist on the system are replaced in a library.
- OPTION(*NEW). Only objects not found on the system are added to a library. The old objects are not replaced.
- OPTION(*ALL). Old objects are replaced and new objects are added to a library. *ALL is the default.
- OPTION(*FREE). Only those objects whose storage has been freed are restored.

The following example restores the general-purpose library QGPL:

```
RSTLIB  SAVLIB(QGPL) DEV(TAP01) OPTION(*ALL) RSTLIB(*SAVLIB)
```

Old objects are replaced, and new objects are added to the library (*ALL specified for OPTION).

Restoring Objects

The Restore Object (RSTOBJ) command is like the RSTLIB command except the additional parameters OBJ and OBJTYPE are used for object selection. Also, the library description and status information for PTFs are not restored using the RSTOBJ command. Objects can be restored to only one library with the RSTOBJ command.

Restoring Database Files

Restoring a database file involves the following considerations:

- A saved version and a copied version of the same file are not the same and cannot be used interchangeably in a restore operation. If a copy is made of file A (using the Copy File (CPYF) command) and then both file A and the copy are saved, the saved version of the copy cannot be used to restore file A. Only the saved version of file A can be used to restore file A. Figure 6-1 on page 6-3 illustrates the save process for files. Figure 6-2 on page 6-3 illustrates how files can be restored.

  There is a restriction if you are journaling File A. You cannot apply journal changes made to File A to any other file. A journal identifier is used to ensure this regardless of the files external name.

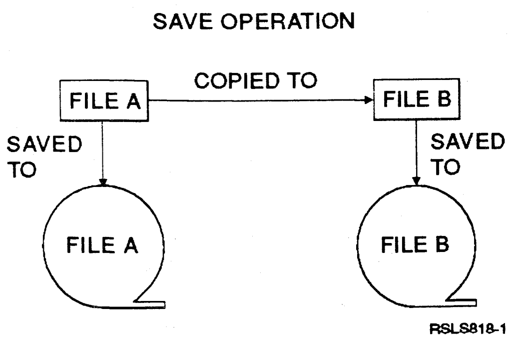


Figure 6-1. How Files Are Saved

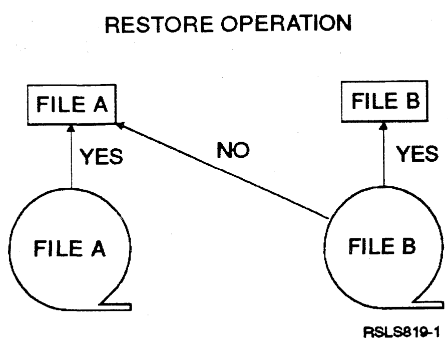


Figure 6-2. How Files Can Be Restored

- When you are restoring to an existing database file and MBROPT(*MATCH) is specified (the default), the list of members in the database file must be identical to that on the media.
- When you restore an existing database file, only the data is restored. If you want to restore the file attributes as they existed at the time of the save operation, delete the file, then restore it.
- When you restore a file, no member of the file can be used during the restore operation, even through logical files. The file is exclusively locked during the restore operation.
- If you specified the allow object differences (ALWOBJDIF(*ALL)) parameter when restoring files or members and the creation date of the file or members does not match the creation date on the save media, the system renames the file or member on the system and then restores the copy from the save media. A message is sent to the user to indicate that the file has a new name and the saved copy has been restored. If you specify the ALWOBJDIF(*NONE) and the creation date does not match, the file or member is not restored.
- When you restore a keyed physical file, its access path is also restored, unless it is a keyed source physical file. In most instances, you will not have keyed physical files because the Create Source Physical File (CRTSRCPF) command will create a non-keyed source file by default. When you restore a keyed source physical file, the access path is rebuilt after the restore operation. After restoring a keyed source physical file or a keyed logical file with ACCPTH(*NO) specified during the save operation, the file and physical files associated with the logical files are unavailable while access paths are rebuilt. This is true for both immediate and delayed maintenance. See the topic "Restoring Access Paths" for information about immediate and delayed maintenance.

Restoring Access Paths

Access paths are restored only if they were saved and all the physical files they are dependent on are restored at the same time.

The logical file must exist on the system when the restore command is issued. If the logical file does not exist on the system, it must be restored as part of the same command, and must exist in the same library as the dependent physical file, before the access path can be restored.

If ACCPTH(*YES) was specified when saving a physical file, and a dependent logical file was in a different library, and the logical file was deleted before the restore operation, do the following:

1. Restore the physical file first using the Restore Object (RSTOBJ) command.
2. Remove the physical file members using the Remove Member (RMVM) command.
3. Restore the logical file.
4. Restore the physical files again.

If the access path for a logical file is not being restored, it is automatically rebuilt when the restore operation is performed for a logical file specified as MAINT(*IMMED) or MAINT(*DLY).

When one or more access paths must be rebuilt, the rebuild operations occur in a separate job called QDBSRV*n*, where *n* can be 1 through 4, depending on the number of processors and memory configured. The start control program function (SCPF) job starts these server jobs during the IPL.

Each server job can rebuild one access path at a time. The access path currently being rebuilt can be seen with the Edit Rebuild of Access Paths (EDTRBDAP) command. If the logical file is MAINT(*REBLD) or MAINT(*DLY), the access path is not rebuilt until the file is opened. Sometimes your restore operation is finished before the QDBSRV*n* jobs have finished rebuilding access paths. (Message (CPF3145) is sent to QHST for each access path rebuilt.)

When your restore operation is finished, enter the EDTRBDAP command. If an access path is being rebuilt, the display shows RUN in the *Status* field for the file. See Appendix D, "Initial Program Load (IPL) Process" for more information about access path rebuild during IPL processing.

Restoring a List of Members

You can restore a list of members for a database file using the FILEMBR parameter of the RSTOBJ command. This list may consist of specifically named members, generically named members, or a combination of both specifically and generically named members.

The FILEMBR parameter is used to specify:

- A list of file members (specific or generic) for a specific database file
- The same group of members from more than one file

The default value *ALL causes all file members of files specified with the OBJ parameter to be restored.

The following restrictions apply to the FILEMBR parameter:

- Each database file specified in the FILEMBR parameter must also be specified in the OBJ parameter by its complete name, a generic name, or *ALL.
- Generic names are not valid for the database file name.
- Generic names are valid for the member name.

  If a generic file member name is used, and the file does not have members that fit the generic name, the file is not restored. If all files specified by the FILEMBR parameter are not restored, a diagnostic message is sent and the restore operation ends with an escape message giving the number of files not restored.

  If a name that is not generic is used, the specific members must exist in the file for any part of the file to be restored.
- The OBJTYPE must be *ALL or include *FILE.
- The MBROPT parameter must not have the *MATCH value.
- The storage (STG) parameter must not be set to *FREE.

Restoring a Different Set of File Members

When restoring database files, the set of members in the database need not be identical to the members on the save media. During the restore operation, members are created or replaced according to the following values, which can be specified on the MBROPT parameter on the RSTLIB and RSTOBJ commands:

- *MATCH. The set of member lists on the save media and in the database must be identical, or the restore operation fails. *MATCH is the default.
- *ALL. All members on the save media are restored.
- *NEW. Only those members on the save media that do not exist in the database file are restored.
- *OLD. Only those members on the save media that already exist in the database file are restored.

If the creation date and time stamp differ for either the file or one of the file members, the file is not restored regardless of what you specify on the MBROPT parameter unless ALWOBJDIF(*ALL) is specified.

Restoring Shared Formats

When a database file is restored and that file, before it was saved, had shared the record format of another file, an attempt is made to find the file whose format was shared, and reestablish the original format sharing.

The search for restoring the shared format starts in the library to which the restored file is directed and continues in the library from which the restored file was saved. Following are the results of the search:

- If the sharing file is found and has not been changed (level check) since the save, then no new format is created for the restored file.
- If the sharing file is not found, or it is found but fails the level check, then a new format for the restored file is created with the same definition as the one it initially shared.
- If a format sharing file has been renamed, deleted, or moved to a library other than the save or restore library, a new format is created for the dependent file when the dependent file is restored.

Restoring Journaled Files

If the journal exists on the system before the files are restored, all files that were saved while being journaled (or saved while having their access paths journaled) are journaled again after they are restored, provided the libraries for the files and journals are in the same auxiliary storage pool.

When you restore a file that was being journaled at the time of the save operation, an entry is written to the journal to indicate that it was restored.

When you restore access paths that were being journaled at the time of the save operation, no journal entry is written to the journal to indicate that it was restored.

If the journal is not intact after the restore operation, the restore operation of a file that was being journaled causes a warning message to be issued and journaling is not resumed. This warning message causes a diagnostic message to be sent at the end of the restore operation. (See the topic "Restore Completion Messages" on page 9-7.)

Restoring Logical Files

You can restore a logical file to a library different than the library for the associated physical file. However, the associated physical file must remain in or be restored to its original library location, or be in the same library as the logical file.

If you try to restore a logical file to a library in which it does not exist, the restore operation fails if any of the associated physical files have had their storage freed.

If the access path for a logical file was saved with the physical file, the access path is automatically restored if all the following conditions exist:

- All physical files on which the logical file is based are being restored with the same command.
- All physical files on which the logical file is based are in the same library.
- The logical file has not been changed to MAINT(*REBLD) since the save operation was performed.

When a logical file is restored, it must be dependent on the same physical files as it was when it was saved:

- The logical file is created over the physical files in the library where they are being restored if any of the following occur:
  - The logical file and the associated physical files existed in the same library at the time of the save operation.
  - The logical file and the associated physical files are present in the library where the files are being restored.
  - The logical file and the associated physical files are being restored to the same library.
- If the files are not present in the restore library, then the logical files are created over the physical files in the original saved library.
- If the correct physical files are not found in either library, then the restore operation of

Save and Restore

the logical file fails. If the restore operation is successful, an informational message (CPF3291) is sent to indicate which library was used for associated physical files.

The creation dates of the physical files must not have changed since the logical file was saved. If the date has changed, an informational message (CPF3293) is sent indicating that the physical file has been changed since the save operation, but the restore operation continues.

Restore physical or logical files with dependent logical files before the dependent logical files, unless the physical and logical files already exist on the system. The following considerations apply to restoring logical files:

- If the dependent physical or logical files are in the same library, the system provides the proper sequencing.
- If the files are in different libraries, you must restore the libraries in order, so that the physical or logical files that have logical files built on them are restored first.
- If the depended-on physical or logical files are not restored before you attempt to restore the logical files, restoring the logical files fails.
- This sequencing also applies to other requirements between files, such as shared formats. You can restore those logical files that failed by using the RSTOBJ command.

Restoring Journals and Journal Receivers

You can restore journals or journal receivers only to the same library from which they were saved [1]. Use the RSTOBJ and RSTLIB commands to restore journals and journal receivers. When you are restoring multiple objects with one of these commands, journals and journal receivers are restored before files.

When you use several commands to restore several objects, restore the objects in this order:

1. Journals
2. Files associated with those journals
3. Journal receivers

Restore the journal receivers in the order of newest to the oldest.

You cannot restore a journal to a library containing a journal with the same name because you would overlay the existing journal with old information. If a journal must be restored (because of damage) to a library, the existing journal must be deleted first.

When you restore a journal, the system creates a journal receiver with the same attributes as the receiver that was attached when the journal was saved, and attaches it to the journal. The system attempts to assign the same owner and to create the journal receiver in the same library as the receiver that was attached when the journal was saved. If the owner of the receiver is not found, the receiver is assigned to the default owner (QDFTOWN) user profile. If the library for the receiver is not found, the receiver is created in the same library as the journal being restored. If the auxiliary storage pool (ASP) does not exist for the receivers that were attached to the journal at the time of the save operation, and the journal is restored, then the receivers are usually created in the same ASP as the library for the receiver.

If two receivers were being used when the journal was saved, two receivers are created and attached to the journal. The system-created journal receiver is named by adding a number to the name of the previous journal receiver.

If the attached journal receivers did not end up in the desired ASP, it is recommended that you use the Change Journal (CHGJRN) command to detach the current receiver and attach a new journal receiver immediately after restoring the journal.

You are informed of the new name of the journal receiver in an informational message. If a receiver cannot be created, you receive an error message indicating the journal was not restored.

You cannot restore a journal receiver currently attached to a journal. If the journal receiver

[1] If a reclaim storage (RCLSTG command) operation was done and journals or journal receivers were in library QRCL at the time of the save operation, the journals or journal receivers can be restored to the libraries they were created in.

being restored contains less information than the one on the system, a message is issued, and the restore operation ends.

Restoring Journaled Files to a Different Library

The system assigns a unique internal journal identifier to every object that is journaled. When restoring a journaled file to a library other than the original library, and if the file still exists on the system, the journal identifier of the restored file is changed. No message is sent telling the user that the journal identifier of the restored file is changed.

If you try to apply journaled changes to the restored file, the operation fails and a message containing this journal identifier is sent indicating that no entries were converted or received from the specified journal.

For example, in Figure 6-3, the original file FILEA in LIBX library has an internal journal identifier of Z that is recorded with every journal entry associated with FILEA in LIBX. When FILEA is restored from the media to LIBC library, it is assigned the journal identifier of Y because FILEA still exists in LIBX.

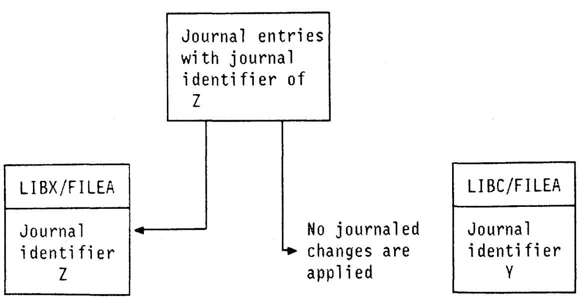


Figure 6-3. Restoring a Journaled File to a Different Library

Any journal operation that references a file by name and involves using journal entries requires that the journal identifier of the file and the journal identifier recorded in the journal entries be the same. Because FILEA in LIBC has journal identifier Y, journal entries with journal identifier Z are not associated with the restored FILEA in LIBC. As a result, journal changes recorded for FILEA in LIBX cannot be applied to FILEA in LIBC. For the same reason, referencing FILEA in LIBC using the Display Journal (DSPJRN), Receive Journal Entry (RCVJRNE), and Retrieve Journal Entry (RTVJRNE) commands does not convert entries for FILEA in LIBX.

Restoring Save File Data

If the save file data is saved to tape or diskette using the SAVSAVFDTA command, and is then restored, it appears as though it was originally coming from tape or diskette. Use the RSTOBJ, RSTLIB, RSTDLO, and RSTUSRPRF commands to restore the data.

If the save file data is saved to tape or diskette using the SAVLIB or SAVOBJ command, the save file must be restored before the objects can be restored.

Restoring Spooled Output Files

You cannot directly save and restore spooled files on an output queue. If you use the technique described in "Saving Spooled Output Files" on page 5-8, you can restore the spooled files by first restoring the database file with a restore command, such as Restore Object (RSTOBJ) or Restore Library (RSTLIB), and then copy the database file members to spooled output files by using the Copy File (CPYF) command specifying TOFILE(QSYSPRT).

Restoring Licensed Programs

Use the RSTLICPGM command to add or replace licensed programs on your system. Refer to the *Licensed Programs and New Release Installation Guide* for more information about installing licensed programs.

Restoring Device Configuration Objects

Use the Restore Configuration (RSTCFG) command to restore all device configuration objects either as a group or individually. Before these objects can be restored, they must be varied off (VRYCFG command).

Examples of device configuration objects are:

- Class-of-service descriptions
- Connection lists
- Controller descriptions
- Device descriptions
- Line descriptions
- Mode descriptions
- Network interface descriptions

Restoring Objects on Auxiliary Storage Pools

If you are using user auxiliary storage pools (ASPs), you can restore libraries and objects that were contained in user ASPs by using the Restore Library (RSTLIB) or Restore Object (RSTOBJ) command. Libraries and objects are automatically restored to the same user ASP from which they were originally saved, but you can place them into either the system ASP or into another user ASP through the RSTASP parameter on the appropriate restore command. For more information on ASPs, refer to "General Information about Auxiliary Storage Pools" on page 14-7.

Restoring Security Information

The following topics discuss restoring security information, such as user profiles, authorization lists, authority holders, distribution objects, and authority to objects. For more information about user profiles, authorization lists, and authority holders, see the manual *Security Concepts and Planning*.

Restoring User Profiles

You can restore a single user profile or a list of user profiles with the RSTUSRPRF command. Before using the RSTUSRPRF command, you must end all subsystems (using the ENDSBS command). The RSTUSRPRF command also allows you to move a user from one AS/400 system to another AS/400 system, to restore interactive information, and to recover a damaged user profile more easily. AS/400 functions use interactive user profile information to keep specified values on a display.

The RSTUSRPRF command restores all user profile attributes originally defined on the Create User Profile (CRTUSRPRF) command. When you restore all user profiles, the RSTUSRPRF command also restores authorization lists, authority holders, and mail (if saved). However, it does not restore private authorities for objects owned by other users. The user profiles, libraries, and objects owned by the profiles must be restored first.

If you use the RSTUSRPRF command more than once during a restore operation, each use of the command resets the profile attributes to their status at the time of the save operation. Each use of the RSTUSRPRF command is followed by the RSTLIB, RSTOBJ, and RSTDLO commands to restore all objects owned by the profiles. Then, use the RSTAUT command to restore authority.

If you are restoring user profiles that were saved with *ALLOBJ special authority and the security level is 30 or 40, the authority will be removed when the user profiles are restored. After the restore operation, use the Change User Profile (CHGUSRPRF) command to grant the authority back to the user profiles that require *ALLOBJ special authority.

When you use the RSTUSRPRF command to restore *all* user profiles, the following conditions occur:

- If a user profile exists on the system, but is not on the media, the profile remains on the system.
- If a user profile exists on the media, but is not on the system, a new user profile is created on the system.
- If a user profile exists on both the media and the system, the user profile from the media replaces the user profile on the system.

  Private authorities for objects owned by other users are not restored. Any such authorities already contained in the user profile on the system remain unchanged.

When you use the RSTUSRPRF command to restore *individual* profiles, consider the following conditions:

- When moving user profiles from one system to another, you must group the objects owned by the user profiles using the save command (SAVLIB or SAVOBJ).

- User profiles, authorization lists, and authority holders are saved using the SAVSYS or the SAVSECDTA command.
- If the user profile on the system is damaged, the last SAVSYS or SAVSECDTA medium can be used for the restore operation to recover the user profile and its authorities.

  Note: If an IBM-supplied user profile is damaged, it cannot be deleted. You must restore the operating system again to restore the IBM-supplied user profile.
- Authorization lists, authority holders, and distribution objects are restored only when RSTUSRPRF USRPRF(*ALL) is specified.
- To transfer the user profiles and their authorities to another system, you must do the following steps:
  1. Restore the user profiles using RSTUSRPRF USRPRF(*ALL) ALWOBJDIF(*ALL).
  2. Restore the needed objects using the RSTLIB, RSTOBJ, or RSTDLO commands specifying ALWOBJDIF(*ALL).
  3. Restore the authorities of the user profiles using the RSTAUT command.

Table 6-2 shows what occurs when you restore all user profiles or individual user profiles from the SAVSYS or SAVSECDTA media. This situation does not apply when migrating from a System/38. See the *Migrating from System/38 Planning Guide* for more information.

Table 6-2 (Page 1 of 2). Results of Restoring User Profiles

User Profile Attributes	**Restore *ALL User Profiles**	**Restoring Individual User Profiles That Existed on the System before Using RSTUSRPRF Command**	**Restoring Individual User Profiles That Do Not Exist on the System before Using RSTUSRPRF Command**
Associated group profile (GRPPRF owner and GRPAUT values)	Values are restored from SAVSYS or SAVSECDTA media.	The values remain as they currently are on the system.	The values restored are GRPPRF(*NONE), OWNER(*USRPRF), and GRPAUT(*NONE).
Sign-on and document passwords	Values are restored from the SAVSYS or SAVSECDTA media.	Password values remain as they are on the system.	The sign-on and document password are *NONE.
Special authorities (*SAVSYS, *JOBCTL, *SECADM, *SERVICE, *SPLCTL, and *ALLOBJ[1])	Values are restored from the SAVSYS or SAVSECDTA media.	Values are restored from the SAVSYS or SAVSECDTA media.	Values are restored from the SAVSYS or SAVSECDTA media.
Private authorities	Authorities from the SAVSYS and the SAVSECDTA media are added to the authorities currently on the system when the RSTAUT command is issued.	Authorities from the SAVSYS and the SAVSECDTA media are added to the authorities on the system when the RSTAUT command is run.	Authorities are restored from the SAVSYS or SAVSECDTA media when the RSTAUT command is issued.
Password change date	The date the password was last changed is taken from the authorized users table[4] on media.	The date on the system is used for the date the password was last changed.	The current date is used.
Basic information[2]	Values are restored from the SAVSYS or SAVSECDTA media.	Values are restored from the SAVSYS or SAVSECDTA media.	Values are restored from the SAVSYS or SAVSECDTA media.
Interactive profiles[3]	Interactive profiles on the SAVSYS or SAVSECDTA media replace the profiles on the system.	Interactive profiles on the SAVSYS or SAVSECDTA media replace the profiles on the system.	Interactive profiles are restored from the SAVSYS or SAVSECDTA media.

Table 6-2 (Page 2 of 2). Results of Restoring User Profiles

User Profile Attributes	Restore *ALL User Profiles	Restoring Individual User Profiles That Existed on the System before Using RSTUSRPRF Command	Restoring Individual User Profiles That Do Not Exist on the System before Using RSTUSRPRF Command
Owner of user profile	If the profile being restored is currently on the system, ownership remains the same. If the profile being restored is not currently on the system, ownership is from the SAVSYS or SAVSECDTA media unless the owning profile does not exist on the system; in that case, the QDFTOWN user profile becomes the owner. If ownership on the media is different than the owner on the system and ALWOBJDIF(*ALL) was specified, then the owner on the system remains the owner.	Ownership remains the same.	Ownership is from the SAVSYS or SAVSECDTA media unless the owning profile does not exist on the system; if that occurs, then the QDFTOWN user profile becomes the owner.
Distribution objects	Distribution objects are restored from the SAVSYS or SAVSECDTA media.	Not allowed.	Not allowed.

1 *ALLOBJ special authority is removed from a user profile on a system at security level 30 or 40, except QSECOFR, QSYS, QLPAUTO, and QLPINSTALL.

2 Basic information, such as the initial program, accounting code, job description, output queue, message queue, maximum storage, priority limit, and text.

3 The interactive profile contains information that is used by certain functions such as IDDU.

4 The authorized user table is an object that contains an entry for each user profile on the system.

If the user profile is on the system, message CPF3714 indicates that the restore operation did not change the password or associated group profile information (owner and group authority values). The password and group information are left as they were. If the user profile is not on the system, message CPF3713 indicates that the associated group profile is not restored, and that the password is now *NONE.

Restoring Object Authorities

When you restore objects and profiles, you do not simultaneously restore the authority to those objects. Private authority is restored only using the Restore Authority (RSTAUT) command. This command cannot be used until the user profiles and some user libraries are restored to the system. Before using this command, you must end all subsystems.

The Restore User Profile (RSTUSRPRF) command restores authorization lists and resets the profile attributes to their status at the time of the save operation, but does not restore any other private authorities. The RSTUSRPRF command builds an authorization table during the restore operation. This table contains information about the association of objects and authorities to the user profiles being restored. The RSTAUT command grants private authorities in addition to any authorities granted since the save operation. Because the RSTAUT function uses and then deletes the authorization table built by the RSTUSRPRF command, the RSTAUT command can be used only once for each restored user profile.

You can use the RSTAUT command to restore private authorities for individual user profiles with the USRPRF parameter. If USRPRF(*ALL) is specified, authorities are restored from each available user profile authorization table.

The following examples illustrate possible security considerations:

- When you restore private authorities for an object, you have the same authorities for that object that you had when the system was saved including authorities granted after the save operation.

  For example, assume Tom creates an object and authorizes Bob to use that object. After saving the object authorities through the SAVSYS or the SAVSECDTA command, Tom revokes Bob's authority. When the RSTUSRPRF and RSTAUT commands are issued, Bob has all the original authorities that he had when the SAVSYS or SAVSECDTA command was run. Tom can revoke the original authority using the RVKOBJAUT command.

  Assume Tom gives Joe authority to use an object after using the SAVSYS or SAVSECDTA command. If Joe's profile exists on the system and is not damaged at the time the RSTUSRPRF command is run, Joe will have authority to the object regardless of whether the RSTAUT command is run.

- If you give a user authority to an object, enter the SAVSYS or SAVSECDTA command, and then restore the user's authority to the object at a later date, the user is granted the original authority to the object, even though the contents may have changed.

  For example, you can give Ann private authorities to program ABC/XYZ, save the system (SAVSYS) or save security data (SAVSECDTA), then delete that program. If a new program called ABC/XYZ is created, and the security officer restores Ann's authority (RSTAUT), Ann has the original authority to the new program ABC/XYZ. See "Security Considerations When Saving and Restoring" on page 9-12 for more information.

- Ownership for an object is established when the object is restored (RSTOBJ, RSTLIB or RSTDLO commands), and is not changed by the RSTAUT command. If you are the owner of an object and save the object from System A and restore your user profile and the object on System B, then you have the same authorities on System B as you had on System A.

- If the owner of the object on the system does not match the owner of the object on the save media and the allow object differences (ALWOBJDIF(*ALL)) parameter was specified on the restore command, then the object is restored. However, the user on the system becomes the owner of the object.

- If the owner of an object being restored does not exist on the system at the time of the restore operation, then the system restored the object with the default owner (QDFTOWN) user profile as the owner.

Limiting the Restore of Programs

When the system security level is 40, the restore of programs that have been changed or that contain restricted instructions is controlled by the allow object differences (ALWOBJDIF) parameter on the Restore commands.

To detect changes to a program (rather than through a normal system interface), a validation value is calculated when the program is created. When the program is restored, the validation value is calculated again and compared to the value that was calculated when the program was created.

- If the program was created on a Version 1 Release 2 or earlier system, no validation value is saved with it. The QSECURITY system value and the ALWOBJDIF parameter on the restore command is checked as follows:

  - For security levels 10 through 30, or when ALWOBJDIF(*ALL) is specified, the program is restored with no further validation or notification given.

  - For security level 40, when ALWOBJDIF(*NONE) (the default) is specified, an attempt is made to translate the program again. If the translation succeeds, the copy of the retranslated program is restored. No *AUTFAIL type journal entry is written in the QAUDJRN journal.

    If the translation fails, the original copy of the program is restored and all public and private authorities are revoked. Ownership of the program is transferred to the QDFTOWN user profile. A *AUTFAIL type journal entry is written to the QAUDJRN journal to indicate the changes. Message CPF375B is sent to the job log. The security officer must decide whether to accept the program (change ownership, grant authority, or reject it).

The translation fails if the observability of the program has been removed.

For more information on setting up and managing the QAUDJRN journal, see the *Security Concepts and Planning* manual.

- If the program has a correct validation value saved with it, the original program is restored. The QSECURITY system value and the ALWOBJDIF parameter on the Restore commands do not apply. No *AUTFAIL type journal entry is written in the QAUDJRN journal.
- If the program has a validation value that fails, an attempt is made to retranslate the program.
  - If the translation succeeds, the program is restored. A *AUTFAIL type journal entry is written to the QAUDJRN journal to indicate what happened. Message CPF375C is sent to the job log.
  - If the translation fails, the original copy of the program is restored from the media. The security and logging actions taken depend on the QSECURITY system value and the ALWOBJDIF parameter on the Restore command.
    - For security levels 30 and below, a *AUTFAIL type journal entry is written to the QAUDJRN journal to indicate that a program was restored that may result in a possible violation of security. Message CPF375A is sent to the job log.
    - For security level 40, when ALWOBJDIF(*ALL) is specified on the Restore command, a *AUTFAIL type journal entry is written to the QAUDJRN journal to indicate that a program was restored (as requested by a user with *ALLOBJ special authority) that may result in a possible violation of security. Message CPF375D is sent to the job log.
    - For security level 40, when ALWOBJDIF(*NONE) is specified on the Restore command, all public and private authorities are revoked and ownership of the program is transferred to the QDFTOWN user profile. A *AUTFAIL type journal entry is written to the QAUDJRN journal to indicate the changes. Message CPF375B is sent to the job log.

Table 6-3 shows what happens at each security level.

Table 6-3. Restore of Programs That Use Interfaces That Are Not Supported

Validation Value	Successful Translation	Level 40 or ALWOBJDIF (*NONE)	Level 40 ALWOBJDIF (*ALL)	Levels 30 and Below
None	Yes	1	N/A	N/A
None	No	2	N/A	N/A
None	Not attempted	N/A	3	3
Valid	Not attempted	4	4	4
Not valid	Yes	5	5	5
Not valid	No	2	6	6

Notes:

1 Restore a copy of the translated program. No notification is given.

2 Restore the original program, revoke all public and private authorities, unlink any authorization lists, and change the owner to the QDFTOWN user profile. Write the journal entry to the QAUDJRN journal and send a message to the job log.

3 Restore the original program. No notification is given.

4 Restore the translated program. No notification given.

5 Restore the translated program. Write the journal entry to the QAUDJRN journal and send a message to the job log.

6 Restore the original program. Write the journal entry to the QAUDJRN journal and send a message to the job log.

Restoring the System in Parts

You do not have to restore all user libraries before restoring authority and beginning operations. When you have installed the OS/400 licensed program, you can restore the remainder of your system in parts.

You can restore all user profiles (or a selected subset with the RSTUSRPRF command) or restore only critical user libraries, and then restore object authorities (RSTAUT command). Later, you can restore less critical libraries, but you must enter the RSTUSRPRF, RSTLIB, and RSTAUT commands again (in that order), specifying the needed user profiles.

The RSTUSRPRF command builds an internal table of authorities that were in existence when the user profiles were saved. The RSTAUT command uses this table to grant authorities and then deletes the table.

Notes:

1. The subsequent RSTUSRPRF command resets the user profile attributes to their status at the time of the save.
2. The subsequent RSTAUT command grants the private authorities saved at the time of the original save, but does not revoke any private authorities granted since then.

Restoring Related Objects in the Correct Order

Certain related objects (see the following list) must be restored in a specific order. When the related objects are in the same library, the *system* restores the objects in the correct order. When the related objects are in different libraries, *you* restore them in the correct order, as follows:

- Journal objects must be restored in the sequence that they were journaled, and then the files being journaled must be restored.

  If you save your libraries using SAVLIB LIB(*NONSYS or *ALLUSR), the system cannot determine if your journals are restored before your files. Also, if the based-on files are not in the same library as the logical files, the system cannot determine if the based-on files are restored before the logical files. SAVLIB LIB(*NONSYS or *ALLUSR) saves the libraries in alphabetic sequence while RSTLIB SAVLIB(*NONSYS or *ALLUSR) restores the libraries in the order they were saved. You can name your journal library in such a way that it is restored before the database files (for example, #JOURLIB).

  For more information, see Chapter 12, "Database Recovery."
- Logical files cannot be restored until the physical files or logical files on which they are dependent are restored.
- Mail distribution objects must be restored using the RSTUSRPRF command before the associated document can be restored using the RSTDLO command.

When a file is restored that causes an access path to be rebuilt, rebuilding occurs asynchronously from the restore operation and can finish later (after you have finished restoring files).

Interrupted Operation

The restore operation can damage an object so that the object and its data may not be restored properly. For instance, if you cancel the restore operation of a member before the entire member is replaced, the member contains some old and some restored data, and is considered logically damaged. You should delete such an object. For each damaged object a message is sent to the history log and to the user who first encounters the damage.

If a restore operation on a database file is interrupted by a system or job end, an escape message is sent. Recovery for that file is started when you attempt to restore the file again, save it, or do an IPL. The system will do restore recovery when any database recovery-sensitive function is done. At that point, informational messages are sent to the history log to indicate if recovery was successful or unsuccessful. Recovery does not mean that the restore operation is completed. Instead, it means that the file is recovered so other operations on it can succeed.

Considerations for Restoring from a Distribution Tape

If you are installing a new release or restoring the system from a distribution tape, any object that you have changed is changed back to the IBM-supplied defaults when it is restored from the distribution tape. The objects you previously changed have to be changed again after the restore operation is completed.

Chapter 7. Saving and Restoring Documents, Folders, and Mail

Documents are objects created by users of an office product such as the word processing function of OfficeVision/400 or the shared folder function of PC Support/400. **Folders** are named areas on a disk that are directories to documents and other folders. The Save Document Library Objects (SAVDLO) command saves documents and folders.

The objects saved during a complete save operation of office services data include office database files, the office services journal (QAOSDIAJRN), journal receivers, distribution objects, distribution documents, and filed documents and folders. Figure 7-1 illustrates a complete save operation.

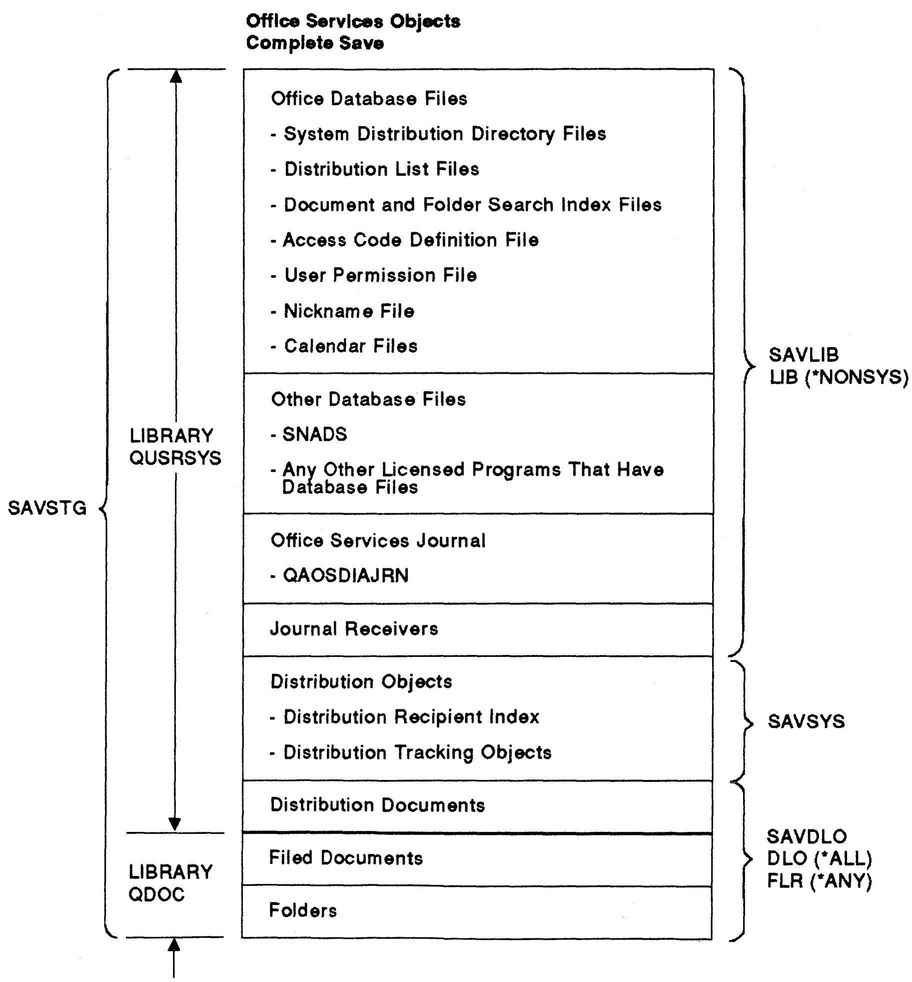


Figure 7-1. Complete Save of Office Services Objects

Saving Documents and Folders

When a document is filed or created by an office product, it is stored in library QDOC as an object with the type *DOC. Documents can be saved individually or in a group using the SAVDLO command. You must have either save system (*SAVSYS) or all object (*ALLOBJ) special authority to use SAVDLO OBJ(*ALL) FLR(*ANY), or SAVDLO DLO(*MAIL).

Parameters for the SAVDLO command allow you to:

- Name specific documents to save by user-defined name or by system object name.
- Specify search values using DLO(*SEARCH) so that the system can identify which documents to save as follows:
  - Folder
  - Marked for offline storage
  - Document end date
  - Creation date
  - Document class
  - Owner
  - Document last changed date and time
- Specify saving all filed documents and all documents referenced by a distribution by specifying DLO(*MAIL).

You can use the search type (SRCHTYPE) parameter to specify that you are searching for only documents, or you can search for both folders and documents. If you specify SRCHTYPE(*DOC), then you can specify any of the search parameters already supported from the previous release for documents:

- FLR
- CHKFORMRK
- CHKEXP
- CRTDATE
- DOCCLS
- REFCHGDATE
- REFCHGTIME

SRCHTYPE(*DOC) is the default. If you specify SRCHTYPE(*ALL), then all the folders (including all the folders and documents within them) that meet the specified search parameters are saved. Search values that are supported for SRCHTYPE(*ALL) are:

- CRTDATE
- OWNER
- REFCHGDATE
- REFCHGTIME

REFCHGDATE and REFCHGTIME parameters are valid for only documents but are allowed when SRCHTYPE(*ALL) is specified in one step for daily saves of all new folders and documents and all new or changed documents.

Limitations When Saving Document Library Objects

You cannot save more than 32 000 objects using a save file. If you attempt to save more than 32 000 to a save file using the SAVDLO command, an error message is sent and no objects are saved.

If you attempt to use the SAVDLO command for 32 000 or more objects using diskette or tape, the save operation is done in multiple files and the labels for those files are set to the system-created defaults.

The total number of objects saved refers to the total number of space objects, not the total number of document and folder objects. A document or folder object can contain up to 256 space objects of 16 megabytes per space object. For example, if 31 999 documents are saved in a save file where 31 990 documents occupy one space object each and one document occupies 9 space objects, then that save file is full with 31 999 objects.

Note: If you plan to save more than 31 999 objects with one SAVDLO command entry, it is recommended that you use tape as your save media.

There are different methods you can use for planning to save documents. If you do not plan to save many documents to diskette or tape, you could save individual documents or a selected group of documents when necessary. For example, if you mark all documents to be saved on diskette or tape or in save files, you could use the SAVDLO command once a week (or once a day or month) to save all marked documents. Or you could use the SAVDLO command once a week (or once a day or month) to save all documents you have finished working with.

When you use the SAVDLO command to save filed and unfiled documents and folders to a save file, the save file can be saved to diskette or tape using the Save Save File Data (SAVSAVFDTA) command. The documents and folders can then be restored using the RSTDLO command directly.

Note: The SAVDLO command saves filed and unfiled documents (distribution documents). It does not save the distribution objects. Distribution documents should not be confused with distribution objects, which are saved with the Save System (SAVSYS) or the Save Security Data (SAVSECDTA) command.

Marking Documents for Storage

You should establish policies regarding saving, deleting, or freeing documents to free disk space. As a regular practice, you should save the following types of documents, or folders containing the following types of documents:

- Documents only needed occasionally
- Lengthy documents
- Confidential documents

You can mark a document for offline storage before you save a document on diskette, tape, or in a save file. You are not required to mark a document for storage before you save it. You can mark the document so that:

- The document text and document details are deleted when the document is saved.
- The document text is deleted to free disk space, but the document details remain in the search index database.
- The document remains the same after it is saved.

When the document details remain in the search index database, the document name is shown when you list the names of documents in the folder. The document name can appear in a search list when you search for documents. That way an Office user can find that the document has been saved to tape or diskette. The document must be restored to the system before a user can look at or change the document contents.

After you have marked the documents you want to save, you can use the SAVDLO command with the CHKFORMRK(*YES) parameter specified to save all documents that are marked for storage. If you marked a document to be deleted, or for only the document text to be deleted, then the document or only the document text is deleted.

Even if you did not mark the documents for storage, you can specify in the SAVDLO command to delete the document from the document library, or delete the document text and keep the document details, using the STG(*DELETE) or STG(*FREE) parameter.

You can save documents that are marked for storage without specifying CHKFORMRK(*YES). For example, you can save a folder that contains some marked and some unmarked documents by specifying SAVDLO DLO(*ALL) FLR(save-folder-name) CHKFORMRK(*NO). If you also specify STG(*DELETE) or STG(*FREE), then those documents that are marked are saved, but the STG(*DELETE) or STG(*FREE) parameter is ignored for the marked documents.

Saving Distribution Objects and Documents (Mail)

The following figure shows the objects associated with the mail and how they are saved:

MAIL

Distribution Objects:
- Distribution Recipient Index
- Distribution Tracking Object
Saved Using SAVSYS or SAVSECDTA

Distribution Document:
- Document Filed in QDOC
- Unfiled Document in QUSRSYS
Saved Using SAVDLO

RV2W380-2

The following describes the objects associated with the mail.

Distribution Recipient Index: An internal object that contains entries for incoming object distributions, incoming document distributions, outgoing document distributions, and error distributions.

Distribution Tracking Object: An internal object that is used to control office distributions.

Distribution Document: An internal document that contains the document content and the document details for distributions.

Saving Distribution Objects: Two commands are used to save the mail.

The first command can be either the Save System (SAVSYS) or Save Security Data (SAVSECDTA) command. Running either of these commands saves all the distribution objects currently on the system, along with all the other objects that are saved by these commands. You need to run only one of these commands to save the distribution objects. If you use the SAVSECDTA command, you must specify the parameter MAIL(*YES). These commands do not save any documents. To save documents you must use the SAVDLO command.

Saving Distribution Documents: The second command to run is the Save Document Library Object (SAVDLO) command.

The Save Security Data (SAVSECDTA) command and the SAVDLO DLO(*MAIL) commands do not require OfficeVision/400 activity to stop or the system to be in a restricted state. However, a limited amount of mail use between the time the SAVSYS or the SAVSECDTA command is run and the time the SAVDLO command is run helps ensure that the distribution objects match the saved documents.

There are two ways that you can use the SAVDLO command to save mail.

The first is by specifying SAVDLO DLO(*ALL) FLR(*ANY). This command will save all the folders (including all documents and folders within them), all documents filed in library QDOC (whether they are in a folder or not), and all distribution documents. All documents that are part of mail are saved.

Note: To save distribution documents, you must not specify the target release parameter TGTRLS(*PRV) with the SAVDLO command. If you do specify TGTRLS(*PRV), then all the folders (including all documents and folders within them) in addition to all documents filed in library QDOC are saved. Distribution documents are not saved.

The second way that you can use the SAVDLO command to save documents that are referenced by distribution objects is to specify SAVDLO DLO(*MAIL). This command saves only those documents filed in library QDOC that are referenced by a distribution object and all distribution documents. SAVDLO DLO(*MAIL) cannot be specified with TGTRLS(*PRV).

Notes:

1. To maintain the integrity of the mail, you should always run the SAVDLO command as soon as possible after saving the distribution objects using the SAVSYS or the SAVSECDTA command.
2. If you run the SAVSYS command, and one or more of the following messages appear in the job log, then any mail that exists for the user profile names specified in the message CPF9006 is not saved.
   a. CPF9006 message: User profile or user identification is not enrolled in the system distribution directory.
   b. Followed by a CPF3772 message: `SAVSYS incomplete. One or more objects not saved.`

   Note: The SAVSYS tape is valid and may be used to restore other objects.

   If you want the mail for the users named in the CPF9006 messages to be saved, you must enroll those users in Office again and then run the SAVSYS or SAVSECDTA MAIL(*YES) command again. Enrolling the users specified in the CPF9006 messages before you run the SAVSYS or SAVSECDTA MAIL(*YES) command again, eliminates these messages.

Saving SAVDLO Output to an Outfile

Information about the SAVDLO operation can be printed or written to a database file. To write the information to a database file, specify OUTPUT(*OUTFILE) OUTFILE(library/file-name) OUTMBR(member-name). The file specified must have the same format as the file QSYS/QAOJSAVO.OJSDLO. The *Office Services Concepts and Programmer's Guide* has a description of the format of this file.

You can add records to this file for each save operation that you do. Then, if you want to write an application program that retrieves information from this file, you can do so. If you use the OUTPUT(*OUTFILE) parameter, you should periodically review the file that has output records being written to it so it does not become too large. If it gets very large, you should consider saving and then clearing it. You do not need to create the file in advance. The SAVDLO operation does this for you.

Printed Output for Distribution Documents

If you specify OUTPUT(*PRINT) along with the SAVDLO command, the printed output for any folders or filed documents that are saved appears as follows:

```
                    Document Library Archive (SAVDLO)

Volume ID.................:
File label ID.............: QDOC                 Sequence.........: 0001
Save date/time............: 05/03/89  11:45:02   Device...........: TAP01
Folder......:

Document    Description         Date        Owner       Object
Name                            Created                 Name

TEST01      Project test doc    03/07/89    BANKS       CDJQ325092
TEST02      Search test         03/07/89    BANKS       CDJQ525064
```

The printed output for any unfiled documents that are saved appears as follows:

```
                    Document Library Archive (SAVDLO)

Volume ID.................:
File label ID.............: QDOC                 Sequence.........: 0001
Save date/time............: 05/03/89  11:45:02   Device...........: TAP01
Folder......:

Sender            Distribution Name                          Date
                                                             Received
  USER1.AS400001  Project test document sent to USER2        05/03/89
```

Note: The heading information on the preceding displays is device-dependent. All information does not appear for all devices.

Saving from Document Lists

The command SAVDLO(*DOCL) DOCL(document-list-name) FLR(folder-name) saves all the documents referenced by a document list. The document list itself is not saved (it can be saved separately). The document list must be in a folder. If any folders are referenced by the document list, they are not saved.

IBM-Supplied Journals Used for Documents or Folders

When a document is filed or a folder is placed in the document library, details assigned by the user are placed in database files in library QUSRSYS that are separate from the document or folder. Every time a user changes the document or folder details, the database files are updated. To manage database files, the system uses journaling, which is the primary means of recovering from errors. The journal receiver can grow quite large. Therefore, you should periodically detach and save a full journal receiver, and attach a new one. For more information, see "Managing System-Created Journals and Journal Receivers" on page 12-10.

Files That Are Journaled for Directories and Search Index Database Files

Some files used by Office are automatically journaled by the system. These files, found in library QUSRSYS, contain information about directories and search index database files. The following is a list of these files:

QAOSAH05	QAOSSH11
QAOSAH07	QAOSSH12
QAOKDYX4	QAOSSH13
QAOKDY01	QAOSSH14
QAOKDY04	QAOSSH15
QAOKDY05	QAOSSH17
QAOKDY07	QAOSSH18
QAOKDY08	QAOSSH19
QAOSSH10	QAOKDY09
QAOKLY02	QAOKLY03
QAOKNY06	

The Restore Library (RSTLIB) and Restore Object (RSTOBJ) commands prevent the restore of search index database files (QAOSSH11 through QAOSSH19) to a system where they already exist. If one of these files becomes damaged or deleted, you must run RCLDLO DLO(*ALL) command after restoring the files from your most recent save tapes.

Restoring Document Library Objects

To restore documents or folders, use the Restore Document Library Object (RSTDLO) command. Each time you use the RSTDLO command, all documents and folders specified in the command are restored from the same save file or from the same file on diskette or tape.

This command allows you to specify the documents and folders you want to restore, in several ways. You will be able to use the RSTDLO command most efficiently and effectively if you are aware of which media or save files contain the documents and folders you want to restore. If printouts from the SAVDLO command are available, you should review these, or use the Display Tape (DSPTAP), Display Diskette (DSPDKT), or Display Save File (DSPSAVF) command if no lists are available to determine which files contain the documents and folders you want to restore.

Documents and folders are stored in library QDOC as an object of type *DOC and *FLR, respectively. Only documents and folders are stored in this library. After documents have been filed and saved, restore them using the Restore Document Library Object (RSTDLO) command.

To restore documents and folders to a folder, you must at least have *CHANGE authority to the folder to which the documents and folders are being restored. You must also have *SAVSYS or *ALLOBJ special authority, or *ALL authority to each document and folder being replaced.

No search values exist for folders. You can restore individual documents or folders within the saved folders.

The name of the document to be restored is specified by the DLO parameter. Folder names are not allowed. Documents that were saved by a user-specified name can be restored by either the user-specified name or the system object name. You can enter up to 300 names on one RSTDLO command. Specifying DLO(*ALL) restores all the folders and documents that match the values specified by all the remaining parameters on this command. To restore a folder object, the entire folder (the folder object plus all document and folder objects within it) must also be restored. However, if the specific folder being restored was saved in other folders at the time it was saved, those higher-level folders do not have to be restored to restore the specific folder.

The RSTDLO command supports RSTDLO DLO(*ALL) SAVFLR(*ANY) to restore all the document and folder objects that are on the tape volume currently placed in the tape unit for the sequence numbers you have specified.

Note: To restore everything from all the files on a tape, you will also need to enter beginning and ending sequence numbers on the SEQNBR parameter.

Documents and folders are filed again when you attempt to restore documents and folders that are unknown to the system, either because they have been deleted, they are being restored on a different system than they were saved on, or they are being restored with the NEWOBJ(*NEW) parameter. For documents, the search index database is updated with the restored detail information for each restored document. For folders, the search index database is updated only if the folder is new.

The following are ways you might restore folders and documents:

- Restore 1 to 300 documents from the same media file by specifying the names of the documents. You have the option of renaming them, restoring them into a different folder, and creating new system object names.
- Restore 1 to 300 folders from the same media file, and optionally create new system object names for each document and folder object restored.
- Restore all filed documents that are not in any folder on the save media and optionally create new system object names for each document restored.
- Restore 1 to 300 documents by system object name, and optionally create new system object names for each document restored.

There are some restrictions on the results you will obtain from the RSTDLO command for certain combinations of parameters, depending on what the input was for the SAVDLO command. To help you plan your restore process, those considerations are discussed in the following topics.

Restoring Document Library Objects from Diskette or Tape Files: When restoring objects from diskette or tape files that resulted from a save operation of 32 000 or more objects, the file label is reset to LABEL(*GEN) while running the SAVDLO command. To restore from a file where the label was reset in this manner, you must specify LABEL(*GEN).

If you are using the RSTDLO command across multiple files with labels in the form of QDOC.XXX (which can be done specifying a sequence number range), you do not need to specify the label of each media file in the LABEL parameter. Specifying LABEL(*GEN) restores each file.

Restoring into an Existing Document Library Object: If you try to restore a document library object that already exists on the system and it is in use or you are not authorized to it, it is ignored and the restore operation continues with the next object. If you try to restore a folder that already exists on the system, and the folder is in use or you are not authorized to it, the entire folder (and all documents and folders in it) is not restored.

Restoring Large Folders

Be careful when restoring folders of 32 000 or more objects, or smaller folders that were in folders of 32 000 or more objects when they were saved. If you specify LABEL(*GEN) but enter a sequence number range or volume IDs that are not enough to get the entire folder, some of your folders may not be restored. For example, assume you save a folder with 70 000 documents and folder objects in it to tape, and save it in files with labels QDOC.001, QDOC.002, and QDOC.003 and sequence numbers 1, 2, and 3. If you try to restore it with LABEL(*GEN) SEQNBR(1 2) specified, the objects in the file with sequence number 3 and label QDOC.003 are not restored, but a partial folder is restored. A message is displayed to alert you that all objects may not be restored when this situation occurs.

Restoring Lists of Documents or Folders

When restoring lists, be sure that all the documents and folders in the list are in the same file on the media. The restore operation is done from the first file on the media that contains all the items in the list. If the entire list does not exist in any file, then nothing is restored.

Restoring Distribution Objects and Documents (Mail)

The following figure shows objects associated with the mail and how they are restored.

MAIL

Distribution Objects:

- Distribution Recipient Index
- Distribution Tracking Object

Restore Using RSTUSRPRF

Distribution Document:

- Document Filed in QDOC
- Unfiled Document in QUSRSYS

Restore Using RSTDLO

RV2W394-2

Note: You cannot restore distribution documents until you have restored library QUSRSYS.

Considerations for Restoring Mail: Consider the following when restoring distribution objects.

- After you run the Restore User Profile (RSTUSRPRF) command to restore the distribution objects, you must run the Restore Document Library Objects (RSTDLO) command to restore the documents that are referenced by the distribution objects.
- Use of the mail logs should be kept to a minimum between the time of running the RSTUSRPRF command and the RSTDLO command. The distribution objects restored by the RSTUSRPRF command have entries in the user's mail logs, but the documents

are not restored until after the RSTDLO command is run. If the entries on a mail log that are restored by the RSTUSRPRF command are deleted before the RSTDLO command is run, then running RSTDLO does not put references to the restored documents back in the mail log. The RSTUSRPRF command must be run again to restore the deleted distribution objects.

- You should make sure that the save media or online save file that you use for the RSTDLO operation is the one that contains the documents that match the distribution objects just restored.

  For example, if you saved your mail using the Save Security Data (SAVSECDTA) and the Save Document Library Objects (SAVDLO) commands on three consecutive Fridays, for example, May 5th, May 12th, and May 19th, and a month later you want to restore the mail from May 19th, then you should make sure that you use both the RSTUSRPRF and RSTDLO commands to restore only information from the save media created on May 19th. You should not use the RSTUSRPRF command to restore distribution objects from the save operation on May 5th or May 12th, and then restore documents saved on the May 19th SAVDLO media. Such a mismatch may cause unpredictable results and many documents may not be restored.

- Anytime a RSTUSRPRF command is run, a RSTAUT command must be run after RSTDLO or RSTDLO DLO(*MAIL).

Restoring Distribution Objects: There are two commands that must be used for restoring mail.

The first command is the RSTUSRPRF command. It is used to restore the distribution objects that were saved using either the SAVSYS or SAVSECDTA command. You cannot restore the distribution objects that were saved for one user profile or a list of user profiles. You must specify RSTUSRPRF USRPRF(*ALL) MAIL(*YES) to restore distribution objects.

Restoring Distribution Documents: The second command is the RSTDLO command. The easiest way to restore all the mail from the save media or online save file is to specify RSTDLO DLO(*MAIL), or if SAVDLO DLO(*MAIL) is used to create the save media or online save file, you can specify RSTDLO DLO(*ALL) SAVFLR(*ANY).

Specifying RSTDLO DLO(*MAIL) restores only those filed documents that have a mail log reference at the time they are saved, plus all the distribution documents from the save media or online save file. Specifying RSTDLO DLO(*ALL) SAVFLR(*ANY) restores all the filed documents, distribution documents, and folders from the save media or online save file.

Distribution documents cannot be restored individually. If you specify any other form of the RSTDLO command, such as RSTDLO DLO(*ALL) SAVFLR(A) and RSTDLO DLO(X) SAVFLR(A/B), then no distribution documents are restored. If the filed documents that are restored using these other forms of the RSTDLO command contain mail log references, then the mail log references are restored if the distribution objects exist on the system.

Mail log references are updated for all existing local recipients of a restored document. Mail log references on remote systems for remote recipients are not restored. If a document being restored still exists in a mail log at the time it is restored, then the contents of the document are restored and the status of the document in the mail log is not changed. If the document being restored has been deleted from a mail log, then the status of the restored document is either *filed* for a filed document or *opened* for a distribution document.

Mail log references are restored for a local sender of a document if there was an entry in the sender's mail log at the time the distributions were saved (at the time the SAVSYS or SAVSECDTA command was run). Entries in the mail logs of remote senders are not saved or restored.

Printed Output for Restored Document Library Objects

Information about the RSTDLO operation can be printed.

If you specify OUTPUT(*PRINT) along with RSTDLO DLO(*ALL) SAVFLR(*ANY) or DLO(*MAIL), the printed output for any folders and filed documents that are restored appears as follows:

```
                    Document Library Archive (RSTDLO)

Volume ID.................:  *MOUNTED
File label ID.............:  *GEN
                                                Device...........:  TAP01
Beginning Sequence number ..:  *SEARCH
Folder......:  *NONE

Document     Description                       Owner      Object
Name                                                      Name
CDLS5220.54  note to users                     BURK       CDLS522054
CDLS7288.69  change in schedules               OWNEN      CDLS728869
```

The printed output for any unfiled documents that are restored appears as follows:

```
                    Document Library Archive (RSTDLO)

Volume ID.................:  XXXXX XXXXX XXXXX XXXXX XXXXX XXXXX XXXXX
File label ID.............:  XXXXXXXXXXXXXXX
Beginning Sequence number....:  XXXX    Ending Sequence Number..:  XXXXXXXX
Save date/time ........: 05/03/89  11:45:02  Device.........:  XXXXXXXX

Sender            Distribution Name                         Date
                                                            Received
 USER1.AS400001   Project test document sent to USER2       05/03/89
```

Note: The heading information on the preceding displays is device-dependent. All information does not appear for all devices.

Restoring Ownership for Document Library Objects

When a document or folder is saved, the name of the owner's user profile is saved with it. When an object is restored that does not already exist on the system, the system determines if the owner's user profile exists in the system distribution directory. If the owner exists and is enrolled as an Office user, that owner's documents and folders are restored with the same owner they had when they were saved. The person doing the restore operation must be authorized to create documents and folders for the owner.

If the owner of a document or folder being restored does not exist on the system or is not enrolled in the system distribution directory, then the document or folder is restored and the default owner (QDFTOWN) user profile becomes the owner.

If the owner of the document on the system does not match the owner of the document on the save media, and the allow object differences (ALWOBJDIF) parameter was specified as *ALL on the restore command, then the ownership of the user on the system is not changed and the document is restored. You must have all object (*ALLOBJ) special authority to specify ALWOBJDIF(*ALL).

Restoring Documents with Differences in Ownership

The allow object differences (ALWOBJDIF) parameter allows documents and folders to be restored even if the owner and system object name on the system, and the owner and system object name on the media do not match, but the user-specified names, folder path names, and system object names do match. You cannot specify NEWOBJ(*NEW) when ALWOBJDIF(*ALL) is also specified. You must have all object (*ALLOBJ) special authority to use the *ALL value with the ALWOBJDIF parameter.

Restoring Security for Documents and Folders

If a document or folder being restored does not already exist on the system at the time it is restored, or if the document is in a storage-free state on the system, then the personal status, authorization lists, and public authority are restored from the save media. If the document already exists on the system and is not in a storage-free state, or if a folder already exists on the system, then the security information remains unchanged on the system.

Restoring Damaged Document Library Objects

If you try to restore into an existing document but the document is damaged, then the restore operation fails for the damaged document. The restore operation continues and a message is sent informing you that the document is damaged and not restored.

If you try to restore into an existing folder but the folder is damaged, you receive a message informing you that the folder is damaged and not restored. The folder and all documents and folders in it are not restored.

Reclaiming Documents and Folders

When users are working with documents and folders, you may need the Reclaim Document Library Object (RCLDLO) command to recover the documents and folders when users are having problems with the documents and folders (such as internal errors) or when trying to restore into an existing document or folder that is damaged. In that case, you must reclaim the document or folder before it can be used.

In all cases, you are notified when a RCLDLO operation is necessary. For more information on this command, see the online information for the RCLDLO command.

Folder Considerations

When you use the RSTDLO command, be aware that the RENAME and RSTFLR parameters do not apply when restoring folders. When you restore a folder, the fully qualified folder path name you are restoring must exist unless you are restoring a root folder. For example, if you save folder A and then delete it, you can enter RSTDLO DLO(*ALL) SAVFLR(A) and restore folder object A plus all the documents and folders in it. However, if you want to restore folder A/B/C/D, you must create folder A, then folder B in folder A, then folder C in folder A/B, before you can restore folder D in folder C. You only have to create the folders that comprise the A/B/C path, and you do not have to create folder D in folder A/B/C before you can restore it.

Document Considerations

The default for restoring documents is to restore them into the folder they were saved from. However, when you restore documents by document name, you can also specify new names for them in the RENAME parameter as well as a new folder to place them in, using the RSTFLR parameter.

For example, RSTDLO DLO(A B C) SAVFLR(Q) RENAME(D E F) would get documents A, B, and C in folder Q from the media and put them in folder Q on the system. Document A is renamed D, B is renamed E, and C is renamed F.

RSTDLO DLO(A B C) SAVFLR(Q) RSTFLR(P) would get documents A, B, and C in folder Q from the media and put them in folder P on the system (P must already exist on the system). The document names would still be A, B, and C.

RSTDLO DLO(A B C) SAVFLR(Q) RENAME(D E F) RSTFLR(P) would get documents A, B, and C in folder Q from the media and rename them as D, E, and F in folder P (P must already exist on the system).

If a duplicate name or a document that cannot be restored is found in the name list you specified on the DLO parameter, and new names are specified in the RENAME parameter, then the RENAME that corresponds to the duplicate or document that cannot be restored is ignored. For example, RSTDLO DLO(A B A C) SAVFLR(Q) RENAME(D E F G) gets documents A, B, and C in folder Q from the media and puts them in folder Q on the system. Document A is renamed as D, document B is renamed as E, the second document A is not renamed to F, and document C is renamed as G.

If you enter fewer names in the RENAME list than in the DLO name list, the RENAME entries are used until the end of the RENAME list is reached, and the remaining documents are restored without new names. If you enter more names in the RENAME list than in the DLO name list, the extra names are ignored. There is always a one-to-one relationship between the names in the DLO name list and the names in the RENAME name list. You cannot use RENAME or RSTFLR when you are restoring documents by system object name.

Using the RSTDLO Command Efficiently

To use the RSTDLO command efficiently, consider the following guidelines:

- If you are restoring many objects, consider getting *SAVSYS authority, especially if you plan on doing large restore operations frequently.
- Producing a printout takes time. Use OUTPUT(*PRINT) if you need it, but do not print needlessly.

- If you are restoring from a file in the middle or end of a tape with many files on it, check the contents of the tape by examining the output from either the SAVDLO printout or the Display Tape (DSPTAP) command before restoring. Then enter the sequence number(s) you need instead of leaving the *SEARCH default.
- Do not run large restore operations (thousands of documents or folders) when the system is at peak capacity, or when there is a lot of activity in a folder that you are restoring many documents or folders into.

Restrictions for Running the SAVDLO, RSTDLO, and RCLDLO Commands at the Same Time: No two of the following commands may be run on one system at the same time: SAVDLO, RSTDLO, RCLDLO DLO(*ALL), and RCLDLO DLO(*INT). The SAVDLO and RSTDLO commands cannot be run at the same time when running other SAVDLO and RSTDLO commands, even if different devices are used. An attempt to run these commands at the same time results in the message CPF8A47:Internal objects are in use. The second set of SAVDLO, RSTDLO, or RCLDLO operation ends with no objects saved, restored, or reclaimed.

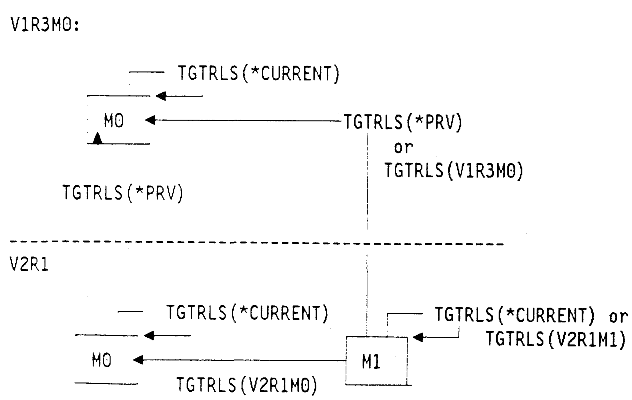
Chapter 8. Release to Release Support

Current Release to Previous Release Support

This support enables objects (such as programs and files) that are created and saved on the **current release** (such as Version 2 Release 1.1) to be restored and used on the **previous release** (such as Version 1 Release 3 Modification 0). Object compatibility is provided for many languages, and most object types are supported on both release levels as long as the objects use only previous release function.

This support is enabled when you create or save an object with the intent of restoring and running it on the previous release by the using **target release (TGTRLS)** parameter on the create or save command.

Figure 8-1 illustrates the TGTRLS parameter and values available for the current and previous releases.



V1R3 = Version 1 Release 3
V2R1 = Version 2 Release 1

M0 = Modification 0
M1 = Modification 1

Figure 8-1. TGLRLS Parameter and Values Available

This support is extremely useful to:

- A network enterprise with a central site development system on the current release and with remote sites still on the previous release.
- An application development business with a single system on the current release that supports customers who may still be on the previous release.

This support provides great savings and productivity improvements to application developers. Most network enterprises and application development businesses no longer need to maintain two development systems (a previous release system that contains previous release objects, and a current release system that contains current release objects). In most cases, this support enables previous and current release objects to exist on one development system.

This support is enabled by the **target release (TGTRLS)** parameter. To create or save an object with the intent of restoring and running it on the previous release, use the TGTRLS parameter.

The following sections describe how to create and save objects on the current release, and how to restore and use them on the previous release.

Installing the Previous Release Compiler on the Current Release

To create program objects (*PGM) that are targeted to run to the previous release, one or more of the following optional compilers must be installed on the current release:

5738RG1 RPG/400*	*PRV Base Support
5738RG1 RPG/400	*PRV System/36 compatible RPG/II
5738CB1 COBOL/400*	*PRV Base Support
5738CB1 COBOL/400	*PRV System/36 compatible COBOL
5738SS1 OS/400	*PRV CL Compiler Support
5738CX1 C	*PRV Compiler Support
5738FT1 Fortran/400	*PRV Compiler Support

Creating the Object for the Previous Release

If the object is a program (*PGM) or a C Local Description (*CLD), create the object on the current release using the appropriate Create command along with the TGTRLS(*PRV) parameter. All other object types can skip this step. If the object was created on, or is restored from, the previous release and is not created again on the current release, you can skip this step. To determine what release the object was created on, use the DSPOBJD command to display the *System level* value.

It is recommended that previous and current release programs be stored in separate libraries to simplify maintenance. Using the DSPPGM command, the *Allow save for previous release* field shows if a program can be saved to the previous release.

For V2R1.1

Using the DSPPGM command, the *Earliest release that program can run* field shows if a program can be saved to the previous release.

The following languages provide Create commands that support the TGTRLS parameter:

- C
- CL (AS/400 Environment)
- COBOL/400 (AS/400 Environment)
- COBOL/74 (S/36 Environment)
- FORTRAN/400*
- RPG/400
- RPG/II (S/36 Environment)

RM/COBOL-85* creates programs that will run on V1R3 or later systems.

The following languages do not support the TGTRLS parameter:

- BASIC
- CL (S/38 Environment)
- COBOL/74 (S/38 Environment)
- PASCAL
- PL/I
- RPG/III (S/38 Environment)

Saving the Object for the Previous Release

You must save the object on the current release using the TGTRLS parameter before restoring it on the previous release. This saves the object in a format that is known to the previous release.

The objects can be moved from the current release system to the previous release system by using communication lines or removable storage media (tape or diskette).

It is recommended that previous release and current release objects be stored in separate libraries to simplify maintenance.

The following save commands support the TGTRLS parameter:

- Save Object (SAVOBJ)
- Save Changed Objects (SAVCHGOBJ)
- Save Library (SAVLIB)
- Save Document Library Objects (SAVDLO)
- Save Calendar (SAVCAL) (you must have OfficeVision/400 licensed program installed to use this command)

Object compatibility is provided for most object types supported on *both* levels as long as the object only uses previous release function.

The following table shows the object types that support and do not support current release to previous release capability. IBM does not support saving IBM-supplied objects (such as system commands, programs, IBM spelling aid dictionaries) from the current release and restoring them on a previous release system.

Table 8-1 (Page 1 of 2). Previous Release Support by Object Type

Object Type	Supported	Not Supported
*ALRTBL	X	
*AUTHLR		X
*AUTL		X
*CFGL	X	
*CHTFMT	X	
*CLD	X	
*CLS	X	
*CMD	X	
*CNNL		X
*COSD		X
*CSPMAP	X	
*CSPTBL	X	
*CSI		X
*CTLD		X
*DEVD		X
*DOC	X	
*DTAARA	X	
*DTADCT	X	
*DTAQ	X	
*EDTD	X	
*FCT	X	
*FILE (database, device, save)	X	
*FLR	X	
*FNTRSC	X	
*FORMDFN	X	
*GSS	X	
*IGCDCT	X	
*IGCSRT	X	
*IGCTBL	X	
*JOBD	X	
*JOBQ	X	
*JRN	X	
*JRNRCV	X	
*LIB	X	
*LIND		X

Table 8-1 (Page 1 of 2). Previous Release Support by Object Type

Object Type	Supported	Not Supported
*MENU	X	
*MODD		X
*MSGF	X	
*MSGQ	X	
*NWID		X
*OUTQ	X	
*OVL	X	
*PAGDFN		X
*PAGSEG	X	
*PDG	X	
*PGM (see note) - BASIC - C - CL[1] - CL[2] - COBOL/400[2] - COBOL/74[1] - COBOL74[3] - CSPAE - FORTRAN/400 - PASCAL - PL/I - RM/COBOL-85 - RPG/II[3] - RPG/III[1] - RPG/400[2]	 X X X X X X X X X	 X X X X X X
*PNLGRP	X	
*PRDAVL		X
*PRDDFN		X
*PRDFUN		X
*PRDLOD		X
*QMFORM	X	
*QMQRY	X	
*QRYDFN	X	
*RCT		X
*SBSD		X
*SCHIDX	X	
*SPADCT	X	
*SSND	X	
*S36		X
*TBL	X	
*USRIDX	X	
*USRPRF		X
*USRQ	X	

Table 8-1 (Page 2 of 2). Previous Release Support by Object Type

Object Type	Supported	Not Supported
*USRSPC	X	

Notes:

1 System/38 environment

2 AS/400 environment

3 System/36 environment

Assuming the program has been created on the current release. If a program was created on an earlier release (V1R1, V1R1M2, V1R2, V1R3) and it has not been created on the current release, it can be saved using TGTRLS(*PRV) and restored and run on a V1R3 system. To determine what release a program was created on, use the DSPOBJD command to display the system level value.

Testing the Object on the Current Release

Once the object has been created and saved using the TGTRLS parameter it can be tested on the current release system. Thus, it is no longer necessary to support and maintain two development systems (one running the current release and one running the previous release). Testing this object should be like testing any other object. Make sure that all the objects that are to be used on the previous release system have been saved using the TGTRLS parameter, restored onto the current release system, and tested as a group on the current release system.

Restoring and Using the Object on the Previous Release

Once testing on the current release system is completed, it is recommended that the object be distributed on a limited basis to previous release systems. Thus, if problems arise, they can be quickly corrected and contained with minimal impact to users.

Restrictions for Current Release to Previous Release Support

The following restrictions apply when you create and save objects on the current release, and then restore and use them on the previous release.

- A program created and saved on a V2R1 system using TGTRLS(*PRV) with the observability tables of the program removed, will restore and execute correctly on a V1R3 system.

  If you decide to remove the observability tables of a program, it is advisable to keep a copy of the observable form of the program and the program source.
- The S/38 environment compilers (CL, COBOL/74, and RPG/III) do not support TGTRLS(*PRV). Programs created using these compilers cannot be saved to, restored, or run on a previous release system.
- The only way to save an object for a previous release system is to use the TGTRLS parameter. If TGTRLS(*PRV) is not specified on the SAVE command, and you attempt to restore the object on the previous release system, the object is not restored.
- TGTRLS(*PRV) supports only the creating or saving of objects on the current release and restoring them to the previous release. For V2R1, the previous release is V1R3.
- IBM does not support saving IBM-supplied objects (such as system commands, system programs, IBM spelling aid dictionaries, and so forth) from a current release system and restoring and using them on a previous release system. As a result, the TGTRLS parameter is not supported on a SAVLIB command that specifies *ALLUSR, *IBM, or *NONSYS on the LIB parameter.
- IBM does not support new function from the current release to be used on a previous release system. The database file attribute to re-use deleted records is not lost when the file is saved for the previous release and then restored on the current release.
- When saving document library objects for a previous release, only folders and filed documents can be saved. Other items, such as mail or unfiled documents, cannot be saved using TGTRLS(*PRV).
- If a current release program temporary fix (PTF) save file is sent from a current release system to a previous release system for distribution to another current release system, object distribution must be used. The Copy PTF (CPYPTF) command or any save file command, such as DSPSAVF, cannot process the PTF save file.

Previous Release to Current Release Support

Generally, the system to which you are restoring objects must be at the same or a higher release level than the system from which the objects were saved. When moving data to a higher level release, you should only move user data. This may include user libraries, user profiles, user objects in IBM-supplied libraries, document library objects (documents and folders), and mail. IBM-supplied libraries should not be restored to a higher release since these are handled during the licensed program install process. The target system should have the current level release installed. This includes the licensed internal code, OS/400 operating system, IBM-supplied libraries QGPL and QUSRSYS, OS/400 optional libraries, and any licensed programs purchased. See the *Licensed Programs and New Release Installation Guide*, to install the current release.

Configuration objects are not normally moved to a different system since each system has different physical devices, communications lines, and so on. If you do have similar configurations, you can use the RSTCFG (Restore Configuration) command on the target system; however, you will probably need to modify the resource IDs in the configuration objects after the restore. Configuration data such as network attributes, system values, RJE configuration, SNADS configuration, DSNX data, Problem Log, and Q&A Database cannot be saved from one system and restored to the higher release system. This data must be re-created manually on the target system.

This section describes restoring objects previously saved on another AS/400. For information about restoring System/38 or System/36 objects, see the *System/38 to AS/400 Migration Aid User's Guide and Reference* and *System/36 to AS/400 Migration Aid User's Guide and Reference*. These instructions are **not** the intended method to perform a release upgrade. The *Licensed Programs and New Release Installation Guide* should be used to upgrade the previous release to the current release.

When saving data on the source system, the contents of output queues, job queues, data queues, and message queues are not saved; only their descriptions are saved. If your source system is at V1R3 or later, the contents of save files are saved if you specify SAVFDTA(*YES) (the default) on the save command. If your system is prior to V1R3, the contents of save files are not saved. To save spooled output files, use the Copy Spooled File (CPYSPLF) command to copy the spooled file to a database file member before saving.

If mail is moved from a previous release system to the current release (V2R1), the following information will be lost:

- The *sent or received* information in a hard-copy reference mail item.
- The *assigned to* information in all mail items.
- The retain ID for a file pending mail item causing the status of the mail item to remain at OPEN.
- The internal *file and delete from mail log* indicator causing the mail item to remain in the mail log after the item has been filed.

Saving Data on a Previous Release System

In order to move the user data to the current release system, you must first save the data from the source system. Perform one or more of the following steps to save the data you wish to move:

1. Print a list of libraries and user objects in IBM libraries.

   Use the DSPOBJD command to print a list of all libraries on your system. Some of the libraries listed are IBM-supplied libraries and should **not** be moved to the target system. These include #CGULIB, #COBLIB, #DFULIB, #RPGLIB, #SDALIB, #SEULIB, #DSULIB, and all libraries that begin with the letter 'Q' except QS36F.

   ```
   DSPOBJD OBJ(*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```

   Use the DSPLIB command to print a list of all objects in IBM-supplied libraries QGPL and QGPL38. These libraries may contain user data that you wish to move to the target system.

   ```
   DSPLIB LIB(QGPL QGPL38) OUTPUT(*PRINT)
   ```

2. Process the objects in library QRCL.

   Enter the following command to print objects in QRCL:

   ```
   DSPLIB LIB(QRCL) OUTPUT(*PRINT)
   ```

   If library QRCL does not exist, or there are no objects in the library, continue with the next step. If there are objects in the library, see "Considerations for Lost Objects" on page 9-4 to delete or move objects found in library QRCL.

3. Change system operator message queue delivery mode.

   Use the CHGMSGQ command to change the system operator message queue to break mode.

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

4. End all subsystems.

   Use the ENDSBS command to end all subsystems. This will result in improved save performance and avoid any save failures due to objects locked.

   ```
   ENDSBS *ALL *IMMED
   ```

5. User profiles and mail (distribution objects)

   If you are moving user profiles to the target system, enter the SAVSECDTA command to save the security objects. If you are moving mail, specify MAIL(*YES) on the SAVSECDTA command; otherwise, specify MAIL(*NO). If your source system is at V1R1 or V1R1.2, you must use the SAVSYS command to save the user profiles and mail cannot be saved. Enter **one** of the following commands to save the user profiles:

   ```
   SAVSECDTA DEV(tape-device) MAIL(*YES)
   ```

   ```
   SAVSYS DEV(tape-device)
   ```

6. User data in QAO* files in library QUSRSYS

   Use the SAVOBJ command if you wish to save user data from files beginning with QAO* in library QUSRSYS. These files include system directory files, distribution list files, office calendar files, office enrollment files, and document library search index files.

   Warning:

   This step should only be performed if the current release has just been installed on the target system and there is no user data on the system (licensed internal code was installed with option 24). In addition, the source system must be at V1R2 or higher. The QAO* files cannot be moved if the source system is at V1R1 or V1R1.2.

   Restoring these files will replace the existing data on the target system. There is no capability to merge the data in the QAO* QUSRSYS files from the source system with the data on the target system.

   Enter the following command to save the data:

   ```
   SAVOBJ OBJ(QAO*) LIB(QUSRSYS) DEV(tape-device)
        OBJTYPE(*FILE) ACCPTH(*YES)
   ```

7. User libraries

   Using the DSPOBJD listing from step 1, identify the user libraries you wish to move. Do NOT move IBM libraries to the target system. These include #CGULIB, #COBLIB, #DFULIB, #RPGLIB, #SDALIB, #SEULIB, #DSULIB and all libraries that begin with the letter 'Q' except QS36F.

   Enter the SAVLIB command to save only the user libraries that you wish to move. Libraries QS36F and #LIBRARY are considered user libraries and should be specified on the SAVLIB command. Up to 50 libraries can be specified on each SAVLIB command when saving the user libraries. If you have a large number of user libraries to save, you may want to create a CL program to perform the SAVLIB operations. This step will preserve the object ownership and public authority; however, private authorities will not be saved with the objects. If you did not save the user profiles, you will need to manually grant private authorities to the objects after restoring to the target system.

   Note: Do not specify *NONSYS for the LIB parameter since this option saves IBM libraries in addition to user libraries.

   Enter the following command to save user libraries:

   ```
   SAVLIB LIB(user1 user2 userN) DEV(tape-device) ACCPTH(*YES)
   ```

8. User objects from IBM-supplied libraries

   IBM libraries that contain user data (for example, QGPL) must be handled with special instructions and must not be restored to the target system using the RSTLIB command. RSTLIB would overlay current release IBM objects with previous release objects and destroy the PTF status information that currently exists in the library. In addition, the install process may delete obsolete objects in these libraries during a release upgrade and you would re-create these objects if you used RSTLIB.

   The IBM libraries that may contain user data are QGPL, QGPL38 and QUSRSYS. To move all user output queues and message queues from library QUSRSYS, enter the following:

   ```
   SAVOBJ OBJ(*ALL) LIB(QUSRSYS) OBJTYPE(*MSGQ *OUTQ)
   ```

   To move user objects contained in libraries QGPL and QGPL38, perform the following for each IBM library containing user data:

   a. Using the DSPLIB listing from step 1, identify the user objects you intend to move. Do not move IBM objects from a previous release to the current release system except for IBM-supplied source files. The IBM-supplied source files in library QGPL have the naming convention QxxxSRC, where xxx identifies the type of source file (such as QCLSRC, QCMDSRC, QDDSSRC). Most IBM objects begin with the letter 'Q'. Other IBM objects can usually be identified by the text associated with the object (for example, created by Auto-Configuration). If you have created members into IBM-supplied source files, include these files on the SAVOBJ command.

   b. Enter the following command:

      ```
      SAVOBJ OBJ(A B C QCLSRC) LIB(library-name)
               DEV(tape-device)
      ```

      This command saves the objects identified in the previous step (A, B, C and QCLSRC in this example) from the specified IBM library. Up to 50 object names can be specified on each SAVOBJ command.

9. Documents, folders and mail (distribution documents)

   The SAVDLO command saves all the documents and folders on the system and all distribution (mail) documents if the source system is at V1R2 or higher. If the source system is prior to V1R3, all subsystems must be ended (ENDSBS *ALL *IMMED) in order to save all documents and folders. If you only wish to move specific documents or folders, specify the desired documents and folders on the SAVDLO command. If you only wish to move mail, specify DLO(*MAIL) on the SAVDLO command. Enter the following command to save all documents, folders, and distribution documents:

   ```
   SAVDLO DLO(*ALL) FLR(*ANY) DEV(tape-device)
   ```

10. Print job log

    The printed output should be kept with the save media to determine what libraries and objects to restore on the target system. Enter the following command to print the job log:

    ```
    SIGNOFF *LIST
    ```

11. Start subsystems

    Enter the STRSBS command for the controlling subsystem and any other subsystems that must be active for normal system operations.

    ```
    STRSBS SBSD(subsystem-name)
    ```

Restoring Data to the Current Release System

The target system should have the current release installed. To verify, enter GO LICPGM and select option 10 (Display installed licensed programs). The *Installed Release* field for the Operating System/400 licensed program should be *V2R1M0*. If not, install the target system with the current release (OS/400 and all ordered licensed programs) before restoring the previous release data. See the *Licensed Programs and New Release Installation Guide* to install the current release.

If your system came preloaded with V2R1, you need to obtain V2R1 distribution media if you are restoring all data from the previous release.

Perform the following instructions to restore the user data:

1. Type QSECOFR in the user prompt and the associated password.

The data should be restored using the QSECOFR user profile to avoid restore failures due to insufficient authorization. Enter the CHGMSGQ command to change the system operator message queue to break mode.

```
CHGMSGQ QSYSOPR *BREAK
```

2. Save V2R1 PC Support/400 data.

   If you have PC Support/400 installed on the target system, save the QIWS* folders to a temporary save file. If PC Support/400 is not installed, or you are **not** restoring all documents and folders from the source system, skip this step. Use the DSPFLR command to print a list of all folders on your system.

   ```
   DSPFLR FLR(*ALL) OUTPUT(*PRINT)
   ```

   Using the DSPFLR listing, identify all folders that start with 'QIWS'. Create a temporary save file TEMPPCS in library QGPL. Specify the QIWS folders on the FLR parameter of the SAVDLO command to save the V2R1 PC Support/400 folders:

   ```
   CRTSAVF QGPL/TEMPPCS
   ```

   ```
   SAVDLO DLO(*ALL) FLR(folder1 folder2 foldern) DEV(*SAVF)
          SAVF(QGPL/TEMPPCS)
   ```

3. End all subsystems.

   If you will be restoring user profiles (RSTUSRPRF) or private authorities (RSTAUT), all subsystems must be ended. The ENDSBS command must be entered from a workstation in the controlling subsystem. Enter the following commands to end all subsystems:

   ```
   ENDSBS *ALL *IMMED
   ```

4. Restore user profiles.

   Restore the user profiles onto the target system using the SAVSECDTA or SAVSYS media. If SAVSYS media is used, the user profiles are located in the next to the last tape file (labelled QFILEUPR). If you are only moving specific user profiles, specify up to 50 names for the USRPRF parameter rather than *ALL. Enter the following command to restore the user profiles:

   ```
   RSTUSRPRF DEV(tape-device) USRPRF(*ALL) MAIL(*NO)
             ALWOBJDIF(*ALL)
   ```

5. Restore user data to QAO* files in library QUSRSYS.

   Perform this step to restore the system directory, personal directories, distribution lists, office enrollment and calendars files to the target system.

   Warning: This step should only be performed if the current release has just been installed on the target system and there is no user data on the system (licensed internal code was installed with option 24). In addition, the source system must be at V1R2 or higher. The QAO* files cannot be moved if the source system is at V1R1 or V1R1.2. The user data in the QAO* files from the source system cannot be merged with existing data on the target system.

   a. Enter the RSTOBJ command to restore the QAO* files to the target system using the corresponding save media. You will receive multiple CPI8A17 messages indicating several files containing document details were not restored. These messages can be ignored.

      ```
      RSTOBJ OBJ(QAO*) SAVLIB(QUSRSYS) DEV(tape-device)
             OBJTYPE(*FILE) MBROPT(*ALL) ALWOBJDIF(*ALL)
      ```

   b. Install QUSRSYS library.

      Perform the following steps to reinstall library QUSRSYS using your V2R1 distribution media. This will initialize the system distribution directory files on the target system.

      1) Type the following and press the Enter key.

         ```
         GO LICPGM
         ```

      2) Select option 11 (Install licensed programs) and press the Enter key.
      3) Type a 1 next to OS/400 - Library QUSRSYS and press the Enter key. The Confirm Install of Licensed Programs display is shown.
      4) Press the Enter key on the Confirm Install of Licensed Programs display.
      5) Type the device name in the *Tape device* field and press the Enter key.

   c. Use the INZSYS command to initialize the office-related files in library QUSRSYS. Inquiry message CPA3703 is displayed if the current release of the system files contains user-created office data. Specify I (Ignore) in response to the message to replace the current release data with the new data.

      ```
      INZSYS
      ```

d. If OfficeVision/400 is installed on the target system, then the following commands must be run. If the *QAO files that were restored to QUSRSYS were saved from a V1R2system, then both steps 1 and 2 must be used. If the *QAO files were saved from a V1R3 system, then only step 2 must be used.

1) Run the following:

```
DLTF       FILE(QUSRSYS/QAOFOWNO)
RNMOBJ     OBJ(QUSRSYS/QAOFDDH)
           OBJTYPE(*FILE) NEWOBJ(QZOFDDP)

CRTDUPOBJ  OBJ(QZOFOWNL)
           FROMLIB(QOFC)  OBJTYPE(*FILE)
           TOLIB(QUSRSYS)

RNMOBJ     OBJ(QUSRSYS/QZOFOWNL)
           OBJTYPE(*FILE) NEWOBJ(QAOFOWNO)

RNMOBJ     OBJ(QUSRSYS/QZOFDDP)
           OBJTYPE(*FILE) NEWOBJ(QAOFDDH)
```

2) Run the following:

```
CRTDUPOBJ  OBJ(QZOFASGP)
           FROMLIB(QOFC)  OBJTYPE(*FILE)
           TOLIB(QUSRSYS)

CRTDUPOBJ  OBJ(QZOFASGL)
           FROMLIB(QOFC)  OBJTYPE(*FILE)
           TOLIB(QUSRSYS)

DLTF       FILE(QUSRSYS/QAOFASGO)

DLTF       FILE(QUSRSYS/QAOFASGH)

RNMOBJ     OBJ(QUSRSYS/QZOFASGL)
           OBJTYPE(*FILE) NEWOBJ(QAOFASGO)

RNMOBJ     OBJ(QUSRSYS/QZOFASGP)
           OBJTYPE(*FILE) NEWOBJ(QAOFASGH)
```

6. Enroll users in system distribution directory.

   If you are restoring mail to the target system and did not perform the previous step, you may need to add users to the system distribution directory. If you performed the previous step, the system directory was already restored and you can skip this step.

   Use the WRKDIR (Work Directory) command to determine if the user profiles you moved are in the system distribution directory. If not, type a 1 (Add) in the *Opt* column to add each user with mail to the system directory on the target system.

   ```
   WRKDIR
   ```

7. Restore the mail (distribution documents).

   If you are restoring mail to the target system, enter the RSTUSRPRF command again with the MAIL parameter set to *YES. Use the SAVSECDTA media and enter the following command to restore the distribution objects (mail).

   ```
   RSTUSRPRF DEV(tape-device) USRPRF(*ALL) MAIL(*YES)
          ALWOBJDIF(*ALL)
   ```

8. Grant *ALLOBJ special authority.

   If the target system is at security level 30 or 40 and any of the user profiles moved to the target system had *ALLOBJ special authority, use the CHGUSRPRF command to grant *ALLOBJ authority back to each profile that requires all object authority. The restore process removes *ALLOBJ special authority from non-IBM user profiles if the security level is 30 or 40. Enter the following command where xxx and yyy are the existing special authority values for the user profile:

   ```
   CHGUSRPRF USRPRF(user-profile) SPCAUT(xxx yyy *ALLOBJ)
   ```

9. Restore the user libraries.

   Use the RSTLIB command for each user library saved. The user libraries were listed on the LIB parameter of the SAVLIB command(s) performed on the source system. If you have a large number of user libraries to restore, you may want to create a CL program to perform the RSTLIB operations.

   ```
   RSTLIB SAVLIB(user1) DEV(tape-device) ENDOPT(*LEAVE)
        MBROPT(*ALL) ALWOBJDIF(*ALL)
   ```

   Note: If you restore library #LIBRARY, you may receive error message CPF3761 indicating object QS36ENV was not restored. This message can be ignored.

10. Restore user objects to IBM-supplied libraries.

    Enter one or more RSTOBJ commands for each IBM library you saved user objects from on the source system (for example, QGPL). The save media should only contain objects you created into the IBM libraries or IBM-supplied source files (for example, QCLSRC) that you have created members into.

    ```
    RSTOBJ OBJ(*ALL) SAVLIB(QUSRSYS) DEV(tape-device)
        OBJTYPE(*MSGQ *OUTQ) ENDOPT(*LEAVE) MBROPT(*ALL)
        ALWOBJDIF(*ALL)

    RSTOBJ OBJ(*ALL) SAVLIB(library-name) DEV(tape-device)
        ENDOPT(*LEAVE) MBROPT(*NEW) ALWOBJDIF(*ALL)
    ```

11. Restore documents and folders.

    Use the RSTDLO command to restore the document library objects saved from the source system. If distribution documents (mail) were saved from the source system, they will be restored at this time. If multiple tape files were created from the SAVDLO on the source system, you must specify a beginning and ending sequence number (SEQNBR parameter) on the RSTDLO command.

If the source system is at V1R1 or V1R1.2 and more than 8000 objects were saved, each group of 8000 objects was saved in separate files with labels DLO001, DLO002, and so on. To restore documents and folders from files with these labels, you must specify LABEL(DLO00n) on the RSTDLO command for each tape file.

Enter the following command to restore the documents, folders, and mail (distribution documents):

```
RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(tape-device)
       ALWOBJDIF(*ALL)
```

12. Restore V2R1 PC Support/400 data.

    If you saved the PC Support/400 documents and folders in step 2, restore the folders from the temporary save file. If you did not save the PC Support/400 folders, skip this step.

    Enter the following command to restore the V2R1 PC Support/400 folders:

    ```
    RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(*SAVF)
           SAVF(QGPL/TEMPPCS) ALWOBJDIF(*ALL)
    ```

    Enter the following command to delete the temporary save file:

    ```
    DLTF QGPL/TEMPPCS
    ```

    Enter the following commands to delete PC Support/400 documents and folders that are not used on V2R1:

    ```
    DLTDLO DLO(VPRT.COM) FLR(QIWSFLR)
    WRKFLR *ALL
    ```

    Type a 4 (Delete) next to QIWSMRI and QIWSMRID and press the Enter key. The Confirm Delete display is shown. Press the Enter key again on the Confirm Delete display.

13. Restore private authorities.

    If you ran the RSTUSRPRF command previously, enter the RSTAUT command to restore private authorities to objects on the target system. If you did not run the RSTUSRPRF command, use the GRTOBJAUT command to grant private authorities to the newly restored objects.

    Enter the following command to restore private authorities:

    ```
    RSTAUT
    ```

14. Start subsystems.

    If you ran the ENDSBS *ALL command, enter the STRSBS command for the controlling subsystem and any other subsystems that must be active for normal system operations.

    ```
    STRSBS SBSD(subsystem-name)
    ```

Chapter 9. Save and Restore Considerations and Command Examples

This chapter provides save and restore considerations and examples of using the save and restore commands.

Considerations for Damaged Objects

Damaged is a term used to describe the condition of any object that for some reason, usually unknown, is in a state in which the system can no longer process it. Although objects can become damaged in many ways, the system is designed to minimize object damage, so damaged objects are not common. Examples of what can cause objects to become damaged follow:

- Physical damage on the auxiliary storage disk drives
- A power failure during a critical update to any object
- A logic error in IBM code that causes the data in an object to be inappropriately updated or left in a state in which the object is unusable
- An equipment error that occurs while an object is being restored

The system provides you with functions to deal with damaged objects should they occur.

When damaged objects are detected during an IPL, the system notifies you by sending messages to the history log and to the system operator message queue QSYSOPR. The message sent to QSYSOPR states that damaged objects were encountered during the IPL and that messages were sent to the history log. You may not be able to use the objects until they have been repaired and are no longer damaged.

When a damaged object is detected other than during an IPL, the system notifies you, and you may not be able to use the object until it has been repaired and is no longer damaged. The first time the system detects a damaged object, it sends messages to your program message queue, the system operator message queue QSYSOPR, and the history log. Thereafter, only the program is notified and the damaged object is referenced in the job log.

Because object damage is not common, your programs probably will not monitor for the escape messages sent for damaged object conditions. The escape messages that are not monitored cause function checks. (See the *CL Programmer's Guide* for more information about monitoring for messages.) For interactive command entry, a function check is always handled by the command processor QCMD or the IBM-supplied program associated with the IBM-supplied menu from which a function was selected. For batch jobs, the default action for a function check that is not handled is to end the job. Only a single function or job is affected. If an object that is critical to the operation of a subsystem or system is damaged, a larger group of jobs is affected.

If a job description becomes damaged, it cannot be changed, displayed, or used to start jobs. A damaged job description is indicated by escape message CPF1618. You must delete the job description and create it again or restore it from a saved copy. If the job description specified for the console work station entry (in the subsystem description, for the controlling subsystem) is damaged, the console will not be usable at the end of the IPL operation.

When this situation occurs and no other work station entries exist in the subsystem description, the system is not useable. To avoid this damage, you can create another controlling subsystem that specifies a work station as another device from which jobs can be accepted.

The recovery procedure you follow for a damaged object depends on the type of damage and what object was damaged (an IBM-supplied object or a user-defined object). If an IBM-supplied object is damaged, you must either delete and then restore or create the object again, do an IPL again, or install the system again (unless the message indicates that the object is automatically created again by the system). (See the *Operator's Guide* for more information about these messages.) For user-defined objects, there are three types of damage:

- Header damage. The object cannot be accessed or used, and the name and type of object are not known.
- Full object damage. The object cannot be used, but the name and type of object are known.
- Partial object damage. Only part of the object is usable, and the name and type of object are known.

The recovery procedure you should follow depends on the type of damage.

In general, if you refer to an object and header damage is detected, a message is sent to the QSYSOPR message queue. Do not try to repair the header damage condition. The system tries to free the storage occupied by the object with header damage. After it has been freed, you will receive an object-not-found condition when you refer to the object. To recover from the object-not-found condition, restore or create the object again. For complex objects such as database files, it may be necessary to reclaim storage to find the lost pieces and either delete them or recover them if the user's data is found. If you cannot proceed, your system may need to be installed again.

For some damaged objects such as database files or device descriptions, special recovery procedures are possible. For example, when a database file is partially damaged, you can attempt to copy the data to another file using the Copy File (CPYF) command to keep the readable data, or you can use a different recovery method if you have one (for example, restoring or journaling). When you copy a file, the system can optionally bypass unreadable data and copy only readable data. (The *Data Management Guide* has more information about bypassing unreadable data.) In some cases of partial object damage to device descriptions, you can recover by varying the associated device off and on.

In most instances, to recover from full or partial object damage to a user-defined object, you must delete the damaged object, and then either restore a saved copy or create the object again. Exceptions to this are job queues and output queues. You cannot use a delete command to recover from these damaged objects. You should power down the system and then start the system again. The system automatically deletes the damaged job queues and output queues. If the default output queue for a printer is damaged, the system automatically creates the queue again and creates queue entries for all files that were on the queue before it was damaged. Other output queues and job queues are not automatically created again by the system. All jobs or files on these queues are lost. You can create these queues again when the system has finished the IPL.

Every part of a damaged object that can be found is deleted and its storage freed. Sometimes, some parts of a damaged object may not be deleted although it appears to you that they have been. If this happens when you are trying to delete an object, a diagnostic message is issued and every part of the damaged object found is deleted.

How the Save Operation Handles Damage

Following is an overview of unique situations when the save operation determines that damage exists in the system. The situations vary depending on the extent of the damage and the type of object involved.

- Prior to the save operation, the system determines if an object is damaged and marks the object. An object marked as damaged cannot be saved, but the save operation continues with the next object. The operation completes with an indication of how many objects were saved and how many were not. Diagnostic messages describe the reason each object was not saved.
- If an object is not marked as damaged, the save operation attempts to save the object. If the object is found to be damaged, the save operation ends and the object is marked as damaged. The objects that are already marked as damaged are bypassed on the next save operation.
- There can be some unusual cases where a save operation does not detect damage. The save operation may detect physical damage on the disk, but it may not detect all damage. For example, the system does not attempt to determine if all bytes within an

object are valid and consistent (logical damage). For some cases, you will not be able to determine a damage condition unless you attempt to use the object (such as call a program object). If this type of damage exists, the object will be restored normally. Database files can be checked for most damage conditions by the VALDBF tool in QUSRTOOL library.

How the Restore Operation Handles Damage

Following is an overview of some unique situations in which a restore operation determines that damage exists on the media or in the system. The situations vary depending on the extent of damage and the type of object involved.

If damage exists within the data portion of a database file that has already been marked as partially damaged, the system bypasses the object and does not restore it. If this situation occurs, you must first delete the object on the system and then do the restore operation.

When a database file that exists on the system is restored, the existing address for the data in storage is used for the restore operation. Consequently, any data on the system is overlaid during the restore operation. When restoring some object types over existing objects, the system gives the restored copy a new address, and deletes the old copy if the restore operation is successful.

Restoring Over Objects Marked as Damaged

The Restore Library (RSTLIB) and Restore Object (RSTOBJ) commands allow a restore operation over objects that are marked as damaged. If a system object is found that is marked as damaged, the object is deleted from the system and the object is restored from the media.

A message is sent to the history log indicating that a damaged object was found. Then the restore operation attempts to delete the object, if eligible, and restore the object from the media. For objects that are not eligible for deletion, a message is sent to the user indicating that a damaged object cannot be restored. The restore operation continues with the next object.

If the object is deleted successfully, two messages are sent indicating that the object was damaged and then deleted. Another message is sent indicating that the object is restored, all authorities to the object are lost, and the public authority for the object is changed to *EXCLUDE.

Consider the following when restoring over objects that are marked damaged:

- The processing for objects that are partially damaged and for damage that is not marked remains the same.
- The damaged system object is deleted and then restored only if the object is eligible for deletion. Object types that are not eligible for deletion are:
  - Database files and members (*FILE - database)
  - Documents (*DOC)
  - Folders (*FLR)
  - Job queues (*JOBQ)
  - Journals (*JRN)
  - Journal receivers (*JRNRCV)
  - Libraries (*LIB)
  - Output queues (*OUTQ)
- Authorities to the damaged object are lost.

Media Damage during a Restore Operation

If you receive a media error from which you can recover, the operation is not ended. Any objects that are damaged on the media are not restored, but all other objects are. The diagnostic message CPF3864 (CPF3811 when a save file is used) is issued for each OS/400 object that cannot be restored indicating that the object was damaged on the media.

If any part of an object, a part of a subsystem description, for example, is damaged on the media, the entire object is considered damaged. If an access path for a member cannot be restored because of media damage, it is automatically rebuilt. Only those members of a file that are damaged on the media are not restored.

If media errors are discovered when loading an object and the object is partially loaded, what happens depends on whether the same address is being used.

- For a database file, the system marks the object as partially damaged and attempts to continue the restore operation.
- If a different address is being used, the system deletes the partially loaded object, keeps the old object, and attempts to continue the restore operation.
- At the completion of the restore command, a message is sent stating the number of restored objects and the number not restored (diagnostic messages exist for objects not restored).

Significant media damage may cause several objects to be bypassed on the restore operation. If unrecoverable media errors exist, the restore operation is ended.

Damaged Document Library Objects: If you try to restore into a document that already exists on the system, but it is damaged, the restore operation fails for the document. The system ignores the damaged objects and continues the restore operation. A message is sent indicating that the document object is damaged.

If you try to restore a damaged folder into an existing folder, you receive a message informing you that it is damaged and that the folder is ignored. The folder and all documents and folders in it are not restored.

Considerations for Lost Objects

Objects may be lost because of power or equipment failures or other types of system problems. An object is considered to be lost if its addressability has been affected (the object cannot be found in a library), its ownership information is recorded incorrectly, or the interrelationship of pieces of the object is lost. By using the Reclaim Storage (RCLSTG) command, you can correct some of these conditions, delete objects, or remove pieces of objects that cannot be made usable.

Consider using the RCLSTG command in the following situations:

- An attempt was made to start the system, but there was not enough storage for the system to be fully operational.
- The system ended abnormally several times since the RCLSTG command was used last.
- Objects shown on the Owned Objects display show no library names were associated with them.
- The system status display shows an unexpectedly high percentage of auxiliary storage used.
- Data cannot be accessed because a database file member is damaged. This condition is indicated by message CPF8113.
- Objects cannot be accessed because they are secured by a damaged authorization list or damaged authority holder.

Note: Using the RCLSTG command requires some auxiliary storage to run, and if auxiliary storage is already being used extensively, the RCLSTG command can fail because not enough storage is available for it to run.

The purpose of the RCLSTG command is to ensure that objects residing permanently in auxiliary storage can be accessed and that all auxiliary storage is either used properly or is available for use. Every object that resides permanently in auxiliary storage is checked, and the following happens:

- If an object is encountered that does not address a library, it is placed in an IBM-supplied library based on the object type. If the object is not required to be in a specific library, it is placed in library QRCL. For all lost objects, information such as the programming temporary fix (PTF) status, save and restore information, and text descriptions may no longer be retrievable, even though the objects themselves are reclaimed.
- If the system encounters a lost object that has the same name as another lost object, is of the same object type, and has already been placed in QRCL, the object is given a new name as it is placed in QRCL. The new name has this format: `QRCLnnnnnn`, where `nnnnnn` is a unique number assigned to the object. The former, duplicate name is kept as the text description for the object.

- If data still exists for a lost or damaged physical file, the file may be rebuilt. The text description for the file indicates whether the file has been rebuilt. If a physical file is rebuilt, it is not a duplicate of the original file because of the damage encountered. The file keeps as many attributes as the system can recover. After the RCLSTG command is complete, the data can be copied from the rebuilt file. However, because the file is not a duplicate of the original file, it should not be used as a production file. (Not all types of damage can be handled.)
- If the system encounters an object that does not have an owner, an IBM-supplied user profile is assigned to the object based on the object type. If the object type is not required to be owned by a specific user profile, it will be assigned to QDFTOWN user profile.
- If the descriptions of the objects in a library (as shown using the Display Object Description (DSPOBJD) command) are damaged in such a way that they cannot be accessed, the library is rebuilt.
- If the system encounters an object that is secured by a damaged authorization list or damaged authority holder, the system authorization list (QRCLAUTL) is granted authority to the object.

  The Display Authorization List Objects (DSPAUTLOBJ) can be used to display the objects secured by Authorization list QRCLAUTL. Authorization list QRCLAUTL is created during a reclaim storage operation if it is needed.

After the reclaim storage function has finished running, you can display the system operator message queue (QSYSOPR) or the history log (QHST) for a record of what was done. You can use the DSPOBJD command or the Display Library (DSPLIB) command to obtain a list of the reclaimed objects in library QRCL. When you have this information, you can do the following:

- Delete unusable objects.
- Move objects to another library and rename them if necessary.
- Grant authority for objects if necessary.
- Transfer ownership of objects.
- Copy data from rebuilt files to other files.

It is possible that some IBM-supplied objects are in QRCL. If so, see your service representative to determine which library these objects belong in.

Using the Reclaim Storage (RCLSTG) Command: Before you can use the Reclaim Storage (RCLSTG) command to reclaim lost storage, you must place the system in a restricted state. *All* subsystems, including the controlling subsystem, must be ended. This can be done using either the ENDSYS or ENDSBS(*ALL). It is recommended that you perform an IPL before running the RCLSTG command.

If power or equipment failures or other system problems occur while running the RCLSTG command, processing might stop. If so, enter the command again immediately or, if the system was ended, enter it after the system is started again.

While you are running the RCLSTG command, status messages that inform you of current processing are shown on the bottom line of the display. During object processing, the percentage of objects already processed is displayed on the bottom line of the display. Messages may be sent informing the operator of actions taken during the reclaim storage (RCLSTG command) operations.

Note: The RCLSTG command is a long-running function and the amount of time it takes depends on the number of objects in the system, the types of those objects, the amount of damage to them, the amount of auxiliary storage configured to the system, and the percentage of auxiliary storage in use. You should consider when to use the RCLSTG command because the system must be in the restricted state, and the RCLSTG command can require a considerable amount of time to run.

For an example of how to use the RCLSTG command in an unattended situation, see the DLYCMD tool in library QUSRTOOL.

Save and Restore Status Information

When an object is saved, the system updates the object description with information about the save and restore process. The following information is updated:

- Save date and time
- Save command used
- Size of the object on the system at the time of the save
- Identifier volumes (up to ten) containing the last copy saved (tape and diskette only)
- If the data portion of the object resides on the media
- File sequence number (tape only)
- Save file name (if the last save operation was to a save file)

When performing a save operation to a save file, you can specify UPDHST(*NO) to prevent updating the save and restore information of the objects being saved. If UPDHST(*NO) is specified on a save, the last save date is not updated. In this case, SAVCHGOBJ and Apply Journal Change (APYJRNCHG) commands cannot use the date of the last save.

You can display the status information using the Display Object Description (DSPOBJD) command and specify DETAIL(*FULL). Only the object attributes are shown. The data attributes and the actual data are not shown. The status information is shown for only the most recent save of each object kept.

Six data areas in library QSYS contain save and restore history information and do not contain data. The system uses the data area's object description to record the save and restore information that can be shown using the DSPOBJD command.

The following is a list of data areas containing save and restore information:

- QSAVLIBALL contains the last use of the SAVLIB LIB(*NONSYS) and RSTLIB SAVLIB(*NONSYS) commands.
- QSAVALLUSR contains the last use of the SAVLIB LIB(*ALLUSR) and RSTLIB SAVLIB(*ALLUSR) commands.
- QSAVIBM contains the last use of the SAVLIB LIB(*IBM) and the RSTLIB SAVLIB(*IBM) commands.
- QSAVUSRPRF contains the last use of SAVSYS, SAVSECDTA, and RSTUSRPRF USRPRF(*ALL) commands.
- QSAVSYS contains the last use of the SAVSYS command.
- QSAVSTG contains the last use of the SAVSTG command and the restore storage function in DST.

  Note: The history information is not updated for individual objects when using the SAVSYS or SAVSTG command. You must use the DSPOBJD command to display this data area to see the date and time of the last save system or save storage operation.

The following is an example of displaying QSAVLIBALL data area using the DSPOBJD command:

```
DSPOBJD  OBJ(QSYS/QSAVLIBALL) OBJTYPE(*DTAARA) DETAIL(*FULL)
```

Save and Restore Completion Messages

The completion message sent for each save and restore command indicates how many objects were saved or restored and how many were not. An object is reported as not saved or restored only if it fits the selection values in the command and it was not saved or restored. Examples of why an object is not saved or restored are:

- The object is in use. If a single member of a multiple member file is in use, the entire file is not saved.
- The object is damaged.
- The user is not authorized to save or restore the object.

Messages When Objects Are Not Saved or Restored: If an object is not saved or restored, a message is sent to the job log. These messages are shown by pressing F10 (Display detailed messages) from the Command Entry display, or by printing the job log. The messages cannot be retrieved by a program in the job that does the save or restore operation. However, you can end the job and have another job copy the completed QPJOBLOG spooled file to a database file where the messages can be accessed. If you specify OUTPUT(*PRINT), most of this information, such as objects not restored, also appears in the list.

For an example of how to access the information in the job log, see the CHKSAVRST tool in library QUSRTOOL.

Save Completion Messages: When a SAVOBJ or SAVLIB command is used, a completion message CPC3701 for tape or diskette, or CPC3722 for save files, is sent for each library saved and contains the number of objects saved. The message help of the completion message includes the volume identifiers of the first ten volumes used (for tape or diskette); these identifiers are used to update the status information of each object saved. The message data contains this information as well as the last volume ID and the last device, or the last save file used.

If multiple libraries are saved in a single command, a final completion message (CPC3720 or CPC3721) also contains the last device used. If a series of save commands is being performed in a CL program and each save command specifies that multiple devices can be used, you can start each save command on a different device. The device name in the message data can be used to determine where the save operation completed so that the alternative device name can be used on the next save command. The CL program can cause the next save command to start writing output to the same device where the previous save operation completed.

The CL program in the topic "Retrieving the Device Name from Save Completion Messages" on page B-4 retrieves the device name from the CPC3701 message (found in positions 126 through 135 of the message data) and uses the information to determine which device is used by the next save command.

Restore Completion Messages: If an object does not fit the selection values in the restore command, it is not included in the count of how many objects were not restored. For example, assume that the following objects (all files) were saved from the DSTPRODLB library with a SAVLIB command on tape:

- ORDFILL
- ORDHDRP
- ORDDTLP
- INVMSTP

If the following command is issued,

```
RSTOBJ OBJ(ORD*) SAVLIB(DSTPRODLB)
OBJTYPE(*FILE)
```

the resulting messages are:

```
CPF3760  Not authorized to save or restore *FILE ORDFILL
         in DSTPRODLB

CPF3773  2 objects restored, 1 not restored to DSTPRODLB
```

In this example, ORDFILL is not restored because the user is not authorized to restore it. INVMSTP is not included in the count of how many objects were not restored because it does not match the generic name ORD* specified on the OBJ parameter.

If a restore operation ends abnormally due to a security or integrity change, a message is sent instructing you to display specific informational messages. You can display these messages by pressing F10 or by displaying the job log.

A diagnostic message for a security change is sent if:

- Private or public authorities are revoked.
- The owner name is changed.
- The authorization list is not linked.

A diagnostic message for an integrity change is sent if:

- Journaling could not be started for an object that was being journaled at the time of the save operation.
- A logical file is restored over a deleted and re-created physical file.

If one of the above conditions occur, an escape message is sent at the completion of the restore command. You can monitor for this escape message through the MONMSG command. If the job is a batch job, you may need to adjust the message severity level through the Submit Job (SBMJOB) command.

Results of a Successful Restore Operation

When a restore operation is successfully completed, the following steps occur:

- The save and restore history for each object in the restore operation is updated.

- A count of the number of objects successfully restored is sent to the user in a completion message.

Results of an Unsuccessful Restore Operation

A restore operation can be unsuccessful either because an error was encountered when trying to restore an object or because the operation was interrupted. If the object existed on the system before the restore operation, it may be damaged by the unsuccessful restore operation.

An object is not restored when an error is encountered. The error is either recoverable or not.

Restore Operation Error Is Recoverable: If an object cannot be restored and the error is recoverable, the following occurs:

- A diagnostic message is sent to the job log for each object that is not restored.
- Each object associated with the errors is not restored. However, other objects not associated with the errors but involved in the same restore operation are restored.
- Only the save and restore status information for the objects that were successfully restored is updated.
- A count of the number of objects successfully restored and a count of the number of objects not restored are sent to the user in a diagnostic message.

Restore Operation Error Is Not Recoverable: If the error is not recoverable, the following occurs:

- Diagnostic messages are sent to the job log for each object.
- The save and restore status information for each object is not updated.
- A diagnostic message identifying the error condition is sent to the user.
- The restore command ends immediately. No other objects are restored.

Recovering from Device and Media Errors

The following discusses recovery from device and media errors. Device error recovery is done by the system. Restarting the save operation is done by the user.

Automatic Recovery From Tape-Write Errors

Automatic tape-write error recovery is supported by the 1/2-inch and 8-mm tape units; included are the 9347, 9348, 2440, 3422, 3430, 3480 3490, and 7208.

When a tape-write error is encountered on one of the supported tape units, the system repositions the tape to an area before the media error occurred, and then writes end-of-volume labels. The system requests that you load a new tape to continue the operation. After the header labels have been written, writing to tape is started again with the area of data that was overwritten by the end-of-volume labels on the previous tape.

Note: Tape-write error recovery is not supported for the data interchange commands such as CPYFRMTAP or DSPTAP. A media error message is sent.

The tape-write error recovery is only supported by the save and restore commands.

Tape Write Error Messages: When data error occurs when writing to tape, message CPA400F is sent asking you if you want to continue on another volume or cancel the job. If you select to continue, and the volume can be closed by writing end-of-volume labels, message CPI4025 is sent. This message informs you that the tape had an error and it should probably be replaced before the next save operation is done with this set of tapes. If the end-of-volume labels can not be written, message CPF389A *Tape media error* is sent to indicate that the recovery failed.

A media error near the beginning of a volume may result in message CPA401A instead of CPA400F. This message informs the user that the volume is not usable and should be removed from this set of tapes. The user has the option of ending the save operation or trying the operation again using a new volume.

Media or Device Error When Running the SAVLIB or RSTLIB Command

The STRLIB parameter is only for recovery situations in which the *NONSYS, *IBM, or *ALLUSR save or restore operation ended or failed because of a media or device error. If a media or device error occurs and cannot be handled by the automatic tape error recovery, you do not have to start from the beginning again. A new tape can be loaded and you can run the RSTLIB or SAVLIB command again and specify the STRLIB parameter to start the save or restore operation again at a specific library, or to use the default value of *FIRST. The default value begins the save or restore operation at the first library. (The libraries are saved or restored in ascending alphabetical order.)

The following restrictions apply when the STRLIB parameter is specified using *NONSYS, *IBM, or *ALLUSR for the LIB parameter on the SAVLIB command or the SAVLIB parameter on the RSTLIB command.

- You should not start the SAVLIB command on the failed tape. The results of attempting the second save on the end of the tape can vary greatly, depending on what was previously on it and how the save library operation ended. The correct procedure is to begin again on a new tape, as if starting the complete save again.
- When you run the restore operation from the SAVLIB tape and an incomplete library is found, the object being restored is marked as damaged on the system. You can run the second restore operation without any recovery from the previously failed restore operation because the system deletes the object, if eligible, and then restores it.
- When you specify *NONSYS, *IBM, or *ALLUSR, the STRLIB parameter is shown on the prompt display. If the STRLIB parameter specifies a starting library, then message CPD3769 is sent to indicate you must specify *NONSYS, *IBM, or *ALLUSR on the LIB or SAVLIB parameter. If the STRLIB value is a system library (not allowed), message CPD3770 is sent to indicate that the STRLIB value is not valid. If the library does not exist, the SAVLIB or RSTLIB command ends with escape message CPF3818, indicating the starting library was not found, and no libraries were saved or restored.

The basic recovery steps for a save operation are:

1. Check the job log to determine the library where the previous SAVLIB LIB(*NONSYS, *IBM, or *ALLUSR) failed. Find the last library saved (successful completion message).
2. Determine the next library to be saved beyond the failure by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
          OUTPUT(*PRINT)
   ```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.
3. Load a new tape and type the following:

   ```
   SAVLIB LIB(*NONSYS, *IBM or *ALLUSR)
          STRLIB(library-name)
   ```

The basic recovery steps for a restore operation are:

1. If the STRLIB parameter was used on the SAVLIB command, determine the next library to be restored beyond the failure by displaying the first SAVLIB LIB(*NONSYS, *ALLUSR, or *IBM) tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). Load the next tape and enter the following:

   ```
   RSTLIB SAVLIB(*NONSYS, *ALLUSR, OR *IBM)
   DEV(TAPE01) ENDOPT(*LEAVE) STRLIB(starting-library-name)
   ```

2. If the STRLIB parameter was not used on the SAVLIB command, look at the job log to determine the library where the previous RSTLIB failed.

   a. Determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). The correct starting library is normally the second library listed after the last library restored successfully.

   b. Load the first tape of the SAVLIB media and type the following command to start the restore operation again.

```
RSTLIB SAVLIB(*NONSYS, *ALLUSER, OR *IBM)
       STRLIB(starting-library-name)
```

3. If the STRLIB value is not *FIRST and the starting library name is not the first library saved on the tapes, message CPA3704 is sent asking you to load the volume where the starting library exists.
4. Load the next tape and answer the message. A search for the starting library is done. If the library is not found on the loaded tape, and the library is expected to be found on the remaining tapes, message CPA3706 is sent requesting the next volume. If the starting library is not located on the current tape, and it is on a previous one, then message CPA3705 is sent asking you to load the previous volume. All of these inquiry messages are sent to the QSYSOPR message queue.
5. Once the starting library tape file is located on tape, the system begins restoring the tape.

Because the starting library for the *NONSYS, *IBM, or *ALLUSR save operation must be a library allowed by the SAVLIB or RSTLIB command, some libraries may not be specified for the STRLIB parameter. The following libraries *cannot* be specified for the STRLIB parameter:

- QDOC
- QRPLOBJ
- QSPL
- QSRV
- QSYS
- QTEMP
- QRECOVERY

If the library is not listed in the description on the first save tape, no tapes are searched for the starting library, no libraries are restored, and message CPF3818 is sent indicating that the starting library was not found.

Lock Conditions When Saving and Restoring Objects

The following reviews the object locking types:

- Exclusive (*EXCL). The object is reserved for the exclusive use of the requesting job; no other job can use the object. However, if the object is already allocated to another job, your job cannot get exclusive use of the object.
- Exclusive allow read (*EXCLRD). The object is allocated to the job that requested it, but other jobs can read the object. This lock is appropriate when you want to prevent any other job from doing any operation other than a read.
- Shared for update (*SHRUPD). The object can be shared either for update or read with another job. That is, another job can request either a shared-for-read lock state or a shared-for-update lock state for the same object. This lock state is appropriate when a job intends to change an object but wants to allow another job to read or change the same object.
- Shared no update (*SHRNUP). The object can be shared with another job if that job requests the object with either a shared-no-update or shared-for-read lock. This lock state is appropriate when you do not intend to have a job change an object but want to ensure that no other job changes the object.
- Shared for read (*SHRRD). The object can be shared with another job if the job does not request exclusive use of the object. That is, another job can request an exclusive-allow-read, shared-for-update, shared-for-read, or shared-no-update lock state.

Note: The address of a library does not restrict the operations that can be performed on the objects within the library. That is, if one job places an exclusive-allow-read or shared-for-update lock state on a library, other jobs can no longer place objects in or remove objects from the library; however, the other jobs can still update objects within the library. Table 9-1 on page 9-11 shows the valid lock state combinations for an object.

Table 9-1. Valid Lock State Combinations

If One Job Step Obtains This Lock State:	Another Job Step Can Obtain This Lock State: Exclusive	Exclusive Allow Read	Shared for Update	Shared No Update	Shared for Read
*EXCL					
*EXCLRD					X
*SHRUPD			X		X
*SHRNUP				X	X
*SHRRD		X	X	X	X

Locks on Objects during a Save Operation: You can save and restore objects while normal system operation continues. In general, a particular object is locked to prevent an update attempt while it is being saved, and locked to prevent a read attempt while it is being restored.

If an object is being updated, or if it is allocated *EXCL to another job, it cannot be saved or restored while it is being used, with the exception of journals and journal receivers. A journal or journal receiver can be active while it is being saved. In addition, an object cannot have its storage freed if it is being used in a job. Database files are locked in *SHRNUP when a save is specified. Another job can have the same file in *SHRNUP or *SHRRD lock state.

Some object types cannot be saved while they are in use. Job queues and output queues are locked *SHRRD while they are being saved. Other objects are locked *SHRNUP unless storage is to be freed, then the objects are locked *EXCL.

Most objects cannot be restored while they are in use. Job queues and output queues are locked *SHRRD. Message queues are locked *EXCLRD. Other objects are locked *EXCL. Restoring a program that is in use usually results in it ending abnormally because the old version (the one in use) is deleted when the new version is restored.

Lock Conditions on Libraries during Save and Restore Operations: You cannot do save and restore operations against the same library at the same time. An internal lock is placed on the library while a save or restore operation is performed. The following lock conditions exist against a library:

- *Restrictions for Libraries When Saving.* When a save library operation is in progress, the library is locked with a *SHRNUP lock. When a save object, save changed object, or restore object operation is in progress, the library is locked with a *SHRUPD lock. Library QUSRSYS is locked with *SHRRD for the SAVLIB command.
- *Restrictions for Libraries When Restoring.* When a restore library operation is in progress, the library is locked with an *EXCLRD lock. When a restore object operation is in progress, the library is locked with a *SHRUPD lock.

  Note: The use of a library does not restrict the operations that can be done on the objects within the library. That is, if one job places a *EXCLRD or SHRUPD lock state on a library, other jobs can no longer place objects in or remove objects from the library; however, the other jobs can still change the object in the library.

Save and Restore Operations Requiring a Restricted System

When doing the following tasks, you must have the system in a restricted state with all subsystems ended:

- Saving the entire system
- Saving storage
- Restoring all *IBM libraries
- Restoring user profiles
- Restoring authority
- Saving and restoring all *NONSYS libraries

The system is placed in the restricted state by the End System (ENDSYS) or End Subsystem (ENDSBS SBS(*ALL)) command.

System Performance during Save and Restore Operations

The following considerations influence performance:

- Size and activity level of the storage pool
- Run priority of the save or restore job
- Size of the system
- Size of save data
- Type of save data

- Number of objects saved
- Type of command
- Number of recoverable errors
- Number of diskettes or tapes used
- Model size
- System status (restricted or non-restricted state)
- Data rate of save and restore device
- Data compression (specified or not specified) provided by tape I/O adapter or programming
- Number of disk units on the system
- Special authority of user running operation

Save and Restore Operations: Save and Restore operations should run without the need for special tuning when it is run in a non-restricted state. Ensure that care has been taken to reduce adverse performance on other jobs while save and restore operations are running. However, performance can be tuned, if necessary, by changing one of the following:

- The run priority of the save or restore job. Priority is established through the job when the device file (diskette or tape) or save file is opened, and it can be changed using the Change Job (CHGJOB) command to run faster or slower. Changes made after the file is opened do not take effect until after the file is closed.
- The storage pool size or the activity level of the subsystem in which the job is running. The save and restore operation obtains buffers from the storage pool in which the save or restore job is running. The minimum storage pool size is 160KB (KB equals 1024 bytes) for diskette and save files and 224KB for tape.

  The amount of storage used as buffers appears in the *Reserved size* field on the Work with System Status display. You can adjust the amount and the effect on other operations by changing the storage pool size and the activity levels of the subsystem (using the Change Subsystem Description (CHGSBSD) command) in which the save or restore job is running.
- Using data compression and decompression. For more information about the effects of using data compression, see the topic "Data Compression and Decompression" on page 4-6.

Save System Special Authority: Performance is improved if the user profile running the operation has *SAVSYS special authority because no authority checking is necessary during the save and restore operation.

System Status: For either a save or restore operation, performance is improved if the operation is run when the system is in a restricted state (all subsystems ended). Performance is improved because the OS/400 licensed program does not have to lock objects and sufficient main storage exists to provide an improved save operation.

Effect on Users: Save and restore operations affect users of the system. The effect on other users can be reduced if you have more disk units on the system.

If your system is not in a restricted state when a high-speed tape unit is used, the performance degradation for other users is greater than if you are using diskettes because the tape unit has a higher data transfer rate.

Security Considerations When Saving and Restoring

Security considerations include ownership of objects and granting of public and private authorities.

Security for the Restore Commands: For systems in which security is especially important, only data or source files should be restored from other systems. Examination of the source, recompilation on the new system, and re-creation of objects (for example, job descriptions) are required to maintain tight security. If you attempt to restore an object whose owner on the medium does not match the owner on the system, then the object is not restored. This is the default for the restore commands. However, if the allow object differences (ALWOBJDIF(*ALL)) parameter is specified on the Restore command, the object on the medium is restored, and the owner on the system remains the owner of the object.

You can secure the restore commands and authorize them to specific user profiles. You can change this public authority except for the RSTUSRPRF, RSTAUT, and Restore Licensed Program (RSTLICPGM) commands, which are authorized only to QSECOFR.

Ownership: Every object on the AS/400 system must have an owner. The restore commands RSTOBJ, RSTUSRPRF, RSTLIB, RSTCFG, and RSTDLO, provide a parameter to allow object differences (ALWOBJDIF). This allows the restore operation to override any differences found in ownership. If there are differences in ownership and ALWOBJDIF(*ALL) is specified, then the system will restore the object to the owner on the system instead of the owner on the media.

The object owner also remains the same if the owner's user profile already exists on the system. This occurs whether or not the object exists on the system at the time of the restore operation.

Public and Private Authorities: The object on the media includes the object owner name and the public authority attributes. If the object exists on the system, the public authority of the system is used (not the authority on the media). If an object has public authority specified on one system and is restored to another system where the object does not exist, its public authority is also restored on the new system. The only exception is when programs adopt their owner's user profile (USRPRF (*OWNER) specified when the program was created).

Programs that adopt their owner's user profile must be restored by the owner or the QSECOFR user profile. If the owner of the program does not exist on the new system or the restore operation is running under a profile other than the owner or the security officer, then, as with other restore operations where the owner does not exist, the public authority for the object is changed to *EXCLUDE. For this reason, a program that adopts its owner's user profile (USRPRF(*OWNER) specified for the program) should be restored by the security officer or by the owner of the program.

Private authorities for an object are saved with the user profile. Private authorities for users other than the owner are not saved when the object or library is saved, but they are saved with the user profiles when saved with the SAVSYS or SAVSECDTA command. When an object is restored to a library where the object does not exist, the owner must grant private authorities again.

If the entire system is restored, two restore commands must be used in the following sequence. The authorization structure is reset to the point when the user profiles were saved.

- The Restore User Profile (RSTUSRPRF) command restores one or more user profiles. This command is normally used when restoring the entire system or when transferring a user from one AS/400 system to another.
- Make sure all of the objects owned by the restored user profiles are restored.
- The Restore Authority (RSTAUT) command grants the private authorities existing when the system was saved. This command does not revoke any private authorities granted since that time.

Authority Holders: When an authority holder exists on the system that has the same name as a file being restored, the restored file is linked to the authority holder. When a file is restored to a library other than the library containing the authority holder where the authority holder does not already exist, and where an authority holder secures the files named in that library, then the file will be restored only if the following conditions exist:

- The owner of the file has *ALL authority to the authority holder.
- If the file is a physical file, the owner had *ALL authority to the file when it was saved.
- If the file is a logical file, the owner has *OBJEXIST, *OBJMGT, and *OBJOPR authorities to the file when it was saved.

Authorization Lists: During a save operation, if an object is secured by an authorization list, the name of the authorization list is saved with the object.

The object is restored and linked to the authorization list if all of the following occur:

1. The object must be restored on the same system from which it was saved unless ALWOBJDIF(*ALL) is specified on the restore command. If ALWOBJDIF(*NONE) is specified, the object will be restored, but will be given *EXCLUDE public authority.
2. If the object (except documents and folders) already exists on the system, it must have the same authorization list as the object on the medium. If not, the object will not be restored.
3. The authorization list must exist on the system. If not, the object will be restored, but will be given *EXCLUDE public authority.

Media and Save File Security: Control of who can use the media device or save file, and for what purpose, is essential to maintain security of the system. The tape, diskette, or save file should be secured so that only authorized users have access to the media or save file.

Printing Security Information: If you specify OUTPUT(*PRINT) on the Restore Object (RSTOBJ), Restore Library (RSTLIB), or Restore Licensed Program (RSTLICPGM) commands, you obtain a list of what was restored. The list indicates the following:

- If a program adopted the owner's user profile.
- The associated user profile of a job description.
- If the object is restored and ownership is given to the default owner profile (QDFTOWN) or the system owner.
- The old owner's user profile.
- If authorization is revoked.

The LABEL Parameter on the Save and Restore Commands

Most save commands allow you to specify a file label up to seventeen characters in length to be associated with the file saved on tape or diskette. The default value for this parameter, either LABEL(*LIB) or LABEL(*GEN), results in the label being generated from the name of the library from which the objects are saved.

If you do not choose the default and enter a label of your own on the label parameter, you must enter that same label on the corresponding Restore command in order to restore the objects. For example, if you specified SAVOBJ DEV(TAP01) LABEL(MYLABEL), and you later want to restore the objects from the save tape, specify RSTOBJ DEV(TAP01) LABEL(MYLABEL). You cannot restore from the file using the default value of LABEL(*LIB) on the RSTOBJ command. This applies to all save and restore commands that support the LABEL parameter. If you do not enter the correct label, you receive the message File label mismatch' in your job log.

Using the Save and Restore Commands

The following subtopics show examples of the save and restore commands. The *CL Reference* has more information about these commands.

Entering the Save and Restore Commands

When you enter a save or restore command at a work station, the work station cannot be used until the command completes. To avoid this inconvenience, submit the command to a job queue and run the function in batch mode.

Checking Objects before Saving

Using the parameter PRECHK(*YES) ensures that all objects you intend to save can be saved. (For example, sometimes an object is not saved because it was locked for update at the time of the save operation.) Objects are verified to ensure that:

- They can be allocated during the save operation.
- Those specified on the save command exist.
- Those specified on the save command are not marked as damaged.
- All members of the database files can be allocated during the save operation.
- The person requesting the save operation is authorized to save all of the objects.

If any one of these conditions is not true and PRECHK(*YES) is specified, the save operation stops. When more than one library is specified, PRECHK(*YES) applies to each library independently. The failure to save one library does not affect the saving of other libraries.

If PRECHK(*NO) is specified, the save operation continues, but bypasses any object that cannot be saved. The command will complete with an escape message (see "Save and Restore Completion Messages" on page 9-6 for more information).

Summary of How to Save Object Types

Table 9-2 summarizes how objects of each type can be saved. It includes the variable specified on the OBJTYPE parameter and shows which commands are used with which objects.

Table 9-2 (Page 1 of 2). Objects Saved by Commands According to Object Type

Object	OBJTYPE	SAVOBJ	SAVLIB	SAVSECDTA	SAVSYS	SAVDLO
Alert table	*ALRTBL	X				
Authority holder	*AUTHLR			X[6]	X[6]	
Authority list	*AUTL			X[6]	X[6]	
Chart format	*CHTFMT	X	X		X[1]	
C local description	*CLD	X	X		X[1]	
Class	*CLS	X	X		X[1]	
Class of service description	*COSD				X[3]	
Command definition	*CMD	X	X		X[1]	
Configuration list[4]	*CFGL	X			X	
Controller description	*CTLD				X[3]	
Cross system product map	*CSPMAP	X	X		X[1]	
Cross system product table	*CSPTBL	X	X		X[1]	
Data area	*DTAARA	X	X		X[1]	
Data queue[2]	*DTAQ	X	X		X[1]	
Data dictionary	*DTADCT		X		X[1]	
Device description	*DEVD				X[3]	
Distribution objects	*DRX			X	X	
	*DTO			X	X	
	*DUO					X
Document	*DOC					X
Edit description[4]	*EDTD	X			X	
File[2, 5]	*FILE	X	X		X[1, 7]	
Folder	*FLR					X
Font resource	*FNTRSC	X	X		X[1]	
Forms control table	*FCT	X	X		X[1]	
Forms definition	*FORMDF	X	X		X[1]	
Graphics symbol set	*GSS	X	X		X[1]	
Job description	*JOBD	X	X		X[1]	
Job queue[2]	*JOBQ	X	X		X[1]	
Journal[2]	*JRN	X	X		X[1]	
Journal receiver	*JRNRCV	X	X		X[1]	
Library	*LIB		X[7]			
Line description	*LIND				X[3]	
Menu	*MENU	X	X		X[1]	
Message file	*MSGF	X	X		X[1]	
Message queue[2]	*MSGQ	X	X		X[1]	
Mode description	*MODD				X[3]	

Table 9-2 (Page 2 of 2). Objects Saved by Commands According to Object Type

Object	OBJTYPE	SAVOBJ	SAVLIB	SAVSECDTA	SAVSYS	SAVDLO
Output queue[2]	*OUTQ	X	X			
Overlay	*OVL	X	X		X[1]	
Page segment	*PAGSEG	X	X		X[1]	
Panel group	*PNLGRP	X	X		X[1]	
Printer description group	*PDG	X	X		X[1]	
Product availability	*PRDAVL	X	X		X[1]	
Program	*PGM	X	X		X[1]	
Query definition	*QRYDFN	X	X		X[1]	
Query form	*QMFORM	X	X		X[1]	
Query manager query	*QMQRY	X	X		X[1]	
Reference code translation table	*RCT	X	X		X[1]	
System/36 machine description	*S36	X	X		X[1]	
Search index	*SCHIDX	X	X		X[1]	
Session description	*SSND	X	X		X[1]	
Spelling help dictionary	*SPADCT	X	X		X[1]	
Subsystem description	*SBSD	X	X		X[1]	
Table	*TBL	X	X		X[1]	
User index	*USRIDX	X	X		X[1]	
User profile	*USRPRF			X[6]	X[6]	
User queue	*USRQ	X	X		X[1]	
User space	*USRSPC	X	X		X[1]	

1 If the object is in library QSYS.

2 For save files, only descriptions are saved when the SAVLIB command SAVFDTA(*NO) is specified. For other objects that have only their descriptions saved, see Table 5-2 on page 5-9.

3 These are restored using the RSTCFG command.

4 Edit descriptions and configuration lists reside only in library QSYS.

5 The SAVSAVFDTA command saves only the contents of save files.

6 User profiles are restored using the RSTUSRPRF command. Authorities are restored using the RSTAUT command after needed objects are restored. Authorization lists and authority holders are restored when RSTUSRPRF USRPRF(*ALL) is specified.

7 If there are save files in the library, the save file data is saved by default.

Save Library (SAVLIB) Command Examples

The following are examples of using the SAVLIB command.

- To save a single library to tape:

```
SAVLIB LIB(TOM)  DEV(TAP01) LABEL(080989)
```

- To save a library to diskette and free storage of all files, programs, and journal receivers after the save operation:

```
SAVLIB LIB(JOE) DEV(DKT01) VOL(VOLxx1)  STG(*FREE)
```

- To save all user libraries, library QGPL, and licensed program libraries to tape and perform data compression:

```
SAVLIB LIB(*NONSYS) DEV(TAP01) DTACPR(*YES)
```

- To save a library in a previous release format:

```
SAVLIB LIB(TOM) DEV(TAP01) TGTRLS(*PRV)
```

Restore Library (RSTLIB) Command Examples

The following are examples of using the RSTLIB command.

- To restore a single library:

```
RSTLIB SAVLIB(TOM)  DEV(TAP01) LABEL(080989)
```

- To restore all user libraries, library QGPL, and licensed program libraries from tape:

```
RSTLIB SAVLIB(*NONSYS) DEV(TAP01)
```

- To restore a single library to user ASP 9:

```
RSTLIB  SAVLIB(TOM) DEV(TAP01) RSTASP(9)
```

Note: When using the RSTASP parameter on the RSTLIB command, the library cannot already exist in a different ASP and the specified ASP cannot contain any isolated objects, such as journals, journal receivers, and save files whose library is in the system ASP.

When restoring libraries to a user ASP, all objects in the library are restored to the specified ASP. Objects that are not allowed in user ASPs are not restored. For a list of objects that are not allowed in a user ASP, see "Object Types Not Allowed in a User ASP" on page 14-10.

Save Object (SAVOBJ) Command Examples

The following are examples of using the SAVOBJ command.

- To save a single file:

```
SAVOBJ OBJ(DSTREF) LIB(DSTORDLB) DEV(DKT01) OBJTYPE(*FILE)
    VOL(SAVOL) STG(*KEEP)
```

  The file in library DSTORDLB remains there because the storage was not freed. It was saved on volume SAVOL, which was loaded in a diskette unit.

- To save a group of objects using a generic name:

```
SAVOBJ OBJ(ORD*) LIB(DSTORDLB) DEV(TAP01) OBJTYPE(*FILE)
    VOL(*MOUNTED) STG(*KEEP)  ENDOPT(*UNLOAD)
```

  The files are saved on magnetic tape on the volume loaded. Storage for these files is not freed and the objects still exist in library DSTORDLB. The tape will be rewound and unloaded from the tape unit as specified in the ENDOPT parameter.

- To save a group of objects in a previous release format using a generic name:

```
SAVOBJ OBJ(TOM*) LIB(TOM) DEV(TAP01) OBJTYPE(*FILE)
   VOL(*MOUNTED) STG(FREE) TGTRLS(*PRV) ENDOPT(*UNLOAD)
```

  All the files whose names begin with TOM in library TOM are saved to magnetic tape in a previous release format on the volume loaded. Storage for these files is freed and the files no longer exist in library TOM. The tape will be rewound and unloaded from the tape unit as specified in the ENDOPT parameter.

Restore Object (RSTOBJ) Command Examples

The following are examples of using the RSTOBJ command.

- To restore a single file:

```
RSTOBJ OBJ(DSTREF) DEV(DKT01) OBJTYPE(*FILE)
    VOL(SAVOL) OPTION(*OLD)
```

  This command restores the file to the library it was saved from. It was restored from volume SAVOL, which was loaded in a diskette unit.

- To restore a group of files using a generic name:

```
RSTOBJ OBJ(ORD*) LIB(DSTORDLB) DEV(TAP01) OBJTYPE(*FILE)
    VOL(*MOUNTED)
```

  This command restores all files beginning with the characters ORD.

- To restore a journal receiver to user ASP 5 when the library is in the system ASP:

```
RSTOBJ  OBJ(JRNRCV001) LIB(TOM) DEV(TAP01)
    OBJTYPE(*JRNRCV) VOL(*MOUNTED) RSTASP(5)
```

  Note: Only journals, journal receivers, and save files can be restored to an ASP that does not contain their libraries. This can only be done when the library is in the system ASP, and the ASP for the journals, journal receivers, and save files does not contain a library.

Save Changed Objects (SAVCHGOBJ) Command Example

The SAVCHGOBJ command is like the SAVOBJ command except that the SAVCHGOBJ command saves only *changed* objects or members. When an object is created or changed, the system time-stamps the object in an internal field, indicating a change has occurred.

The following example saves those changed objects that:

- Have names beginning with ORD
- Are files in library DSTPRODLB
- Have changed since 22 July 1988

```
SAVCHGOBJ OBJ(ORD*) LIB(DSTPRODLB) DEV(TAP01)
       OBJTYPE(*FILE) REFDATE(220788)
```

Example of Restoring Changed Objects after the RSTLIB Command: The following command restores all the user libraries.

```
RSTLIB SAVLIB(*NONSYS) DEV(TAP01)
```

The following is an example of restoring changed files to the user libraries that were previously restored.

```
RSTOBJ OBJ(*ALL) DEV(TAP01) OBJTYPE(*FILE)
       OPTION(*ALL) MBROPT(*ALL) ALWOBJDIF(*ALL)
```

Examples of Saving Access Paths: The following are examples of saving access paths using the ACCPTH parameter on the Save Object (SAVOBJ) command.

- To save access paths by specifying a generic name:

  ```
  SAVOBJ OBJ(ORD*) LIB(DSTPRDLB) OBJTYPE(*FILE)
      ACCPTH(*YES)
  ```

  This command saves the access paths for all database files beginning with the characters ORD.

- To save all access paths for a library:

  ```
  SAVLIB LIB(DSTPRDLB)  ACCPTH(*YES)
  ```

  This command saves the access paths for library DSTPRDLB.

When you specify ACCPTH(*YES), a message is issued for each physical file saved indicating how many logical file access paths were saved with that physical file.

Examples of Restoring Access Paths: The following is an example of restoring access paths if the logical file does not exist on the system or is in different library.

1. Restore the physical file first:

   ```
   RSTOBJ OBJ(PAYLIB/PAYFILE) SAVLIB(MYLIB) DEV(TAP01)
          OBJTYPE(*FILE) VOL(*MOUNTED) SEQNBR(*SEARCH)
   ```

2. Remove the physical file members:

   ```
   RMVM FILE(PAYLIB/PAYFILE) MBR(*ALL)
   ```

3. Restore the logical file:

   ```
   RSTOBJ  OBJ(PAYLIB/PAYLF) SAVLIB(MYLIB) DEV(TAP01)
           OBJTYPE(*FILE) VOL(*MOUNTED) SEQNBR(*SEARCH)
   ```

4. Restore the physical files again:

   ```
   RSTOBJ OBJ(PAYLIB/PAYFILE) SAVLIB(MYLIB) DEV(TAP01)
          OBJTYPE(*FILE) VOL(*MOUNTED) SEQNBR(*SEARCH)
   ```

When you specify ACCPTH(*YES), a message is issued for each physical file saved indicating how many logical file access paths were saved with that physical file.

Examples of Saving a List of Members: In the following example, DBFA and DBFB are database files with members P1, P2, M1, and M2.

- To save a single, specific member from a specific database file:

  ```
  SAVOBJ OBJ(DBFA) FILEMBR(DBFA (M1)) DEV(TAP01)
      LIB(DSTPRODLB)  OBJTYPE(*FILE)
  ```

  Member M1 from file DBFA will be saved.

- To save a generic set of members from a specific database file:

  ```
  SAVOBJ OBJ(DBFA) FILEMBR(DBFA (P*)) DEV(TAP01)
      LIB(DSTPRODLB)  OBJTYPE(*FILE)
  ```

  Members P1 and P2 from file DBFA will be saved.

- To save a specific set of members from a specific database file:

  ```
  SAVOBJ    OBJ(DBFA) FILEMBR(DBFA (M1 M2 P1)) DEV(TAP01)
       LIB(DSTPRODLB)  OBJTYPE(*FILE)
  ```

  Members M1, M2, and P1 from file DBFA will be saved.

- To save a specific set of members from several database files:

  ```
  SAVOBJ OBJ(DBFA DBFB)  FILEMBR(*ALL (M1 P1 P2))
      DEV(TAP01) LIB(DSTPRODLB) OBJTYPE(*FILE)
  ```

  Members M1, P1, and P2 from both file DBFA and DBFB will be saved.

- In the following example, *NONE indicates that only the file description is saved:

  ```
  SAVOBJ OBJ(DBFA) FILEMBR(DBFA(*NONE)) DEV(TAP01)
      LIB(DSTPRODLB)  OBJTYPE(*FILE) MBROPT(*ALL)
  ```

  None of the members for file DBFA are saved; only the file descriptions are saved.

Examples of Restoring a List of Members: You can restore a list of members for a database file using the FILEMBR parameter of the RSTOBJ command. In the following example, DBFA and DBFB are database files with members P1, P2, M1, and M2.

- To restore a single, specific member from a specific database file:

  ```
  RSTOBJ OBJ(DBFA) FILEMBR(DBFA (M1)) DEV(TAP01)
      SAVLIB(DSTPRODLB)  OBJTYPE(*FILE) MBROPT(*ALL)
  ```

  Member M1 from file DBFA will be restored.

- To restore a generic set of members from a specific database file:

```
RSTOBJ OBJ(DBFA) FILEMBR(DBFA (P*)) DEV(TAP01)
    SAVLIB(DSTPRODLB)  OBJTYPE(*FILE) MBROPT(*ALL)
```

  Members P1 and P2 from file DBFA will be restored.

- To restore a specific set of members from a specific database file:

```
RSTOBJ OBJ(DBFA) FILEMBR(DBFA (M1 M2 P1)) DEV(TAP01)
    SAVLIB(DSTPRODLB)  OBJTYPE(*FILE) MBROPT(*ALL)
```

  Members M1, M2, and P1 from file DBFA will be restored.

- To restore a specific set of members from several database files:

```
RSTOBJ OBJ(DBFA DBFB)  FILEMBR(*ALL (M1 P1 P2))
DEV(TAP01) SAVLIB(DSTPRODLB) OBJTYPE(*FILE) MBROPT(*ALL)
```

  Members M1, P1, and P2 from both files DBFA and DBFB will be restored.

- In the following example, *NONE indicates that only the file description is restored:

```
RSTOBJ OBJ(DBFA) FILEMBR(DBFA (*NONE)) DEV(TAP01)
    SAVLIB(DSTPRODLB)  OBJTYPE(*FILE) MBROPT(*ALL)
```

  None of the members for file DBFA will be restored, only the file descriptions.

Save Save File Data (SAVSAVFDTA) Command Examples

Only one library can be saved to a save file.

- When objects are saved to a save file, they can be saved to tape or diskette using the following command:

```
SAVSAVFDTA  SAVF(ONLINE) DEV(TAP01) SEQNBR(1) CLEAR(*ALL)
```

  This command saves the contents of the save file ONLINE to the first file on the tape device TAP01. Files that have not ended on either the first volume or following volumes are overwritten without any inquiry message because CLEAR(*ALL) is specified.

  When the data in a save file is saved to tape or diskette, the data can be restored as though it were coming from tape or diskette.

- If the last save command to save library USERLIB and the save file was ONLINE as used in the previous examples, then the following command must be specified to restore the library from tape.

```
RSTLIB LIB(USRLIB) DEV(TAP01) VOL(*MOUNTED)
```

Save Document Library Object (SAVDLO) Command Examples

The following examples are some possible uses of the SAVDLO command.

- To save a document within a folder:

```
SAVDLO DLO(A) DEV(device-name) FLR(X)
```

  This command saves document A within folder X.

  Note: The DEV parameter is where you would specify the diskette unit name, the tape device name, or *SAVF.

- To save all documents and folders within a folder:

```
SAVDLO DLO(*ALL) FLR(X/Z) DEV(device-name)
```

  This command saves all folders and documents within folder X/Z.

- To save all documents and folders:

```
SAVDLO DLO(*ALL) FLR(*ANY) DEV(device-name)
```

  This command saves all folders and documents on the system. You must have SAVSYS or all object authority to use the SAVDLO command this way.

- To save all documents not stored in folders:

```
SAVDLO DLO(*ALL) FLR(*NONE) DEV(device-name)
```

  This command saves all documents that are not stored within folders on the system.

- To save documents for a specific owner:

```
SAVDLO DLO(*SEARCH) DEV(device-name) OWNER(XYZ)
```

  This command saves all file documents owned by XYZ.

- To save documents that are created between specific dates.

```
SAVDLO DLO(*SEARCH) DEV(device-name)
    CRTDATE((*AVAIL '02/10/89' *AVAIL '03/10/89'))
```

  This command saves all filed documents created between February 10, 1989 and March 10, 1989 (inclusively).

- To save mail objects:

```
SAVDLO DLO(*MAIL)
```

  This command saves all the documents that are filed and unfiled that are referenced by a distribution object in addition to all filed documents in library QDOC that are referenced by a distribution.

- To save filed documents:

Restore

```
SAVDLO DLO(*SEARCH) DEV(device-name)
REFCHGDATE('03/10/89')
```

This command saves all the filed documents changed after March 10, 1989.

- To save a list of documents:

```
SAVDLO DLO(*DOCL) FLR(Y) DEV(device-name) DOCL(X)
       OUTPUT(*OUTFILE)  OUTFILE(MYFILE)
```

This command saves a list of documents specified by document list X in folder Y and writes information about the saved files to output file MYFILE.

- To save a list of documents in a previous release format:

```
SAVDLO DLO(A) FLR(X) DEV(device-name) TGTRLS(*PRV)
```

This command saves document A in folder X in a previous release format.

- To save documents and folders by specifying the type of search:

```
SAVDLO  DLO(*SEARCH) SRCHTYPE(*ALL)
     CRTDATE((*AVAIL '01/01/90' *AVAIL '03/01/90'))
     REFCHGDATE('01/01/90') OWNER(*ALL)
```

This command saves all folders (and all documents and folders within them) created between January 1, 1990 and March 1, 1990 (inclusively), in addition to all documents that were created or changed after January 1, 1990.

For more information on the SAVDLO command, see the online information for this command.

When you use the SAVDLO command to save documents and folders in a save file, the save file can be saved to diskette or tape using the SAVSAVFDTA command. The documents and folders can then be restored using the RSTDLO command directly.

Restore Document Library Object (RSTDLO) Command Examples

The following examples show some possible uses of the RSTDLO (Restore Document Library Object) command and describe what it does.

- `RSTDLO DLO(A) SAVFLR(X) DEV(device-name)`

This command restores document A from within folder X to folder X.

Note: The DEV parameter is where you specify the diskette unit name, the tape device name, or *SAVF.

- `RSTDLO DLO(*ALL) SAVFLR(A) DEV(device-name)`

This command restores all of folder A (including folders in folder A). If folder A is on the system when you specify the RSTDLO command and new documents are added to this folder, the new documents are merged in with the old documents from the restored folder A.

- `RSTDLO DLO(A,B,C,D) SAVFLR(X/Y) DEV(device-name) RSTFLR(P/Q)`

This command restores documents A, B, C, and D from folder X/Y into a different folder named P/Q.

- `RSTDLO DLO(A,B,C,D) SAVFLR(X/Y) DEV(device-name) NEWOBJ(*NEW)`

This command restores documents A, B, C, and D from folder X/Y to folder X/Y, and generates new system object names for A, B, C, and D.

- To restore distribution documents:

```
RSTDLO DLO(*MAIL) DEV(device-name)
```

This command restores all filed and unfiled documents that are referenced by distributions at the time they were saved, in addition to all saved distribution documents.

- To restore document A in folder Z:

```
RSTDLO  DLO(A) SAVFLR(Z) DEV(device-name)
     ALWOBJDIF(*ALL)
```

This command restores document A, saved in folder Z on the medium, into folder Z on the system, even if the owner and system object names for document A on the system is not the same as the owner and system object names on the medium. The owner and system object name on the system are kept.

For more information on the RSTDLO command, see the online information for this command.

Save System (SAVSYS) Command Examples

The following are examples of using the SAVSYS command.

- To save the system to tape:

```
SAVSYS DEV(TAP01) CLEAR(*ALL)
```

This command saves the system objects, all user profiles (including private authority for objects), licensed internal code, and configuration objects. They are saved on the tape loaded in tape unit TAP01. Each tape used in the save operation that is not cleared is automatically cleared, and the save operation continues without operator intervention.

- To save the system using a specific volume:

```
SAVSYS DEV(TAP01) VOL(SAVA02)
```

This command saves the system data on the TAP01 tape unit, starting with the tape volume labeled SAVA02. If the save operation exceeds the storage capacity of one tape, a message is sent to the operator requesting another volume be loaded in tape unit TAP01.

- To save the system using two tape units:

```
SAVSYS DEV(TAP01 TAP02)
```

This command saves the system data on tape units TAP01 and TAP02 in alternating order. If the save operation exceeds the storage capacity of two tapes, a message is sent to the operator requesting another volume be loaded in tape unit TAP01. The tapes rewind at the end of the save operation.

Save Security Data (SAVSECDTA) Command Examples

The following are examples of using the SAVSECDTA command.

- To save distribution objects associated with user profiles:

```
SAVSECDTA  DEV(TAP01) MAIL(*YES)
```

This command saves the user profiles, authorization lists, authority holders, and distribution (mail) objects to tape.

- To save the security data to a save file:

```
SAVSECDTA  DEV(*SAVF) SAVF(ONLINE) MAIL(*NO)
```

This command saves the user profiles, authorization lists, and authority holders to the ONLINE save file.

Restore User Profile (RSTUSRPRF) Command Examples

The following are examples of using the RSTUSRPRF command.

- To restore all user profiles, authorization lists, authority holders, and distribution objects currently on tape:

```
RSTUSRPRF DEV(TAP01) SEQNBR(*SEARCH) ENDOPT(*REWIND)
          MAIL(*YES)
```

This command restores all user profiles saved on the tape loaded on tape unit TAP01 to the system. The file that contains the user profiles is searched for, and the tape is rewound at the end of the restore operation. Distribution objects are also restored.

- To restore specific user profiles and also specify a generic name:

```
RSTUSRPRF DEV(TAP01) USRPRF(BURK BROWNS USER*)
```

This command restores from tape unit TAP01 user profiles BURK, BROWNS, and all other user profiles beginning with the characters USER. The user profiles must exist on the tape loaded on tape unit TAP01.

Restore Authority (RSTAUT) Command Examples

The following are examples of using the RSTAUT command:

- To restore authority to each user profile restored:

```
RSTAUT
```

This command restores to each user profile the authority to each object that the user profile had at the time the system was saved. The user profiles and the associated libraries and objects must be restored before the RSTAUT command is run.

- To restore specific user profiles and their authorities:

```
RSTUSRPRF USRPRF(BURK SMITHP THOMPSON SANDERS)

RSTLIB  LIB(USERLIB)

RSTAUT USRPRF(BURK SMITHP THOMPSON)
```

To each user profile that was successfully restored, the RSTAUT command restores authority to each object that the user profile had at the time the security data or system was saved. The user profile and the associated objects and libraries must be restored before the RSTAUT command is run. Because SANDERS was not specified on the RSTAUT command, the authorities are still available and can be restored at a later date.

- To restore all user profiles and their authorities from the media:

```
RSTUSRPRF USRPRF(*ALL)
RSTLIB  LIB(USERLIBA)

RSTLIB  LIB(USERLIBB)

RSTLIB  LIB(USERLIBC)

RSTAUT USRPRF(*ALL)
```

The RSTAUT command restores private authorities for all user profiles restored by the RSTUSRPRF commands.

Restore Configuration (RSTCFG) Command Examples

The following are examples of using the RSTCFG command.

- To restore all device configuration objects:

  ```
  RSTCFG DEV(TAP01) OBJ(*ALL) OBJTYPE(*ALL)
  ```

  This command restores all the device configuration objects from the tape loaded on tape unit TAP01.

- To restore a specific device configuration object:

  ```
  RSTCFG DEV(TAP01) OBJ(PRT01) VOL(SAVA02) OBJTYPE(*DEVD)
  ```

  This command restores the device description PRT01 that was saved on tape volume SAVA02 to the system. If the device description PRT01 already exists on the system, it must be varied off before it can be restored.

Save Licensed Program (SAVLICPGM) Command Examples

The following are examples of using the SAVLICPGM command. An option also exists on the Work with Licensed Programs menu that allows you to save licensed programs from a list.

- To save an individual licensed program:

  ```
  SAVLICPGM LICPGM(5738WP1) DEV(TAP01) CLEAR(*ALL)
  ```

  This command saves the Office licensed program on the tape put on tape unit TAP01. Each tape that is not cleared is automatically cleared during the save operation, and the save operation continues without operator intervention.

- To save a licensed program to a specific tape volume:

  ```
  SAVLICPGM LICPGM(5738WP1) DEV(TAP01) VOL(SAVA02)
  ```

  This command saves the Office licensed program on the tape loaded on tape unit TAP01, starting with the tape volume labeled SAVA02. If the save operation exceeds the storage capacity of the tape, a message is sent to the operator asking that the next tape volume be loaded on tape unit TAP01.

- To save the online help option:

  ```
  SAVLICPGM LICPGM(5738SS1) OPTION(2) DEV(TAP01 TAP02)
  ```

  This command saves the online help option of the OS/400 licensed program to tape units TAP01 and TAP02 in alternating order. If the save operation exceeds the storage capacities of the tapes, a message is sent to the operator asking that the another tape be loaded in tape unit TAP01. The tape rewinds when the save operation completes.

Restore Licensed Program (RSTLICPGM) Command Example

The following is an example of using the RSTLICPGM. An option also exists on the Work with Licensed Programs menu that allows you to restore licensed programs from a list.

- To restore a licensed program:

  ```
  RSTLICPGM LICPGM(5738SS1) OPTION(2) DEV(TAP01)
  ```

  This command restores the online help option of the OS/400 licensed program from tape unit TAP01. The licensed program objects must exist on the tape loaded in tape unit TAP01. Because no other parameters are specified on this command, the defaults are used.

Example of Saving Spooled Output Files Using the Copy Spooled File (CPYSPLF) Command

The system does not directly support saving spooled files on an output queue, but you can accomplish the same thing by following these steps:

1. To save spooled output files:

   ```
   CPYSPLF  FILE(QPRINT) JOB(PAYROLL01) SPLNBR(4)
            TOFILE(MYLIB/SP4) TOMBR(MYMBR) CTLCHAR(*FCFC)
   ```

2. Save the database file using a save command such as the SAVOBJ, SAVLIB, or SAVCHGOBJ command:

   ```
   SAVOBJ OBJ(MYLIB/SP1) OBJTYPE(*FILE) DEV(TAP01)
   ```

This approach saves the textual data in the file. However, advanced functions such as graphics and variable fonts are not saved.

Example of Restoring Spooled Output Files Using the Copy File (CPYF) Command

If you used the CPYSPLF command to save spooled files on an output queue, you can then restore the spooled files using the following procedure:

1. Restore the database file:

```
RSTOBJ  OBJ(MYLIB/SP4) DEV(TAP01)
```

2. Copy the database file member back to spooled output files:

```
OVRPRTF FILE(QSYSPRT CTLCHAR(*FCFC) CHLVAL(*NORMAL)

CPYF  FROMFILE(MYLIB/SP4) TOFILE(QSYSPRT)
```

This approach restores SP4 to library MYLIB, then prints the file using QSYSPRT.

Chapter 10. Working with the Save and Restore Procedures

This chapter describes the following:

- Saving security data
- Saving the entire system
- Installing or restoring the licensed internal code
- Restoring the operating system
- Restoring user profiles, device configuration objects, user libraries and authorities
- Restoring program temporary fixes (PTF)

Restore

Before you begin any save procedure, it is recommended that you initialize enough tapes to complete the save operation. The number of tapes required depends on the size of the system, the number of objects being saved and the capacity of the tape. Use Table 4-2 on page 4-4 to determine tape capacities.

Initialize at least three more tapes than you think you will need. Each tape should have a volume ID and an external label that allows you to easily identify the tape.

An external label on the tape should identify the date and time of the save operation and what was saved. If an object needs to be restored, the tape that contains the object can be located quickly. A backup log should be provided to write down important information.

To initialize tapes that are new, do the following:

```
INZTAP DEV(TAP01) NEWVOL(BACKUP) NEWOWNID(user-profile-name)
       CHECK(*NO) ENDOPT(*UNLOAD)
```

To initialize tapes that have data on them, display the tape to verify that you do not want to keep the data; then do the following:

```
INZTAP DEV(TAP01) NEWVOL(BACKUP) NEWOWNID(user-profile-name)
       CHECK(*NO) ENDOPT(*UNLOAD)
```

If you are using tapes for the save operation that have data on them, you can use the CLEAR parameter to specify that data be cleared from the tape during the save operation. For example:

```
SAVSYS  DEV(TAP01) VOL(*MOUNTED) CLEAR(*ALL)
```

For more information about using tapes, see the topic Chapter 4, "Save and Restore Media Considerations" on page 4-1.

Using the Save and Restore Menus

The options on the Save and Restore menus allow you to perform save and restore operations without entering the commands. The following provides information about the save and restore menus and the options used in this guide.

Save Menu Options

Figure 10-1 and Figure 10-2 on page 10-4 show the options available on the save menu.

```
SAVE                                Save
                                                       System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Changed objects only
     7. Licensed programs
     8. Security data
     9. Storage

    20. All libraries other than system library
    21. Entire system
    22. All IBM libraries other than system library
                                                                More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
     (C) COPYRIGHT IBM CORP. 1980, 1991.
```

Figure 10-1. Save Menu, Part 1

Options 1 through 9: You will be prompted with the appropriate command.

Option 20. All libraries other than system library

- This option will:
  - End all subsystems
  - Save all libraries other than the system library
  - Start the controlling subsystem
- This option will not:
  - Save the contents of job queues, output queues, or data queues that exist on the system
  - Save libraries QSYS, QDOC, QSPL, QTEMP, QSRV, QRECOVERY, and QRPLOBJ

Option 21. Entire system

- This option will:
  - End all subsystems
  - Save the licensed internal code
  - Save the operating system
  - Save the security data
  - Save the device configuration objects
  - Save all user libraries (including libraries for licensed programs)
  - Save all documents and folders

- Save all distribution and mail objects
- Start the controlling subsystem

- This option will not:
  - Save the contents of job queues, output queues, or data queues that exist on the system

Note: To prevent an interrupted save caused by incomplete save messages, run the following commands before selecting option 21 from the Save menu:

1. To display the reply list sequence numbers currently used, type the following and press the Enter key.

   ```
   WRKRPYLE
   ```

2. To add a reply list entry, type the following (where xxxx is an unused sequence number 1-9999) and press the Enter key.

   ```
   ADDRPYLE SEQNBR(xxxx) MSGID(CPA3708) RPY('G')
   ```

3. To change the job, type the following and press the Enter key.

   ```
   CHGJOB INQMSGRPY(*SYSRPYL)
   ```

4. To change the QSYSOPR message queue, type the following and press the Enter key.

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(95)
   ```

You can replace the system programs that save the entire system with your own CL programs to tailor your system's save procedure. The menus currently call the following programs:

QMNSRBND	Determines if you are running in the current controlling subsystem and asks you for the ENDSBS command.
QMNSAVE	Asks you for the SAVSYS, SAVLIB LIB(*NONSYS) and SAVDLO DLO(*ALL) commands.

Option 22. All IBM libraries other than system library

- This option will:
  - Save all the IBM-supplied libraries that contain only IBM objects and user defined libraries that begin with a Q. See Table 5-1 on page 5-3.
- This option will not:
  - Save user profiles
  - Save device configuration objects
  - Save the system library (QSYS)
  - Save documents or folders

Restore

```
SAVE                                   Save
                                                          System:   RCHAS791
Select one of the following:

     23. All user libraries
     24. All changed objects in user libraries

     50. Save in System/36 format

     70. Related commands
```

Figure 10-2. Save Menu, Part 2

Option 23. All user libraries

- This option will:
  - Save all libraries that contain user data.
- This option will not:
  - Save the IBM-supplied libraries and user libraries beginning with a Q. See Table 5-1 on page 5-3.

Option 24. All changed objects in user libraries

- This option will:
  - Save all changed objects in all user libraries.
- This option will not:
  - Save the IBM-supplied libraries

When you select option 20 (All libraries other than system library) or option 21 (Entire system) on the save menu, the following display is shown.

```
                          Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .    TAP01         Names
                                    _____
                                    _____
                                    _____

  Prompt for commands  . . . . .    Y             Y=Yes, N=No

  Check for active files . . . .    Y             Y=Yes, N=No
```

Tape devices: You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device when the first tape is full.

Prompt for commands: You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting (for unattended save operations) and uses the default values.

Check for active files: Allows you to specify whether or not you want to check for active files. If you specify Y=Yes, the system sends a message when active files are encountered. You can end the checking process or clear the existing files and continue. If N=No is specified, all active files encountered during the save are cleared.

Using the Restore Menu

Figure 10-3 shows the options available on the restore menu.

```
RESTORE                            Restore
                                                        System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Licensed programs

    20. All libraries other than system library
    21. The system
    22. All IBM libraries other than system library
    23. All user libraries

    50. Restore from System/36 format
                                                                  More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

Figure 10-3. Restore Menu

Options 1 through 6: You will be prompted with the appropriate command.

Option 20. All libraries other than system library

- This option will:
  - End all subsystems
  - Restore the data that was saved by option 20 (All libraries other than system library)
  - Start the controlling subsystem
- This option will not:
  - Restore library QSYS
  - Restore user profiles
  - Restore authorities

Option 21. The system

This option is intended for use immediately after an install of the licensed internal code and OS/400 in a recovery scenario.

- This option will:
  - End all subsystems

  - Restore the security data
  - Restore device configuration objects
  - Restore all user libraries
  - Restore documents and folders
  - Restore distribution and mail objects
  - Start the controlling subsystem
- This option will not:
  - Restore library QSYS
  - Restore any line, controller, or device descriptions that are not varied off

Note: To prevent an interrupted restore caused by incomplete restore messages, run the following commands before selecting option 21 from the Restore menu:

1. To display the reply list sequence numbers currently being used, type the following and press the Enter key.

   ```
   WRKRPYLE
   ```

2. To add a reply list entry, type the following (where xxxx is an unused sequence number 1-9999) and press the Enter key.

   ```
   ADDRPYLE SEQNBR(xxxx) MSGID(CPA3709) RPY('G')
   ```

3. To change the job, type the following and press the Enter key.

   ```
   CHGJOB INQMSGRPY(*SYSRPYL)
   ```

4. To change the QSYSOPR message queue, type the following and press the Enter key.

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(95)
   ```

Option 22. All IBM libraries other than system library

- This option will:
  - Restore all the IBM-supplied libraries that contain only IBM objects and user-defined libraries that begin with a Q.
- This option will not:
  - Restore user profiles
  - Restore device configuration objects
  - Restore the system library (QSYS)
  - Restore documents or folders

Option 23. All user libraries

- This option will:
  - Restore all libraries that contain user data (except user-defined libraries that begin with a Q).
- This option will not:
  - Restore the IBM-supplied libraries including user-defined libraries beginning with a Q.

When you select option 20 (All libraries other than system library) or 21 (The system), the following display is shown.

```
                         Specify Command Defaults

 Type choices, press Enter.

   Tape devices . . . . . . . . .  : TAP01              Names
                                     _____
                                     _____
                                     _____

   Prompt for commands  . . . . .  :   Y                Y=Yes, N=No









 F3=Exit   F12=Cancel
```

Tape devices: You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device after the first tape is read.

Prompt for commands: You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting and uses the default values.

Note: If the saved data is contained in more than one file with folders and documents (for example, more than 32 000 objects if saved with Version 2 Release 1), you must prompt for the commands. On the RSTDLO command, you must specify beginning and ending sequence numbers. Otherwise, only the first file with folders and documents will be restored.

Saving the Security Data

You can use this procedure without the system being in a restricted state.

To save the security data, do the following:

1. To change the system operator message queue so all messages will appear on the display, type the following and press the Enter key.

   `CHGMSGQ QSYSOPR *BREAK SEV(60)`

2. Load the first tape, and make the tape device ready.
3. Save the security data:

   If you are saving distribution (mail) objects, type the following and press the Enter key.

   `SAVSECDTA DEV(TAP01) MAIL(*YES)`

If one or more tapes have not been initialized, a message similar to the following may appear:

```
Volume on device TAP01 wrong type (C INZ R)
```

Enter INZ and press the Enter key to initialize the tape.

4. When the following message appears, load the next tape, make the device ready, and then enter R.

```
Device was not ready or next volume not loaded (C R)
```

You have completed saving the security data. See the topic "Restoring User Profiles, Device Configurations, User Libraries, and Authority" on page 10-33 for the procedure to restore the security data.

Saving the Entire System

Task Overview

Use the following procedure to save the entire system. Saving the entire system includes:

- Licensed internal code
- Operating system
- Security data
- Device configuration objects
- All libraries (including libraries for licensed programs)
- All documents and folders
- All distribution and mail objects

Note: The contents of output queues, job queues, data queues, and message queues are not saved with the SAVLIB LIB(*NONSYS) command. Only their descriptions are saved. The contents of save files are not saved if you specify SAVFDTA(*NO) on the save command. The default for the SAVFDTA parameter is *YES.

If you have data in a save file that must be saved, you can also use the Save Save File Data (SAVSAVFDTA) command to save the data to tape or diskette.

When you are moving units from the system ASP to another ASP, and you have application programs that are dependent on spooled output files, it is recommended that you save the spooled output files. To save spooled output files, use the Copy Spooled File (CPYSPLF) command. An example of how to save spooled output files can be found in the topic "Example of Saving Spooled Output Files Using the Copy Spooled File (CPYSPLF) Command" on page 9-22.

Before You Begin . . .

- ☐ You must have save system (*SAVSYS) special authority to save the entire system.
- ☐ Correctly initialize the tapes as standard-labeled tapes.
- ☐ Clean the read and write head of the tape unit.
- ☐ If you are going to use the SAVSYS tapes to restore the entire system, the SAVSYS operation must be done using a tape device compatible (same density type) with the IPL tape device. If your IPL tape device cannot read the SAVSYS tapes, you may not be able to restore the licensed internal code using the SAVSYS tapes.
- ☐ If you are saving the entire system to prepare for a system upgrade, and you are using mirrored protection, you must print the current level of mirrored protection on your system. The service representative will need this information in case the level of mirrored protection is reduced by the model upgrade. To print the level of protection, do the following:
  1. Type STRSST on a command line and press the Enter key. You must have authority to use this command.
  2. Select option 3 (Work with disk units) on the System Service Tools (SST) display.
  3. Select option 1 (Display disk configuration) on the Work with Disk Units display.
  4. Select option 3 (Display disk configuration protection) on the Display Disk Configuration display.
  5. Print the displays (there may be several) using the PRINT key for each display. Keep the output with your save tapes. Your service representative will need them when he performs a model upgrade.

There are two methods you can use to save the entire system:

1. Using option 21 (Entire system) on the Save menu allows you to save the entire system without entering the commands.
2. Using the Save commands allows you to save the entire system by entering the commands from the command line.

Method 1. Using Option 21 (Entire system) on the Save Menu

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:

   ```
   WRKACTJOB
   ```
3. Display the system log QHST to verify it is up to date.

   ```
   DSPLOG  LOG(QHST)
   ```

   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:

   ```
   WRKF FILE(QSYS/QHST*)
   ```

   Look at the list to verify that you saved all copies of the log that will be needed later.

5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.

6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)

7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)

   Keep this list with your backup log or your save system tapes for future reference.

8. Go to the Save menu:

   GO SAVE

   The Save menu is shown.

```
SAVE                                    Save
                                                           System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Changed objects only
     7. Licensed programs
     8. Security data
     9. Storage

    20. All libraries other than system library
    21. Entire system
    22. All IBM libraries other than system library
                                                                      More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
     (C) COPYRIGHT IBM CORP. 1980, 1991.
```

Note: To prevent an interrupted save caused by incomplete save messages, run the following commands before selecting option 21 from the Save menu:

a. To display the reply list sequence numbers currently used, type the following and press the Enter key.

   WRKRPYLE

b. To add a reply list entry, type the following (where xxxx is an unused sequence number 1-9999) and press the Enter key.

   ADDRPYLE SEQNBR(xxxx) MSGID(CPA3708) RPY('G')

c. To change the job, type the following and press the Enter key.

   CHGJOB INQMSGRPY(*SYSRPYL)

d. To change the QSYSOPR message queue, type the following and press the Enter key.

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(95)
```

9. Select option 21 (Entire system) from the Save menu and press the Enter key.

```
                         Specify Command Defaults

 Type choices, press Enter.

   Tape devices . . . . . . . . .   TAP01        Names
                                    ______
                                    ______
                                    ______

   Prompt for commands  . . . . .   Y            Y=Yes, N=No

   Check for active files . . . .   Y            Y=Yes, N=No
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device when the first tape is full.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting (for unattended save operations) and uses the default values.

Check for active files

Allows you to specify whether or not you want to check for active files. If you specify Y=Yes, the system sends a message when active files are encountered. You can end the checking process or clear the existing files and continue. If N=No is specified, all active files encountered during the save are cleared.

Option 21 will guide you through the following if you selected Y on the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS SBS(*ALL) OPTION(*IMMED)
b. SAVSYS
c. SAVLIB LIB(*NONSYS) ACCPTH(*YES)
d. SAVDLO DLO(*ALL) FLR(*ANY)
e. STRSBS SBSD(controlling-subsystem)

If you want to be notified when the subsystems are ended, change the QSYSOPR message queue by typing the following and pressing the Enter key.

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
```

Messages are sent to the QSYSOPR message queue indicating when the subsystems have ended and the system is in a restricted state.

10. Continue loading tapes when the system sends a message asking you to load the next volume.

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous SAVLIB LIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be saved beyond the failure by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
          OUTPUT(*PRINT)
   ```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.
3. Load a new tape and type the following:

   ```
   SAVLIB LIB(*NONSYS) DEV(*TAP01) ENDOPT(*LEAVE)
            STRLIB(library-name) ACCPTH(*YES)
   ```

   Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

Method 2. Using the Save commands

If you do not want to use option 21, you can do the following steps from the command line of a menu:

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:

   ```
   WRKACTJOB
   ```
3. Display the system log QHST to verify it is up to date:

   ```
   DSPLOG  LOG(QHST)
   ```

   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:

   ```
   WRKF FILE(QSYS/QHST*)
   ```

   Look at the list to verify that you saved all copies of the log that will be needed later.
5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.
6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```
7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)
   ```

Keep this list with your backup log or your save system tapes for future reference.

8. Change the QSYSOPR message queue:

```
CHGMSGQ QSYSOPR DLVRY(*BREAK)  SEV(60)
```

9. End all subsystems:

```
ENDSBS SBS(*ALL) OPTION(*IMMED)
```

Messages are sent to the QSYSOPR message queue indicating that the subsystems have ended and the system is in a restricted state.

10. Load the first tape and make the tape device ready.
11. Save the system:

```
SAVSYS DEV(TAP01) ENDOPT(*LEAVE)
```

12. When a message similar to the following appears, load the next tape or make the device ready, and then enter R.

```
Device was not ready or next volume was not loaded (C R)
```

13. Save all user and IBM libraries:

```
SAVLIB LIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
       ACCPTH(*YES)
```

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous SAVLIB LIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be saved beyond the failure by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

```
DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
        OUTPUT(*PRINT)
```

The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.

3. Load a new tape and type the following:

```
SAVLIB LIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
     STRLIB(library-name) ACCPTH(*YES)
```

Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

14. Save the documents, folders, and distribution documents:

```
SAVDLO DLO(*ALL) FLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
```

15. Start the subsystems:

```
STRSBS SBSD(controlling-subsystem)
```

If you are installing a new release of the OS/400 licensed program, continue with the manual *Licensed Programs and New Release Installation Guide*.

Procedure for Installing or Restoring Licensed Internal Code Using the SAVSYS or Distribution Tapes

Installing or restoring the licensed internal code is done using option 23 (Restore) or option 24 (Install), which is run completely from the control panel. Because they are run completely from the control panel, these functions are referred to as **stand-alone** functions. The following is a description of these options.

Option — **Description**

23 (Restore)
Copies all licensed internal code from tape and replaces the licensed internal code found on disk. Select this option to do the following:

- Replace an existing system's licensed internal code without losing customer data already on the system.
- Recover the PTF index after moving or removing a disk unit from the system ASP, or starting checksum protection for the system ASP.
- Update a system's licensed internal code to a new release.
- After a disk unit failure other than unit 1 in the system ASP requires that you restore the licensed internal code. The system configuration will not be lost.

24 (Install)

Warning: Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

The data on all the remaining disk units will not be deleted at this time. However, during the restore of the OS/400 licensed program the disk units are added back to the configuration and all data and user ASPs are deleted. Select this utility only when:

- Restoring storage from the SAVSTG tapes.
- Doing an initial program load (IPL) on a system that contains no customer data and no licensed internal code.
- Replacing the load source unit. The service representative installs the licensed internal code during the service procedures.

When you perform either of these options, the system will continuously display system reference codes (SRCs) in the control panel display lights. The yellow System Attention light will be on whenever user input is needed. SRCs that start with A6 are indicating that the system is waiting for you to do something, such as answer a question or make a tape device ready. When xx is shown in the SRC (such as A6xx 6001), a variety of characters may be shown in the place where the xx appears. The Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" has a description of the SRCs. All other SRCs are discussed in the associated device manual.

Procedure for Installing or Restoring Licensed Internal Code Using the SAVSYS

The documentation for these functions makes reference to a **licensed internal code tape**. This is the **first** volume of the customer's most recent set of tapes that were used to save the system or the distribution tapes. The **model-unique licensed internal code tape** is the tape found in the service kit delivered with your system.

If installing or restoring the licensed internal code from the distribution tapes, you must also install the last cumulative program temporary fix package after restoring the operating system. For information about restoring the remaining parts of the system, including program temporary fixes, see the topic "Restoring User Profiles, Device Configurations, User Libraries, and Authority" on page 10-33.

Before You Begin ...

☐ Under certain circumstances, it may be necessary for you to install or restore the licensed internal code from the SAVSYS or distribution tapes. Use the correct function code as follows:

- After a failure of the disk unit that contains unit 1, use option 24.
- After a failure of a disk unit (that is not unit 1) in the system ASP if it does not have checksum protection, use option 23.
- After starting checksum protection, use option 23.
- When moving or removing a unit from the system ASP and user ASPs are configured, use option 23.
- When a new release of the operating system requires you to restore a new release of the licensed internal code, use option 23.
- When a restore of the licensed internal code is recommended by the IBM Software Support Center, use option 23 or 24 as directed.

Note: If you have not saved your system since installing the system, use the distribution tapes to install the licensed internal code. Otherwise, use the most recent SAVSYS tapes.

☐ Find the most recent set of SAVSYS tapes or distribution tapes supplied by IBM.

☐ Find the most recent cumulative program temporary fix package if you are restoring or installing the licensed internal code from the distribution tapes.

☐ Find the list of all the licensed internal code fixes applied to your system at the time you saved the entire system. This list should be attached to your backup log or found with the tapes used to save the entire system.

Print a list of all the licensed internal code fixes currently on the system, type the following and press the Enter key:

```
DSPPTF LICPGM(5738999) OUTPUT(*PRINT)
```

☐ If you are installing (option 24) the licensed internal code, find the model-unique licensed internal code tape.

☐ Clean the read and write head of the tape unit.

Perform the following steps to install or restore the licensed internal code.

1. Sign on the system as QSECOFR.
2. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ  MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```
3. End all jobs before installing or restoring the licensed internal code:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Press the Enter key. Messages are displayed indicating when the subsystems have ended and the system is in a restricted state. After the subsystems have ended, continue with the next step.
4. Power down the system before installing or restoring the licensed internal code:

   ```
   PWRDWNSYS OPTION(*IMMED)
   ```

   Note: When the Power On light goes off on all racks of the 9406, or the control panel of the 9402 or 9404, continue with the next step.
5. Ensure the key is in the keylock switch on the control panel.
6. Turn the key to the Manual position.
7. Press the Function Select switch to display **02** in the Function display on the control panel.
8. Press the Enter button on the control panel.
9. Select IPL type D (this specifies that the IPL source comes from tape) by pressing the Function Select switch on the control panel until **D** is shown on the Data display.
10. Press the Enter button on the control panel.
11. For the 9406 system unit, ensure that the power switches for the IPL tape unit and all disk units are in the On position.
12. Find the licensed internal code tape, which is the first volume of the most recent set of save tapes.
13. Place the tape in the tape unit used for the IPL. See the setup manual for the device for more information on loading the tape.

    Note: If your tape unit cannot be loaded when the power is off, continue with the next step. You will be prompted later by an SRC code for the tape.
14. Turn on the power to the system by pushing the Power switch up. The switch returns to center after you push it. The 9402 system unit has a green button labeled Power On.
15. Wait as explained below for the tape unit to power on. See the following explanations:

    Notes:

    a. SRC A100 1938 (Tape not found) may be displayed along with the System Attention light until the tape unit is turned on. Continue with the next step.

    b. SRC A100 1933 or 1934 (Tape not ready) may also be displayed along with the System Attention light. When the tape is ready, continue with the next step. If this SRC is displayed for more than 1 or 2 minutes, call your service representative.

c. SRC 2507 XXXX appears if manual select IPL intervention is required. If intervention is required, press the Function Select switch until 03 appears. Press the Enter button on the control panel.

16. Ensure that the tape is online or ready. No action is required for tape units that perform this step automatically (such as the tape cartridge unit).

17. Ensure that the console display is turned on.

18. Wait for the yellow System Attention light on the control panel to light up.

    There is a delay while the system loads information from the tape. SRCs showing status are continuously updated on the control panel while processing occurs. This can take from 5 to 20 minutes; the time varies depending on the speed of the tape unit and the processor speed for the specific system model.

    When SRC A6xx 6001 is displayed, the system is prepared to start installing or restoring the licensed internal code on the disk unit containing unit 1. Continue with the next step.

19. Select the correct function code by pressing the Function Select switch on the control panel until the correct function code is displayed. (See the "Before You Begin" section in this topic to determine the correct function code.)

    Warning: Option 24 (Install) is used only to recover from the loss of unit 1 in the system ASP, or to recovery from a disaster. Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

    If you selected function code 24, the System Attention light light may appear in one or two minutes and SRC A6xx 6002 is shown. If you are sure you want to install the licensed internal code, select function code 24 again.

    Notes:

    a. When SRC A6xx 6002 is displayed, option 23 can also be selected. If option 23 is selected, a restore of the licensed internal code is performed (not an install). Data on the disk units will not be lost.

    b. If another SRC appears after A6xx 6001 that is not in the A6xx xxxx format, then the system needs additional attention. Call your service representative.

       If the following SRCs appear after SRC A6xx 6001 is displayed, see Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" for an explanation of these SRCs and the steps to follow.

       - A6xx 6002 Disk unit may contain a valid system
       - A6xx 6003 Disk unit not currently a load source
       - A6xx 6004 Disk unit not currently a load source
       - A6xx 6005 Disk unit not found

20. Press the Enter button on the control panel.

    Note: The system will start displaying status SRCs again, which will be continuously updated to show the status of installing the licensed internal code. An example of a status SRC is D6xx 6201 (stand-alone install operations is running).

21. If the yellow System Attention light is on again, and SRC A6xx 6048 (New tape volume needs to be loaded) appears, the system needs the next tape.

The xx tells which volume needs to be loaded. Load the correct tape and make the device ready. The install operation will automatically continue.

If SRC A6xx 6051 appears, the stand-alone function is requesting the model-unique licensed internal code tape found in the service kit. Unload the current tape from the tape device and load the model-unique licensed internal code tape.

- A6xx 6051 Model-unique licensed internal code tape needs to be loaded
- A6xx 6052 Tape loaded was not the model-unique licensed internal code,

If another SRC A6xx xxxx is displayed, look up the displayed SRC in Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" on page A-1 and follow the instructions. For all other SRCs call your service representative.

22. After the install or restore operation of licensed internal code or the model-unique licensed internal code is complete, the system will automatically perform an IPL.
23. Remove the model-unique licensed internal code tape from the tape unit (if it is loaded) when the IPL or Install the System menu is shown.
24. If you have other restore steps to perform and the model-unique licensed internal code tape was loaded, load the previous save tape.

This completes the installing or restoring of the licensed internal code. Continue with "Restoring the OS/400 Licensed Program" on page 10-19.

Procedure for Restoring the Operating System

The following procedure is used to restore only the OS/400 licensed program from tape. The procedure for restoring the operating system assumes that the licensed internal code is already installed or restored on the system. Normally, the licensed internal code is installed. However, if you must install or restore the licensed internal code, first follow the procedure described in the topic "Procedure for Installing or Restoring Licensed Internal Code Using the SAVSYS or Distribution Tapes" on page 10-14 before installing the operating system.

Note: If you are using this procedure as part of restoring the entire system after installing or restoring the licensed internal code, go to step 1 in "Restoring the OS/400 Licensed Program"

To restore the operating system without restoring the licensed internal code, you must:

- Perform a manual IPL of the system
- Select install options
- Select restore options

For the following procedure, you need either the distribution media containing the OS/400 program or the media from the last SAVSYS operation.

Performing a Manual Initial Program Load (IPL)

Use the following steps to perform a manual IPL. These steps are done only if you have not installed or restored the licensed internal code.

Note: Do not perform a manual IPL if you have installed or restored the licensed internal code as part of the restore operation for restoring the entire system. Continue with step 1 in the topic "Restoring the OS/400 Licensed Program."

To perform a manual IPL, use the following steps:

1. Ensure the key is in the keylock switch on the control panel.
2. Turn the key in the keylock switch until it points to the Manual position.
3. Power down the system:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
   ```

Restoring the OS/400 Licensed Program

To select the install options, you use two displays. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system and for the system date and time.

Use this procedure if:

- You are restoring only the operating system and have performed a manual IPL. If you have not performed an IPL, see the topic "Performing a Manual Initial Program Load (IPL)." If you have already performed a manual IPL, go to step 1.
- You are using this procedure to restore the entire system after you have installed the licensed internal code (function code 24). Go to step 1.
- If you restored the licensed internal code (function code 23) and want to restore the operating system. Go to step 1.

When the IPL completes after installing the licensed internal code, the IPL or Install the System display is shown. Continue with the following:

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                         IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

   Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

```
                    Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.















 F12=Cancel
```

3. Press the Enter key.

4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                    Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .   2924

F3=Exit          F12=Cancel
```

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                 Confirm Language Feature Selection

Language feature . . . . . . . . . . . . . . . . . :   2924

Press Enter to confirm your choice for language feature.
  Installing the system will continue.
Press F12 to return to change your choice for
  language feature.

F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. The following display is shown only if disk units have been attached to the system and are in nonconfigured status.

```
                 Add All Disk Units to the System

Select one of the following:

        1. Add all disk units to the system auxiliary storage pool
        2. Keep the current disk configuration
        3. Perform disk configuration using DST




Selection
    __
```

Option 1 (Add all disk units to the system auxiliary storage pool)

Select this option if you want to add all of the nonconfigured units to the system auxiliary storage pool. Before adding the units to the system, all data stored on the non-configured units is deleted.

Note: Adding units can change the checksum set configuration of the system ASP. You can use option 3 (Perform disk configuration using DST) to calculate the effect of adding units to the system ASP.

Option 2 (Keep the current disk configuration)

Select this option if you plan to add the nonconfigured units to user ASPs or use them as spare units. This option continues the IPL without adding units to the system configuration. The disk units that are in nonconfigured status will remain so.

Option 3 (Perform disk configuration using DST)

This option starts the Dedicated Service Tools (DST). On the DST main menu, select option 4 (Work with Disk Units).

8. Press the Enter key.

   As the disk units are configured, the following display is shown:

```
Formatting additional disk units in progress.
        Seconds running . . . . : ____
```

Adding disk units takes several minutes. The time it takes depends on the size of each unit and the ability of the system to do multiple adds at the same time.

9. Status messages are displayed.

   The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                          Running IPL Step

Current IPL step . . . . : Storage Management Recovery
```

After the IPL steps complete, the Install the Operating System menu appears.

```
                      Install the Operating System

Type choices, press Enter.

    Default
      option . . . .  _           1=Take defaults, show no
                                    other installing displays
                                  2=Change installing options

    Date:
      Year . . . . .  _           00-99
      Month  . . . .  _           01-12
      Day  . . . . .  _           01-31

    Time  . . . . . . _   00-23   HH is hours
                      _   00-59   MM is minutes
                      _   00-59   SS is seconds
```

10. When the Install the Operating System display is shown, use the following information to respond to the prompts.

    Default Option

Value	**Description**
1	Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation. If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.
2	If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system. If you select 2 for *Default option*, the Installing Options display appears.

    Date

    The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

    Time

    The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

11. Press the Enter key.

    If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages are for your information only. Continue loading tapes in sequence when

messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

If you selected option 2 (Change installing options), the following display is shown.

```
                         Specify Install Options

Type choices, press Enter.

    Restore option . . . . __          1=Restore programs and language
                                         objects from current tape
                                       2=Do not restore programs or
                                         language objects
                                       3=Restore only language objects
                                         from current tape
                                       4=Restore only language objects
                                         from a different tape




    Job and output
      queue options. . . . 1           1=Clear,  2=Keep
```

12. When the Installing Options display appears, use the following information to respond to the prompts.

    Restore Option

Value	**Description**
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

    Notes:

    a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.
    b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.
    c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.
    d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.
    e. The system entry-point table is created again.
    f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user

decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	Description
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                       Specify Restore Options

Type choices, press Enter.

    Restore from tape:

      System values . . . . . 2      1=Restore,  2=Do not restore
      Edit descriptions . . . 2      1=Restore,  2=Do not restore
      Message reply list. . . 2      1=Restore,  2=Do not restore
```

13. Using the following information, respond to the prompts on the Restore Options display.

System Values

Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID	Default system code page and character set
QCURSYM	Currency symbol
QDATFMT	Date editing format
QDATSEP	Date separator character
QDECFMT	Decimal data editing format
QKBDTYPE	Default work station keyboard type
QLEAPADJ	Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Value	**Description**
1=Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

14. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

15. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    `Load the next volume on the tape device (1 2).`

16. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    `Operating system has been installed. IPL in progress.`

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

17. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).
18. Press the Enter key. Informational messages are displayed.
19. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.
20. When the IPL Options display is shown, respond to the prompts using the following information.

```
                                  IPL Options

 Type choices, press the Enter key.

   System date . . . . . . . . . . . . . .   07 / 26 / 88    MM / DD / YY
   System time . . . . . . . . . . . . . .   12 : 00 : 00    HH : MM : SS
   Clear job queues . . . . . . . . . . . .  N               Y=Yes,  N=No
   Clear output queues . . . . . . . . . .   N               Y=Yes,  N=No
   Clear incomplete job logs . . . . . . .   N               Y=Yes,  N=No
   Start print writers . . . . . . . . . .   Y               Y=Yes,  N=No
   Start this device only . . . . . . . . .  N               Y=Yes,  N=No


   Set major system options . . . . . . . .  N               Y=Yes,  N=No
   Define or change system at IPL . . . . .  N               Y=Yes,  N=No







 Last power-down operation was normal
                                     (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure 10-4. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a Y to start the console device only, or type an N to start all devices.

d. Enter the value for set major system options.

   Type an N for each option to leave them unchanged, or type Y to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to N. If you installed the licensed internal code using option 24 (Install), the default value is set to Y.

21. Press the Enter key.

22. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

```
                  Edit Rebuild of Access Paths                RCHAS331
                                                      05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                   ------------Access Paths----------  Unique  Rebuild
Seq   Status       File        Library     Member      Keyed   Time
25__  IPL          FILE234512  LIBRARY111  MBR1234567  No      00:00:56
25__  IPL          FILE234513  LIBRARY111  MBR1234567  No      00:00:56
75__  IPL          FILE234514  LIBRARY111  MBR1234567  Yes     00:00:56
75__  IPL          FILE234515  LIBRARY111  MBR1234567  Yes     00:00:56
88__  IPL          FILE234516  LIBRARY111  MBR1234567  No      00:00:56
99__  AFTIPL       FILE234517  LIBRARY111  MBR1234567  Yes     00:00:56
*OPN  OPEN         FILE126789  L123456789  MBR4567890  Yes     12:34:56
*OPN  OPEN         FILE346789  L123456789  MBR4567890  No      12:34:56
*HLD  HELD         F123336789  L123456789  MBR4567890  No      10:30:06
*HLD  HELD         F123456789  L123456789  MBR4567890  Yes     99:56:01
                                                          More...

F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                         Display Access Path Status

IPL Threshold . . . . . . : 88

          ----------Access Paths----------   Rebuild     Current
Status    File        Library     Member       Build Time  Run Time
RUN       F123456789  L123456789  MBR4567890               00:00:01
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
SYS       F123456789  L123456789  MBR4567890   12:34:56
SYS       F123456789  L123456789  MBR4567890   12:34:56
IPL       F123456789  L123456789  MBR4567890   12:34:56
                                                             More...




F3=Exit and continue IPL    F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN – indicates that the access path is being rebuilt.
- SYS – indicates the access path is a system access path and is waiting to be rebuilt.
- JRN – indicates that the access path is being recovered from its associated journal.
- IPL – indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

23. Press the Enter key to continue.
24. Ensure the keylock switch is in the Normal position.
25. This completes the restore operation for the operating system if you have no other restore steps to perform.
26. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

Restoring User Profiles, Device Configurations, User Libraries, and Authority

Before You Begin. . .

- ☐ Clean the read and write head of the tape unit.
- ☐ Find the tape volume that contains the user profiles.

You may want to do the following:

- If you do not know where the user profiles are stored on tape, determine where they are by using the DSPTAP command with DATA(*SAVRST) specified. (This is not necessary if you are restoring user profiles from the SAVSECDTA media.)

  Examine each tape until you find the file labeled QFILEUPR containing object type *USRPRF.
- Find the file on tape containing the device configuration objects using the DSPTAP command with DATA(*SAVRST) specified. Examine each tape until you find the file labeled QFILEIOC containing object types *DEVD, *CTLD, *LIND, *COSD, and *MODD)
- Use the VRYCFG command to vary off the device configuration objects to be restored. The device configuration objects cannot be active when being restored. When doing an IPL, specify **Yes** for the *Start this device only* prompt on the IPL options display. This does an IPL with only the console and the tape unit varied on.

Other Considerations: Use the restore commands (not option 21 on the Restore menu) if:

- You prefer to enter the commands manually.
- You saved changed objects or have journal changes to apply.
- You performed individual save operations instead of SAVLIB(*NONSYS), you must use a RSTLIB command for each saved library. If you saved individual objects using the SAVOBJ or SAVCHGOBJ command, you must use a RSTOBJ command for each group of saved objects.
- You saved the security information with the Save Security Data (SAVSECDTA) command, you must restore the information using the restore commands.
- You saved logical file access paths using either the SAVOBJ or SAVCHGOBJ command, you must restore the logical files the same way you restored the physical files using the RSTOBJ command.
- Your documents and folders were saved in multiple tape files using SAVDLO DLO(*ALL) FLR(*ANY), use DSPTAP DATA(SAVRST) to display the beginning and ending sequence numbers. You will need this information later during the restore operation.

Use **one** of the following two methods to restore the user profiles, device configuration objects, user libraries, and authority:

1. Use option 21 (The system) on the Restore menu if you are restoring the user profiles from a SAVSYS tape and the considerations listed previously in "Other Considerations" do not apply.

2. Use the Restore commands (not option 21 on the Restore menu) if any of the consideration listed previously in "Other Considerations" apply.

Method 1. Using Option 21 (The system) on the Restore Menu

To restore user profiles, device configuration objects, user libraries, and authority, use the following steps:

1. Sign on the system as the security officer; type QSECOFR in the user prompt and the password (if password security is active) associated with that user ID on the Sign On display.
2. Press the Enter key.
3. Type the following command on the command line and press the Enter key.

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

4. Ensure that the correct volume of your last set of save tapes is loaded and make the tape device ready. The tape should begin with a file containing the user profiles. Run the DSPTAP using *SAVRST to find the file labeled QFILEUPR.
5. Go to the Restore menu:

   ```
   GO RESTORE
   ```

   The Restore menu is shown.

```
RESTORE                           Restore
                                                       System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Licensed programs

    20. All libraries other than system library
    21. The system
    22. All IBM libraries other than system library
    23. All user libraries

    50. Restore from System/36 format
                                                                More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

Note: To prevent an interrupted restore caused by incomplete restore messages, run the following commands before selecting option 21 from the Save menu:

a. To display the reply list sequence numbers currently used, type the following and press the Enter key.

   ```
   WRKRPYLE
   ```

b. To add a reply list entry, type the following (where xxxx is an unused sequence number 1-9999) and press the Enter key.

   ```
   ADDRPYLE SEQNBR(xxxx) MSGID(CPA3708) RPY('G')
   ```

c. To change the job, type the following and press the Enter key.

```
CHGJOB INQMSGRPY(*SYSRPYL)
```

d. To change the QSYSOPR message queue, type the following and press the Enter key.

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(95)
```

6. Press the Enter Key.
7. Select option 21 (The system) on the Restore menu and press the Enter key. The following display is shown.

```
                          Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .   : TAP01                Names
                                     ________
                                     ________
                                     ________

  Prompt for commands . . . . .   :   Y                   Y=Yes, N=No










F3=Exit   F12=Cancel
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device after the first tape is read.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting and uses the default values.

Note: If you have multiple tape files from the SAVDLO DLO(*ALL), type a *Y* (the default) for the *Prompt for commands* field on the Specify Command Defaults display. When the RSTDLO command prompt is displayed, enter the beginning and ending sequence numbers to restore all SAVDLO tape files.

Option 21 will guide you through the following if you selected Y for the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS *ALL *IMMED
b. RSTUSRPRF *ALL
c. RSTCFG *ALL
d. RSTLIB SAVLIB(*NONSYS)

Save and Restore

e. RSTDLO DLO(*ALL) SAVFLR(*ANY)
f. RSTAUT
g. STRSBS

8. Continue loading the save tapes in sequence when the system sends a message to load the next volume.

Note: Depending on the circumstances and how much of the system you have restored, the DSPTAP command may or may not work. If not enough of the system is restored, refer to the list of libraries you created when you saved the system.

If a media error occurs....

If an unrecoverable media error occurs during the RSTLIB procedure, you can restart the procedure using the STRLIB parameter on the RSTLIB command. Do one of the following:

1. If the STRLIB parameter was used on the SAVLIB command, determine the next library to be restored beyond the failure by displaying the first SAVLIB LIB(*NONSYS) tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). Load the next tape and enter the following:

```
RSTLIB SAVLIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
     STRLIB(starting-library-name)
```

2. If the STRLIB parameter was not used on the SAVLIB command, look at the job log to determine the library where the previous RSTLIB failed.

   a. Determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). The correct starting library is normally the second library listed after the last library restored successfully.

   b. Load the first tape of the SAVLIB media and type the following command to start the restore operation again.

```
RSTLIB SAVLIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
     STRLIB(starting-library-name)
```

9. If you are restoring the system using the SAVSYS tapes, this completes the restore operation. There are no other restore steps to perform.

10. Power down the system (PWRDWNSYS OPTION(*IMMED) RESTART(*YES) to perform a normal IPL and return the system to normal operations.

Method 2. Using the Restore Commands

To use the commands to restore the system, do the following:

1. Sign on the system.
2. Change the QSYSOPR message queue:

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
```

3. End all subsystems:

```
ENDSBS SBS(*ALL) OPTION(*IMMED)
```

Messages are sent indicating when the subsystems have ended and the system is in a restricted state.

4. Restore user profiles from the correct save tape file (label QFILEUPR).

   Note: Use the tapes from the most recent complete save operation (SAVSYS), or if the security data was saved since the last complete save operation, use the SAVSECDTA tapes.

   If the SAVSYS tape is used:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
   ```

   If the save security data tape (SAVSECDTA) is used:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

5. Restore the device configuration objects from the SAVSYS tape:

   ```
   RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*LEAVE)
   ```

   The time that this takes can vary significantly.

6. Restore the IBM and user libraries in one of the following ways:

   If you used SAVLIB LIB(*NONSYS) to save the IBM-supplied and user libraries, load the correct volume and then type the following:

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
   ```

   Or, if you used SAVLIB LIB(*IBM) and SAVLIB LIB(*ALLUSR) to save the IBM and user libraries, load the correct tape and then type the following two commands. The first command must complete before entering the second command.

   ```
   RSTLIB SAVLIB(*IBM) DEV(TAP01) ENDOPT(*LEAVE)
          MBROPT(*ALL) ALWOBJDIF(*ALL)
   ```

   ```
   RSTLIB SAVLIB(*ALLUSR) DEV(TAP01) ENDOPT(*LEAVE)
          MBROPT(*ALL) ALWOBJDIF(*ALL)
   ```

   Notes:

   a. If you saved individual libraries and objects with the SAVLIB, SAVOBJ, and SAVCHGOBJ commands, then you will have to restore the individual libraries and objects with the RSTLIB command (not RSTLIB SAVLIB(*NONSYS)) and the RSTOBJ command with ALWOBJDIF(*ALL) and MBROPT(*ALL) specified.

   b. Depending on the circumstances and how much of the system you have restored, the DSPTAP command may or may not work. If not enough of the system is restored, refer to the list of libraries created when you saved the system.

If a media error occurs....

If an unrecoverable media error occurs during the RSTLIB procedure, you can restart the procedure using the STRLIB parameter on the RSTLIB command. Do one of the following:

1. If the STRLIB parameter was used on the SAVLIB command, determine the next library to be restored beyond the failure by displaying the first SAVLIB LIB(*NONSYS) tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). Load the next tape and enter the following:

```
RSTLIB SAVLIB(*NONSYS, *ALLUSR, or *IBM) DEV(TAP01)
       ENDOPT(*LEAVE) STRLIB(starting-library-name)
```

2. If the STRLIB parameter was not used on the SAVLIB command, look at the job log to determine the library where the previous RSTLIB failed.

   a. Determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). The correct starting library is normally the second library listed after the last library restored successfully.

   b. Load the first tape of the SAVLIB media and type the following command to start the restore operation again.

```
RSTLIB SAVLIB(*NONSYS, *ALLUSR, or *IBM) DEV(TAP01)
       ENDOPT(*LEAVE) STRLIB(starting-library-name)
```

7. If you have documents and folders to restore, load the correct tape and type the following:

   Note: If the saved data is contained in one or more files with folders and documents, you must specify beginning and ending sequence numbers on the RSTDLO command. Otherwise, only the first file with folders and documents will be restored.

```
RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
       SEQNBR(beginning-sequence ending-sequence)
```

   Note: If you are not using journaling, or do not have changed objects or daily mail (SAVSECDTA tape) to restore, continue with the next step. Otherwise, continue with "Restoring Changed Objects and Applying Journal Changes."

8. To restore the authority, type the following:

```
RSTAUT
```

   This completes the restore operation.

9. Power down the system (PWRDWNSYS OPTION(*IMMED) RESTART(*YES)) to perform a normal IPL and return the system to normal operations.

Restoring Changed Objects and Applying Journal Changes

If you are using journaling and need to later apply journaled changes, continue with the following steps. Otherwise ignore these steps and continue with "Restoring Changed Objects."

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily save operations using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform the steps in "Working with Journals" for the system supplied journal QUSRSYS/QAOSDIAJRN. If you specified OBJJRN(*YES) on the SAVCHGOBJ command, you do not need to apply journal changes.

Working with Journals

1. Type the following and press the Enter key:

   ```
   WRKJRN
   ```

2. The Specify Journal Name display is shown. Specify *ALL for the *Library name* prompt and press the Enter key.

3. The Work with Journals display is shown. Type a **5** in the *Opt* field for each journal that you want to apply journaled changes to display the name of the currently attached journal receiver. Write down the name of the journal and journal receiver.

   Note: If you are using OfficeVision/400 or PC Support/400, you must apply journaled changes to the system supplied journal QUSRSYS/QAOSDIAJRN.

4. You cannot restore journal receivers from the SAVCHGOBJ media if active journal receivers are attached. To later apply all journaled changes that have occurred since the last complete save operation, you must restore the receivers to the system from the SAVCHGOBJ media.

   For each journal identified in the previous step, do the following steps:

   a. Create a journal receiver that will be used as a temporary receiver. Give it a name that will identify it as a temporary receiver, for example, TEMPnn. You can enter a description in the text (TEXT parameter) that identifies it as a temporary receiver for disaster recovery.

      ```
      CRTJRNRCV JRNRCV(library-name/TEMPnn)
             TEXT('temporary journal receiver for journal xxx')
      ```

   b. To detach the current receiver and attach the new TEMPnn receiver, type the following and press the Enter key.

      ```
      CHGJRN JRN(library-name/journal-name) JRNRCV(library-name/TEMPnn)
      ```

   c. Delete the detached journal receiver (identified in the step where you wrote down the name of the journal and journal receiver) using the Delete Journal Receiver (DLTJRNRCV) command. (This allows the journal receivers on the SAVCHGOBJ media to be restored successfully.

      ```
      DLTJRNRCV JRNRCV(library-name/journal-receiver)
      ```

      If you receive message CPA7025 *Receiver never fully saved*, enter an **I** to ignore and press Enter to continue the delete.

Restoring Changed Objects

Load the SAVCHGOBJ tape and enter the following to restore changed objects.

```
RSTOBJ OBJ(*ALL) DEV(tape-device) SAVLIB(library-name)
          OBJTYPE(*ALL) ENDOPT(*LEAVE) MBROPT(*ALL) ALWOBJDIF(*ALL)
```

You must repeat this step for every library saved using the SAVCHGOBJ command. If you specified SAVCHGOBJ LIB(*ALLUSR), type the following to determine the libraries that were saved:

```
DSPTAP *SAVRST
```

If you are using journaling, perform the steps in "Applying Journal Changes" for each journal you wish to apply journal changes to. Otherwise, continue with "Restoring Changed Documents and Folders."

Applying Journal Changes

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily saves using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform this step for the system supplied journal QUSRSYS/QAOSDIAJRN.

1. Determine the name of the last journal receiver (the last receiver restored) by entering the following:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

2. Press F15 (Work with receiver directory) to display the last receiver with a status of SAVED or PARTIAL. Write down the name of the receiver.

3. Create a receiver that follows the same naming convention as the last receiver but assign it a number of one greater.

   ```
   CRTJRNRCV JRNRCV(library-name/journal-receiver-nameNN)
   ```

   By doing this, you are doing what the CHGJRN command would normally do if the receiver was the current receiver being detached with a new receiver name being created. This allows your naming convention for journal receivers to continue.

4. Use the CHGJRN command to detach the temporary receiver and attach the new receiver you just created.

   ```
   CHGJRN JRN(library-name/journal-name)
        JRNRCV(library-name/journal-receiver-nameNN)
   ```

5. Determine the chain of receivers to be used in the APYJRNCHG command by entering the following command:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

   Press F15 (Work with receiver directory) to display the receivers. Write down the first and last receiver that you restored (last receiver is prior to the TEMPnn receiver). Notice that the first and last is the same receiver if only one journal receiver was restored.

6. When applying journal changes and the ending receiver has a status of PARTIAL (saved while attached), the TOENT parameter requires that a sequence number be specified on the APYJRNCHG command. Determine the last entry to be applied by entering the following command for the last receiver (identified in previous step):

   ```
   DSPJRNRCVA JRNRCV(library-name/last-journal-receiver-name)
   ```

   Write down the value for the *Last Sequence Number* field.

7. Enter the following command to apply the journaled changes using the first and last journal receivers identified on the Work with Receiver Directory display.

   ```
   APYJRNCHG JRN(library-name/journal-name)
            FILE((library-name/*ALL))
            RCVRNG(lib-name/first-receiver lib-name/last-receiver)
            FROMENT(*LASTSAVE) TOENT(last-entry)
   ```

Attention

Do not apply journal changes to the document and folder search index database files (QAOSSH10 through QAOSSH19) for journal QAOSDIAJRN in library QUSRSYS. You must specify individual files on the FILE parameter instead of *ALL.

```
APYJRNCHG JRN(QUSRSYS/QAOSDIAJRN)
        FILE((QUSRSYS/QAOSAH05) (QUSRSYS/QAOKLY02)
             (QUSRSYS/QAOSAH07) (QUSRSYS/QAOKLY03)
             (QUSRSYS/QAOKDYX4) (QUSRSYS/QAOKNY06)
             (QUSRSYS/QAOKDY01) (QUSRSYS/QAOKDY04)
             (QUSRSYS/QAOKDY05) (QUSRSYS/QAOKDY07)
             (QUSRSYS/QAOKDY08) (QUSRSYS/QAOKDY09))
        RCVRNG(lib-name/first-receiver lib-name/last-receiver)
        FROMENT(*LASTSAVE) TOENT(last-entry)
```

Restoring Changed Documents and Folders

If you performed daily save operations for documents and folders, do the following steps. Otherwise, continue with the RSTAUT command.

1. Load the last daily SAVDLO tape.
2. If you performed daily save operations to backup all new folders and new and changed documents since the last complete SAVDLO operation, type the following and press the Enter key.

   ```
   RSTDLO DLO(*ALL) DEV(TAP01) SAVFLR(*ANY) ALWOBJDIF(*ALL)
   ```

   If you saved all documents referred to by the mail using SAVDLO DLO(*MAIL), type the following and press the Enter key.

   ```
   RSTDLO DLO(*MAIL) DEV(TAP01)
   ```

3. Restore users' authority by entering:

   ```
   RSTAUT
   ```

   The time it takes for the RSTAUT command to complete can vary significantly. The time depends on the number of user profiles and private authorities that were saved during the save operation.

4. This completes the restore operation.

   Power down the system by entering the following:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES)
   ```

   to perform a normal IPL and return the system to normal operations.

   Note: If you turned auto configuration off during the restore of the operating system, you must perform an IPL with the keylock switch in the **Normal** position.

Restoring Programming Temporary Fixes (PTFs)

If you are using this procedure to finish restoring the entire system and have applied program temporary fixes (PTFs) after the last save operation, you must restore the PTFs.

If you need to restore program temporary fixes (PTFs) as part of recovering the entire system, find the most recent cumulative program temporary fix tape. This package could be on a distribution tape or on a stand-alone tape.

You can use option 31 (Install program temporary fix package) on the Work with Licensed Programs menu to install all of the PTFs. All of the PTFs in the cumulative PTF package will be installed for the licensed programs you have installed on your system. Refer to the *AS/400 System PTF Shipping Information Letter* for special instructions that are required.

If you are restoring individual PTFs, see the *System Operator's Guide*, SC41-8082, for more information about applying individual PTFs.

1. Print a list of all the licensed internal code PTFs currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(5738999) OUTPUT(*PRINT)
   ```

2. Compare the list of PTFs in the previous step with the list you printed when you saved the system.

   Any PTFs that are not found must be loaded again.

3. If you do not have the PTFs identified in the previous step, order them and then apply them.

Chapter 11. Save and Restore Storage Processes and Procedures

The save storage process copies the licensed internal code and all of the disk unit data to tape. The tape produced is a sector-by-sector copy of all permanent data on configured disk units. The save and restore storage processes are intended for disaster backup and recovery and are to be used along with the standard save and restore commands. They are not intended for copying or distributing to other systems. This procedure does not allow for single objects to be saved or restored.

The save and restore storage processes provide the fastest method for backing up and recovering an entire system that does not have checksum protection. The restore storage process is the fastest method for restoring the entire system. The performance difference is greater on faster tape units for both processes.

You can start the save storage process with the Save Storage (SAVSTG) command or an option on the Save menu. This command can only be started when the system is in a restricted state (no batch or interactive jobs running and all subsystems are ended). When the Save Storage command is run, the save function will cause an IPL of the system.

To use the restore storage function, the system must be in the DST-restricted state. The restore storage process is started by using an option on the Dedicated Service Tools (DST) menu. Understanding of libraries is not required to run the save and restore storage operations. The restore storage operation can only be started when the operating system is not active (before an IPL of the operating system).

When using the save and restore storage processes, the system cannot be used until the save or restore storage is complete.

Considerations When Using the Save Storage Process

The save storage process has the following considerations:

- The save storage media is intended for disaster recovery and cannot be used to restore individual objects. You must complement a save storage approach with the SAVSYS, SAVLIB, and SAVDLO commands.
- To properly implement a save storage approach, you should have multiple levels of your backup media.
- If a tape error occurs, the system attempts to recover from the error by automatically trying the operation again. If the system cannot recover, you can resume the save storage operation on a new tape volume. The operation continues from the last completed tape volume that was saved.

  As the amount of storage on the system increases, the chance of an unrecoverable media error increases. Clean the tape unit frequently.
- If the save storage operation is interrupted, it should not be started again using the tape volume that was being written to at the time of the interruption. For example, if the system ends abnormally, the data saved to the tape volume that was loaded is not usable. The save storage operation must be resumed using a new tape to replace the one in use at the time the error occurred.

- The save storage operation can only be run when the system is in a restricted state.
- The user must have save system (*SAVSYS) special authority to use the SAVSTG command.
- The SAVSTG command causes the system to power down and starts the system again as if PWRDWNSYS RESTART(*YES) was specified. An IPL of the system occurs after the command completes. The save storage function implicitly occurs during the IPL of the system from the dedicated service tools (DST) function.
- The first tape can be saved without an operator being present. After the first tape is saved, DST messages appear asking for the next tape so the save operation can continue.
- If the tape unit supports hardware data compression, then hardware data compression is used. If device data compression is not supported, then programming data compression may be used. In general, if the device operates at a faster rate than data compression can be performed, then uncompressed data is written to the device.
- The save storage operation does not save the following:
  - Disk sectors that are not used or that contain temporary data.
  - The model-unique licensed internal code. (Tape found in the service kit shipped with the system.)
- Only one tape unit can be used.
- A device name must be specified on the command. Expiration date (EXPDATE) and clear (CLEAR) parameters are optional. No volume ID can be specified.
- The save storage process does not start unless the system console is available. If the console is not available, a system reference code is displayed on the control panel.
- The save storage process does not start unless all of the configured disk units are operating.
- Save storage performance decreases significantly if the system is using checksum protection because of the extra number of checksum sectors that must be copied to tape.
- If the system is using mirrored protection, only one copy of the data from each mirrored pair is saved.
- When the save storage operation completes successfully, a normal IPL occurs.

The restore storage process has the following considerations:

- The save storage media is intended for disaster recovery and cannot be used to restore individual objects.
- If a tape error occurs, the system attempts to recover from the error by automatically trying the operation again.

  If an unrecoverable media error occurs, the system cannot be used until you do one of the following:

  - Try the restore storage operation again from the beginning using the same save storage media (consider cleaning the tape heads first).

- Restore a different level of save storage media. (To properly implement a save storage approach, you should have multiple levels of your backup media.)
- Restore your last set of SAVSYS, SAVLIB, and SAVDLO tapes using the appropriate restore commands.
- Use the Resume restore storage option starting with the next tape volume after the tape volume that had the error. Due to loss of data on the tape volume with the error, damaged objects will probably occur.

Save and Restore

- If the restore storage operation is interrupted, you can start the process again by using the volume that was being read at the time the restore operation was interrupted. For example, if the system ends abnormally, the data restored from the current tape volume is not usable, and then the restore operation must be started again from the beginning of the volume.
- The user must have the DST security level of full. For a description of the DST security levels, see *Chapter 2. Security Considerations* in the *Security Concepts and Planning* manual. After the restore storage process completes, the DST passwords are lost and must be entered again.
- The disk configuration of the restoring system must be the same disk configuration of the saving system. The disk types and models must be the same or equivalent with some additional devices. Serial numbers and physical addresses do not have to be the same. All disk units that were saved are required for the restore operation. For example, a 9332 Model 400 is equal to two 9332 Model 200s, or one half of a 9335 Model B01 is equal to another one half of a 9335 Model B01. See "Restoring the Save Storage Media on a Different System" on page 11-4.
- If the disk configuration cannot support mirrored protection because of an insufficient number of units, a restore operation can still be done provided there is one unit for each mirrored pair on the current system. Some mirrored units will now be unprotected because units are missing from the configuration. However, the system is available for operation. Missing units should be repaired or replaced in order for mirrored protection to be effective.
- Restore storage performance decreases significantly if the system is using checksum protection because the write operations must go through the checksum update process of reading old data and old checksum data, and then writing new data and new checksum data.
- If a system using checksum protection is saved, and then restored to a system whose disk configuration does not support checksum protection, then checksum protection is ended. Stopping checksum protection lengthens the next IPL of the system following the restore operation.
- If the system using mirrored protection is saved and then restored, the data is restored to only one unit of each mirrored pair. Each mirrored pair is synchronized again at the next IPL of the system, lengthening the IPL process.

  If the disk configuration has changes since the last save operation, there may be one unit of one or more mirrored pairs missing. The DST/SST Replace Disk Unit function must be used before synchronization can occur.
- To start the restore storage process, the system must be in the DST-restricted state.
- A type D IPL is required as the first step in the restore storage process.

This will restore the licensed internal code from the save storage tape. The licensed internal code PTFs are restored correctly.

- The model-unique licensed internal code must be restored using Option 24 (Install) of the licensed internal code stand-alone utilities unless unit 1 of the system ASP has not been changed between the time of the save operation and the time of the restore operation.
- To restore the rest of the system, the restore storage function must be accessed from the DST function. The DST function cannot be accessed after an IPL of the OS/400 licensed program.
- If the tape unit used during the save operation used hardware data compression, then the tape unit used during the restore operation must use hardware data decompression. If the tape unit used during the save operation did not use hardware data compression, then the tape unit used for the restore operation does not have any hardware data decompression dependencies.
- Only one tape unit can be used.
- The restore storage process does not start unless the console is available. If the console is not available, a system reference code is displayed on the operator control panel.
- If no errors occur during the restore process, it is not necessary to build the database file access paths again.
- When the restore operation completes, and no errors occurred, the system exists at the level when the SAVSTG command was run (including messages, spooled files, message queues, and licensed program PTFs.)

Restoring the Save Storage Media on a Different System

If you save storage on one system and restore the save storage media on a different system, the disk unit configuration must be same or equivalent with some additional devices.

When you do an IPL of the new system, the licensed internal code recognizes that the machine serial number is different and will reset the programming value to agree with the equipment value. This can cause the resource names to be different even if the configuration is the same.

If you plan to save storage on one system and restore it on another, you should do the following on the restoring system:

1. On the IPL following the restore storage operation, be sure the value for automatic configuration is set to off.
2. Delete all configuration objects.
3. Create the required configuration objects manually or set the automatic configuration value on to automatically create local configuration objects. You still need to manually create the remote configuration objects, such as the line description for SDLC (CRTLINSDLC command).

Because some of the resources may have been renamed, some functions can be misleading. For example, assume you opened the error log to look at old problems. Some entries for the renamed resource may be meaningless.

Working with the Save and Restore Storage Procedures

Before you begin any save procedure, it is recommended that you initialize enough tapes to complete the save operation. The number of tapes required depends on the size of the system, the number of objects being saved and the capacity of the tape. Use Table 4-2 on page 4-4 to determine tape capacities.

Initialize at least three more tapes than you think you will need. Each tape should have a volume ID and an external label that allows you to easily identify the tape.

An external label on the tape should identify the date and time of the save operation and what was saved. If an object needs to be restored, the tape that contains the object can be located quickly. A backup log should be provided to write down important information.

To initialize a new tape, do the following:

```
INZTAP DEV(TAP01) NEWVOL(BACKUP) NEWOWNID(user-profile-name)
       CHECK(*NO)
```

To initialize tapes that have data on them, display the tape to verify that you do not want to keep the data; then do the following:

```
INZTAP DEV(TAP01) NEWVOL(MONWK1) CHECK(*NO)
       ENDOPT(*UNLOAD)
```

If you are using tapes for the save operation that have data on them, you can use the CLEAR parameter to specify that data be cleared from the tape during the save operation. For example:

```
SAVSTG  DEV(TAP01) CLEAR(*ALL)
```

For more information about using tapes, see Chapter 4, "Save and Restore Media Considerations" on page 4-1.

Procedure for Saving Storage

Before You Begin . . .

- ☐ Correctly initialize the tapes as standard-labeled tapes and specify the maximum density for the tape unit you are using.
- ☐ Clean the read and write head of the tape unit.
- ☐ Apply any temporary licensed internal code fixes and then print a list of all the licensed internal code fixes presently applied to your system. See step 1. Keep this list with your backup log or your save storage tapes for future reference.

Note: The system will not be available for use until the save storage process completes.

Use the following procedure to save storage.

1. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key.

```
DSPPTF LICPGM(5738999) OUTPUT(*PRINT)
```

2. Notify users that the system is going down immediately.
3. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ QSYSOPR *BREAK  SEV(60)
   ```
4. Type the following to end all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Messages are sent to the QSYSOPR message queue indicating that the subsystems ended and the system is in a restricted state. When the subsystems have ended, continue with the next step.
5. Load the first tape, and make the tape unit ready.
6. Turn the keylock switch to a position other than Manual.
7. Enter the save storage command, such as:

   ```
   SAVSTG DEV(TAP01) CLEAR(*ALL)
   ```

   You can also enter an expiration date (EXPDATE(mmddyy)).
8. Press the Enter key. The system will power down with a restart IPL. This is similar to PWRDWNSYS OPTION(*IMMED) RESTART(*YES). This means that when the command is entered, the system will power down and perform an automatic IPL.

   When the IPL occurs, a dedicated service tools (DST) function starts saving storage. The operator does not need to be present for the first tape if it is positioned correctly and the expiration date checking does not cause an error.
9. If the tape is loaded correctly, the following save status display continually displays the progress of the save operation.

```
                    Save Storage

Status of the save . . . . . . . . . . . . . . . . :  Running

Percent saved. . . . . . . . . . . . . . . . . . . :

Number of sectors not readable . . . . . . . . . . :
```

   The *Percent saved* field on the display indicates the percentage of the total amount of disk sectors that have been saved. However, the percentage cannot be used to accurately estimate the time it will take, or the number of tapes needed to complete the save operation, because unused sectors are not saved.
10. One of the following displays may appear during the save operation.
    a. If another tape needs to be loaded, the following display appears:

```
                    Handle Tape or Diskette Intervention

Device:
 Type. . . . . . . . . . . . . . . . . . . :        ____
 Model . . . . . . . . . . . . . . . . . . :        ____
 Serial number . . . . . . . . . . . . . . :        __________
 Address . . . . . . . . . . . . . . . . . :        ___-_________
I/O manager code . . . . . . . . . . . . . :        _______
Reference code . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . :        ____
   Device  . . . . . . . . . . . . . . . . :        ____
Volume or file . . . . . . . . . . . . . . :
   Loaded  . . . . . . . . . . . . . . . . :        ________________
   Requested . . . . . . . . . . . . . . . :        ________________

Type choice, press Enter.

  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    _____________________
                                                    3=Continue
                                                    _____________________

F3=Exit             F12=Cancel
End of tape encountered.  Load next volume.
```

Load the next tape, select option 3 (Continue), and press the Enter key.

b. If a tape with active files is loaded and CLEAR(*NONE) was specified on the SAVSTG command, the following display is shown:

```
                        Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :        ____
Device model . . . . . . . . . . . . . . . . . . :        ____
Device serial number . . . . . . . . . . . . . . :        __________
Device address . . . . . . . . . . . . . . . . . :        ___-_________
I/O manager code . . . . . . . . . . . . . . . . :        _______
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . . :        ____
   Device  . . . . . . . . . . . . . . . . . . . :        ____

Type choice, press enter

  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    2=Ignore
                                                    3=Continue
                                                    _____________________

F3=Exit             F12=Cancel
Active files exist on media.
```

To continue the save operation to tape, select option 2 (Ignore) to ignore the active files, and press the Enter key.

c. If the tape unit is not ready, the following display is shown:

```
                    Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . :      ____
Device model . . . . . . . . . . . . . . . . . :      ____
Device serial number . . . . . . . . . . . . . :      __________
Device address . . . . . . . . . . . . . . . . :      ___-________
I/O manager code . . . . . . . . . . . . . . . :      ________
Reference code . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . :      ____
   Device  . . . . . . . . . . . . . . . . . . :      ____

Type choice, press enter

  Action  . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                  ____________________
                                                  3=Continue
                                                  ____________________



F3=Exit            F12=Cancel
Tape unit not ready.
```

Make the tape unit ready, select option 3 (Continue), and press the Enter key.

d. If the media that is loaded is write-protected, the following display is shown:

```
                    Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . :      ____
Device model . . . . . . . . . . . . . . . . . :      ____
Device serial number . . . . . . . . . . . . . :      __________
Device address . . . . . . . . . . . . . . . . :      ___-________
I/O manager code . . . . . . . . . . . . . . . :      ________
Reference code . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . :      ____
   Device  . . . . . . . . . . . . . . . . . . :      ____

Type choice, press enter

  Action  . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                  ____________________
                                                  3=Retry
                                                  ____________________



F3=Exit            F12=Cancel
Media is write protected.
```

Replace the tape with a tape that is not write-protected and select option 3 (Retry). Press the Enter key.

e. If the tape unit cannot process the tape, the following display is shown:

```
                    Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :       ____
Device model . . . . . . . . . . . . . . . . . . :       ____
Device serial number . . . . . . . . . . . . . . :       __________
Device address . . . . . . . . . . . . . . . . . :       ___-_______
I/O manager code . . . . . . . . . . . . . . . . :       _______
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . . :       ____
   Device  . . . . . . . . . . . . . . . . . . . :       ____

Type choice, press enter

  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    ______________________
                                                    3=Continue
                                                    4=Format

F3=Exit            F12=Cancel
Device is not able to process the media format
```

Select option 4 (Format), and press the Enter key.

f. If the tape loaded is not formatted, the following display is shown:

```
                    Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :       ____
Device model . . . . . . . . . . . . . . . . . . :       ____
Device serial number . . . . . . . . . . . . . . :       __________
Device address . . . . . . . . . . . . . . . . . :       ___-_______
I/O manager code . . . . . . . . . . . . . . . . :       _______
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . . :       ____
   Device  . . . . . . . . . . . . . . . . . . . :       ____

Type choice, press enter

  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    ______________________
                                                    3=Continue
                                                    4=Format

F3=Exit            F12=Cancel
Tape or diskette loaded is blank
```

Select option 4 (Format), and press the Enter key.

g. If the tape unit cannot format the tape to the requested density, the following display is shown:

```
                         Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :        ____
Device model . . . . . . . . . . . . . . . . . . :        ____
Device serial number . . . . . . . . . . . . . . :        __________
Device address . . . . . . . . . . . . . . . . . :        ___-_______
I/O manager code . . . . . . . . . . . . . . . . :        8000 0001C
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller . . . . . . . . . . . . . . . :        ____
   Device  . . . . . . . . . . . . . . . . . . . :        ____

Type choice, press enter

  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    ______________________
                                                    3=Retry
                                                    ______________________

F3=Exit             F12=Cancel
```

Replace the tape with a tape that can be formatted to the requested density and select option 3 (Retry). Press the Enter key.

h. If an unrecoverable tape media error has occurred, the following display is shown:

```
                         Save Storage

Status of save . . . . . . . . . . . . . . . . . :        Waiting

A media error occurred on tape.

To cancel the save storage, press F3 or F12.
To resume the save storage, do the following:
 1.  Remove the tape from the tape device.
 2.  Clean the tape path using the cleaning procedure
     described in the tape device operator's guide.
 3.  Insert a different tape into the tape device.
 4.  Make the tape device ready, if necessary.
 5.  Press Enter to continue.  The system will resume the
     save storage.
 6.  If the error occurs again on this tape, contact your
     next level of support.

F3=Exit             F12=Cancel
```

1) Remove the failing tape from the tape device. Do not put the failing tape with the other tapes that have already been saved during the save storage operation. The failing tape cannot be used during the restore storage operation.

2) Load a different tape in the tape device.

3) Press the Enter key to resume the save storage operation.

The data that is on the failing tape is written to the next tape so that no data is lost because of the tape media failure.

i. If the save storage operation was cancelled, and the operation can be resumed, the following display is shown.

```
                          Save Storage

Status of save . . . . . . . . . . . . . . . . . . :  Ended

Percent saved . . . . . . . . . . . . . . . . . . . :

The save storage was ended before the save was complete.

You may either resume the save storage, or cancel
the save storage and then start the save storage again from
the beginning.

To resume the save storage, press Enter.
To cancel the save storage, press F3 or F12.








F3=Exit        F12=Cancel
```

Press the Enter key to resume the save storage operation. The following display is shown.

```
                        Resume Save Storage

You have selected to resume the save storage.

Do the following:

 1. Locate the set of tapes created during the save storage
    which was interrupted. The last tape which was completely
    written before the save storage was interrupted has the
    following identification:
        Volume identifier . . . . . . . . :  ________
        Sequence number . . . . . . . . . :  ____

 2. Ensure that an initialized and write-enable tape is
    loaded and ready in the tape device. Follow the
    procedures described in the tape device operator's
    guide.

 3. Press Enter to resume the save storage.


F3=Exit        F12=Cancel
```

Load the tape that was being written to when the save operation was interrupted and press the Enter key.

j. If the save storage operation was cancelled, and the save operation cannot be resumed, the following display is shown.

```
                              Save Storage

Status of save . . . . . . . . . . . . . . . . . . :   Ended

Percent saved . . . . . . . . . . . . . . . . . . :   ___

The save storage was ended before the save was complete.

You must start the save storage again from the beginning to
have a complete system save.







F3=Exit         F12=Cancel
```

11. When the last tape is complete and no errors have occurred, the tape automatically rewinds and a normal IPL occurs.

12. The data area QSAVSTG in library QSYS is updated to show the date and time of the save operation. Use the Display Object Description (DSPOBJD) command to display the date and time of the save storage operation.

13. Use the Display Log (DSPLOG) command to display the QHST system log or use the Display Message (DSPMSG) command to display the QSYSOPR messages. Look at the save storage completion message CPC3734: `System storage save at &1. &2 sectors cannot be read.` If any sectors were found damaged and could not be read, call your service representative.

    ```
    DSPLOG QHST
    ```

    or

    ```
    DSPMSG *SYSOPR
    ```

This completes the save storage procedure. If you do not want the system to perform an automatic IPL, you can use an autostart job, which powers down the system.

Procedure for Resuming the Save Storage Operation

You can use this procedure only if the following conditions are true:

1. The save storage operation has finished saving the licensed internal code.
2. The save storage operation has completed writing to at least one tape during the save storage operation.
3. All disk units are attached and operating.

One method of resuming the save storage operation following a tape media error is discussed in step 10h1 on page 11-10 in the topic "Procedure for Saving Storage." If another type of error occurs that causes the save storage operation to end (for example, system power loss, operator error, or tape drive error), the save storage operation can be started again.

To start the save storage operation again after the system ends abnormally (for example power loss to the system), do the following:

1. Turn the keylock switch to the Manual position.
2. Turn the power on the system by pushing the Power switch up. (The switch returns to center after you push it.)
3. The IPL or Install the System menu is shown. Select option 3 (Use dedicated service tools (DST) and press the Enter key.)
4. Sign on DST using the password assigned to your system for full DST authority. The Use Dedicated Service Tools (DST) menu is shown.
5. From the Use Dedicated Service Tools (DST) menu, select option 9 (Work with save storage and restore storage) and press the Enter key.
6. Select option 4 (Resume save storage) and press the Enter key.
7. If resuming the save storage operation is not allowed, a display with an explanation is shown.
8. If the following display is shown, load the tape that was being written to when the save storage operation was interrupted and press the Enter key.

```
                          Resume Save Storage

You have selected to resume the save storage.

Do the following:

 1. Locate the set of tapes created during the save storage
    which was interrupted. The last tape which was completely
    written before the save storage was interrupted has the
    following identification:
        Volume identifier . . . . . . . . : ______
        Sequence number . . . . . . . . . : ___

 2. Ensure that an initialized and write-enable tape is
    loaded and ready in the tape device. Follow the
    procedures described in the tape device operator's
    guide.

 3. Press Enter to resume the save storage.


F3=Exit        F12=Cancel
```

9. If the volume identifier of the tape that is loaded is different from the volume identifier of the first save storage tape, the following display is shown.

```
                        Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . :        ____
Device model . . . . . . . . . . . . . . . . . :        ____
Device serial number . . . . . . . . . . . . . :        __________
Device address . . . . . . . . . . . . . . . . :        ___-____
I/O manager code . . . . . . . . . . . . . . . :        ________
Reference code . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . :        ____
   Device  . . . . . . . . . . . . . . . . . . :        ____

Type choice, press enter

  Action  . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                  2=Ignore
                                                  3=Continue
                                                  4=Format

F3=Exit           F12=Cancel
Wrong volume loaded
```

To continue the save operation, select option 2 (Ignore) and press the Enter key.

Procedure for Restoring Storage from the Save Storage Media

The restore storage procedure will reset the system back to the point when the save storage operation was run.

Task Overview

You will perform the following tasks during this procedure:

1. Power down the system
2. Install the licensed internal code
3. Restore the remaining save storage tapes
4. Restore the licensed internal code fixes
5. Restore changed objects and apply journal changes

Before You Begin . . .

- ☐ You will use function code 24 to install the licensed internal code.
- ☐ Clean the read and write head of the tape unit.
- ☐ Find the list of licensed internal code fixes that you created when you saved storage. This list should be with your SAVSTG tapes or with your backup log.
- ☐ If you applied any licensed internal code fixes since the last save storage operation, you will need your most recent cumulative PTF tape.

Use the following procedure to restore storage. The system will verify the sequences of the SAVSTG tapes during the restore operation to verify that the correct tape is loaded.

Note: The system will not be available for use until the restore storage process successfully completes.

If a tape media error occurs that is not recoverable, see the "Procedure for Resuming the Restore Storage Operation" on page 11-29. If the system cannot recover from the tape media error, you will have to load the system again using a previous SAVSTG tapes or restore the system using the standard restore commands with the last tapes used to save the entire system.

Task 1. Power Down the System

1. Type the following to power down the system:

   ```
   PWRDWNSYS OPTION(*IMMED)
   ```

Task 2. Restore the Licensed Internal Code

1. Ensure the key is in the keylock switch on the control panel.
2. Turn the key to the Manual position.
3. Press the Function Select switch to display **02** in the Function display on the control panel.
4. Press the Enter button on the control panel.
5. Select IPL type D (this specifies that the IPL source comes from tape) by pressing the Function Select switch on the control panel until **D** is shown on the Data display.
6. Press the Enter button on the control panel.
7. For the 9406 system unit, ensure that the power switches for the IPL tape unit and all disk units are in the On position.
8. Find the licensed internal code tape, which is the first volume of the most recent set of save tapes.
9. Place the tape in the tape unit used for the IPL. See the setup manual for the device for more information on loading the tape.

   Note: If your tape unit cannot be loaded when the power is off, continue with the next step. You will be prompted later by an SRC code for the tape.
10. Turn on the power to the system by pushing the Power switch up. The switch returns to center after you push it. The 9402 system unit has a green button labeled Power On.
11. Wait as explained below for the tape unit to power on. See the following explanations:

    Notes:

    a. SRC A100 1938 (Tape not found) may be displayed along with the System Attention light until the tape unit is turned on. Continue with the next step.

    b. SRC A100 1933 or 1934 (Tape not ready) may also be displayed along with the System Attention light. When the tape is ready, continue with the next step. If this SRC is displayed for more than 1 or 2 minutes, call your service representative.

c. SRC 2507 XXXX appears if manual select IPL intervention is required. If intervention is required, press the Function Select switch until 03 appears. Press the Enter button on the control panel.

12. Ensure that the tape is online or ready. No action is required for tape units that perform this step automatically (such as the tape cartridge unit).
13. Ensure that the console display is turned on.
14. Wait for the yellow System Attention light on the control panel to light up.

    There is a delay while the system loads information from the tape. SRCs showing status are continuously updated on the control panel while processing occurs. This can take from 5 to 20 minutes; the time varies depending on the speed of the tape unit and the processor speed for the specific system model.

    When SRC A6xx 6001 is displayed, the system is prepared to start installing or restoring the licensed internal code on the disk unit containing unit 1. Continue with the next step.
15. Select the correct function code by pressing the Function Select switch on the control panel until the correct function code is displayed. (See the "Before You Begin" section in this topic to determine the correct function code.)

    Warning: Option 24 (Install) is used only to recover from the loss of unit 1 in the system ASP, or to recovery from a disaster. Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

    If you selected function code 24, the System Attention light light may appear in one or two minutes and SRC A6xx 6002 is shown. If you are sure you want to install the licensed internal code, select function code 24 again.

    Notes:

    a. When SRC A6xx 6002 is displayed, option 23 can also be selected. If option 23 is selected, a restore of the licensed internal code is performed (not an install). Data on the disk units will not be lost.

    b. If another SRC appears after A6xx 6001 that is not in the A6xx xxxx format, then the system needs additional attention. Call your service representative.

       If the following SRCs appear after SRC A6xx 6001 is displayed, see Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" for an explanation of these SRCs and the steps to follow.

       - A6xx 6002 Disk unit may contain a valid system
       - A6xx 6003 Disk unit not currently a load source
       - A6xx 6004 Disk unit not currently a load source
       - A6xx 6005 Disk unit not found
16. Press the Enter button on the control panel.

    Note: The system will start displaying status SRCs again, which will be continuously updated to show the status of installing the licensed internal code. An example of a status SRC is D6xx 6201 (stand-alone install operations is running).

17. If the yellow System Attention light is on again, and SRC A6xx 6048 (New tape volume needs to be loaded) appears, the system needs the next tape. The xx tells which volume needs to be loaded. Load the correct tape and make the device ready. The install operation will automatically continue.

    If SRC A6xx 6051 appears, the stand-alone function is requesting the model-unique licensed internal code tape found in the service kit. Unload the current tape from the tape device and load the model-unique licensed internal code tape.

    - A6xx 6051 Model-unique licensed internal code tape needs to be loaded
    - A6xx 6052 Tape loaded was not the model-unique licensed internal code,

    If another SRC A6xx xxxx is displayed, look up the displayed SRC in Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" on page A-1 and follow the instructions. For all other SRCs call your service representative.

18. After the install or restore operation of licensed internal code or the model-unique licensed internal code is complete, the system will automatically perform an IPL.
19. Remove the model-unique licensed internal code tape from the tape unit (if it is loaded) when the IPL or Install the System menu is shown.
20. If you have other restore steps to perform and the model-unique licensed internal code tape was loaded, load the previous save tape.
21. Continue with "Task 3. Restore the Remaining Save Storage Tapes."

Task 3. Restore the Remaining Save Storage Tapes

1. After the IPL completes, the IPL or Install the System display appears.

```
                              IPL or Install the System

 Select one of the following:

              1. Perform an IPL
              2. Install the operating system
              3. Use Dedicated Service Tools (DST)
              4. Perform automatic installation of the operating system










 Selection
      3

 Licensed Internal Code is subject to the licensed
 granted in the Agreement for Purchase of IBM machines.
                                              © COPYRIGHT IBM CORP. 1980,
 1990.
```

2. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .    _____
```

3. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords. The Use Dedicated Service Tools (DST) menu is shown.

```
                     Use Dedicated Service Tools (DST)

Select one of the following:

           1. Perform an IPL
           2. Install the operating system
           3. Work with licensed internal code
           4. Work with disk units
           5. Work with DST environment
           6. Select DST console mode
           7. Start a service tool
           8. Perform automatic installation of the operating system
           9. Work with save storage and restore storage

Selection
     9

F3=Exit              F12=Cancel
```

4. Select option 9 (Work with save storage and restore storage) on the Use Dedicated Service Tools (DST) menu, and press the Enter key.

5. Select option 1 (Restore storage) and press the Enter key. The Specify Volume Identifier display is shown:

```
                         Specify Volume Identifier

 Type choice, press Enter.

  Volume identifier . . . . . . . . . .    _____
```

6. Type the volume name in the *Volume Identifier* prompt.

7. If the Select Tape or Diskette Unit display appears, select the proper unit and press the Enter key.

```
                         Select Tape or Diskette Unit

 Type option, press Enter.
   1=Select

                                             Serial
 Option      Type     Model                  Number          Address
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
   _         ____     ____                   __________      ____-______
                                                                          More...

 F3=Exit           F12=Cancel
```

8. If the wrong volume is loaded, the following display appears:

```
                         Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :          ____
Device model . . . . . . . . . . . . . . . . . . :          ____
Device serial number . . . . . . . . . . . . . . :          __________
Device address . . . . . . . . . . . . . . . . . :          ____-______
I/O manager code . . . . . . . . . . . . . . . . :          ________
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . . :          ____
   Device  . . . . . . . . . . . . . . . . . . . :          ____
Volume or file loaded  . . . . . . . . . . . . . :          ____

If the wrong volume was loaded, type change, press Enter.

Type choice, press enter
  New volume or file  . . . . . . . . . . . . . .           ________________






F3=Exit             F12=Cancel
Wrong volume loaded
```

Type the name of the correct volume or file, and press the Enter key. The following display is shown:

```
                    Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . :        ____
Device model . . . . . . . . . . . . . . . . . :        ____
Device serial number . . . . . . . . . . . . . :        __________
Device address . . . . . . . . . . . . . . . . :        ____-_______
I/O manager code . . . . . . . . . . . . . . . :        ________
Reference code . . . . . . . . . . . . . . . . :
   I/O controller . . . . . . . . . . . . . . . :        ____
   Device . . . . . . . . . . . . . . . . . . . :        ____

Type choice, press enter
  Action . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                   ______________________
                                                   3=Retry
                                                   ______________________

F3=Exit          F12=Cancel
```

Select option 3 (Retry), and press the Enter key.

9. There is a delay while the tape is read to determine what has been saved on the tape. The Confirm Restore of Storage display is shown.

```
                    Confirm Restore of Storage

Warning: A restore of storage will destroy the current
data on the system.  The restore will take several minutes for
each unit saved.  An automatic IPL is part of the restore.

Press F10 to confirm your choice to restore all disk units.
Press F12 to return to change your choice.

---------------- Restore To ---------------------  ------ Saved From ---------
                       Serial                      Serial
Unit  ASP  Type  Model  Number      Address        Number      Address
___   __   ___   ___    ________    ____-______    ________    ____-______
___   __   ___   ___    ________    ____-______    ________    ____-______
___   __   ___   ___    ________    ____-______    ________    ____-______
___   __   ___   ___    ________    ____-______    ________    ____-______
___   __   ___   ___    ________    ____-______    ________    ____-______
___   __   ___   ___    ________    ____-______    ________    ____-______
___   __   ___   ___    ________    ____-______    ________    ____-______
___   __   ___   ___    ________    ____-______    ________    ____-______
                                                                   More...

F10=Confirm restore          F12=Cancel
```

If the save storage operation was done on a system using checksum protection, and the disk configuration on the restoring system does not match the configuration of the saved system, the following display is shown instead.

```
                    Confirm Restore of All Disk Units

 Warning: A restore of all disk units will destroy the current
 data on the system.  The restore will take several minutes for
 each unit saved.  The saved configuration has checksum protection
 that can not be supported on the current disk configuration.
 Checksum protection will be stopped.  An automatic IPL is part
 of the restore.

 Press F10 to confirm your choice to restore all disk units.
 Press F12 to return to change your choice.

---------------- Restore To ---------------------  ----- Saved From --------
                       Serial                          Serial
Unit  ASP  Type  Model  Number       Address           Number       Address
____  __   ____  ___   _________   ____-______      _________   ____-______
____  __   ____  ___   _________   ____-______      _________   ____-______
____  __   ____  ___   _________   ____-______      _________   ____-______
____  __   ____  ___   _________   ____-______      _________   ____-______
____  __   ____  ___   _________   ____-______      _________   ____-______
                                                                    More...

 F10=Confirm restore          F12=Cancel
```

10. Press the F10 (Confirm restore) key to confirm.
11. The restore status display on the console continually displays the progress of the restore operation.

```
                         Restore Storage

Status of the restore. . . . . . . . . . . . . . . :  Running

Percent restored . . . . . . . . . . . . . . . . . :
```

The *Percent restored* field on the display indicates the percentage of the total amount of disk sectors that have been restored.

12. If no errors occur, the system performs a programmed IPL when the restore storage process completes.
13. One of the following displays may be shown during the restore storage.
    a. If the tape unit is not ready, the following display is shown:

```
                    Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :         ____
Device model . . . . . . . . . . . . . . . . . . :         ____
Device serial number . . . . . . . . . . . . . . :         __________
Device address . . . . . . . . . . . . . . . . . :         ___-_______
I/O manager code . . . . . . . . . . . . . . . . :         ________
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . . :         ____
   Device  . . . . . . . . . . . . . . . . . . . :         ____

Type choice, press enter
  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    ______________________
                                                    3=Continue
                                                    ______________________

F3=Exit             F12=Cancel
Tape unit not ready
```

Make the tape unit ready, select option 3 (Continue), and press the Enter key.

b. When the restore operation for a tape is complete, the tape is rewound and the following display appears:

```
                    Handle Tape or Diskette Intervention

Device:
 Type. . . . . . . . . . . . . . . . . . . :       ____
 Model . . . . . . . . . . . . . . . . . . :       ____
 Serial number . . . . . . . . . . . . . . :       __________
 Address . . . . . . . . . . . . . . . . . :       ___-________
I/O manager code . . . . . . . . . . . . . :       ________
Reference code . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . :       ____
   Device  . . . . . . . . . . . . . . . . :       ____
Volume or file . . . . . . . . . . . . . . :
   Loaded  . . . . . . . . . . . . . . . . :       _________________
   Requested . . . . . . . . . . . . . . . :       _________________

Type choice, press Enter.

  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    ______________________
                                                    3=Continue
                                                    ______________________

F3=Exit             F12=Cancel
End of tape encountered.  Load next volume.
```

Load the next tape, select option 3 (Continue), and press the Enter key.

c. The system determines the next tape by the volume identifier you specified at the beginning of the procedure. If the wrong tape is loaded, the following display appears:

```
                        Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :        ____
Device model . . . . . . . . . . . . . . . . . . :        ____
Device serial number . . . . . . . . . . . . . . :        __________
Device address . . . . . . . . . . . . . . . . . :        ___-_______
I/O manager code . . . . . . . . . . . . . . . . :        _______
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . . :        ____
   Device  . . . . . . . . . . . . . . . . . . . :        ____

Type choice, press enter
  Action  . . . . . . . . . . . . . . . . . . . . . 1=Cancel
                                                    _____________________
                                                    3=Retry
                                                    _____________________

F3=Exit             F12=Cancel
Wrong volume loaded.
```

Remove the tape, find the correct tape, select option 3 (Retry), and press the Enter key.

d. If the tape could not be read because of a media error, the following display is shown:

```
                          Restore Storage

Status of restore  . . . . . . . . . . . . . . . :       Ended

A media error was found on tape.

If this is the first time the restore storage has ended because
a media error occurred on this tape, do the following:
 1.  Remove the tape from the tape device.
 2.  Clean the tape path using the cleaning procedure
     described in the tape device operator's guide.
 3.  Press Enter, F3, or F12 to continue.  The system will
     perform an IPL, an then display either the IPL or Install
     the System menu or the Missing Disk Units display.
 4.  Select the option to use Dedicated Service Tools (DST)
 5.  Select the option to Work with Save Storage and Restore
     Storage.
 6.  Select the option Resume restore storage.
 7.  Insert the tape which had the media error into the tape
     device.
 8.  Make the tape device ready, if necessary.


F3=Exit             F12=Cancel
```

For information on how to recover from a media error, see the "Procedure for Resuming the Restore Storage Operation" on page 11-29.

14. When the IPL completes the restore operation, the IPL or Install the System menu appears.

```
                          IPL or Install the System

 Select one of the following:

      1. Perform an IPL
      2. Install the operating system
      3. Use Dedicated Service Tools (DST)
      4. Perform automatic installation of the operating system

 Selection
      _

 Licensed Internal Code is subject to license
 granted in the Agreement for Purchase of IBM machines.
                                       (C) COPYRIGHT IBM CORP 1980, 1990.
```

15. Select option 1 (Perform an IPL).

    If disk units have been attached to the system that are not part of the system disk configuration, the following display appears:

```
                       Add All Disk Devices to the System

 Select one of the following:

           1. Add all disk devices to the system auxiliary pool
           2. Keep the current disk configuration
           3. Perform disk configuration using DST

 Selection
      _
```

16. Select option 1 (Add all disk devices to the system auxiliary pool) if the disk devices do not have checksum protection or mirrored protection, and press the Enter key.

    As the disk devices are configured, the following display is shown:

```
Formatting additional disk devices in progress.
        Seconds running . . . . :  ____
```


17. When the IPL completes, the restore history information for data area QSAVSTG in library QSYS is updated to show the date and time of the last restore storage operation. Use the Display Object Description (DSPOBJD) to display the last date and time of the restore storage operation.
18. Use the Display Log (DSPLOG) command to display the QHST log or use the Display Messages (DSPMSG) command to display the QSYSOPR messages. Look at the restore storage message CPF3735 to see if any sectors that could not be read were found during the restore operation, if checksum protection was stopped, and if the restore storage process is complete.

Task 4. Restore the Programming Temporary Fixes

If you applied program temporary fixes to your system after the last save storage operation, do the following steps. Otherwise, continue with "Restoring Changed Objects."

If you are using this procedure to finish restoring the entire system and have applied program temporary fixes (PTFs) after the last save operation, you must restore the PTFs.

If you need to restore program temporary fixes (PTFs) as part of recovering the entire system, find the most recent cumulative program temporary fix tape. This package could be on a distribution tape or on a stand-alone tape.

You can use option 31 (Install program temporary fix package) on the Work with Licensed Programs menu to install all of the PTFs. All of the PTFs in the cumulative PTF package will be installed for the licensed programs you have installed on your system. Refer to the *AS/400 System PTF Shipping Information Letter* for special instructions that are required.

If you are restoring individual PTFs, see the *System Operator's Guide*, SC41-8082, for more information about applying individual PTFs.

1. Print a list of all the licensed internal code PTFs currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(5738999) OUTPUT(*PRINT)
   ```

2. Compare the list of PTFs in the previous step with the list you printed when you saved the system.

   Any PTFs that are not found must be loaded again.

3. If you do not have the PTFs identified in the previous step, order them and then apply them.

Task 5. Restoring Changed Objects and Applying Journal Changes

Use the following steps to restore user profiles, changed objects, and authority.

1. Type QSECOFR in the user prompt and the password (if password security is active) associated with that user ID on the Sign On display.
2. Press the Enter key.
3. Type the following command on the command line and press the Enter key.

   ```
   CHGMSGQ QSECOFR *BREAK SEV(60)
   ```

4. Type the following command to end all subsystems.

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

5. A message will be sent indicating that all subsystems are ended and the system is in a restricted state.
6. Type the following command to restore user profiles from the SAVSECDTA tape file (label QFILEUPR).

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

If you are using journaling and need to later apply journaled changes, continue with the following steps. Otherwise ignore these steps and continue with "Restoring Changed Objects."

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily save operations using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform the steps in "Working with Journals" for the system supplied journal QUSRSYS/QAOSDIAJRN. If you specified OBJJRN(*YES) on the SAVCHGOBJ command, you do not need to apply journal changes.

Working with Journals

1. Type the following and press the Enter key:

   ```
   WRKJRN
   ```

2. The Specify Journal Name display is shown. Specify *ALL for the *Library name* prompt and press the Enter key.
3. The Work with Journals display is shown. Type a **5** in the *Opt* field for each journal that you want to apply journaled changes to display the name of the currently attached journal receiver. Write down the name of the journal and journal receiver.

   Note: If you are using OfficeVision/400 or PC Support/400, you must apply journaled changes to the system supplied journal QUSRSYS/QAOSDIAJRN.

4. You cannot restore journal receivers from the SAVCHGOBJ media if active journal receivers are attached. To later apply all journaled changes that have occurred since the last complete save operation, you must restore the receivers to the system from the SAVCHGOBJ media.

For each journal identified in the previous step, do the following steps:

a. Create a journal receiver that will be used as a temporary receiver. Give it a name that will identify it as a temporary receiver, for example, TEMPnn. You can enter a description in the text (TEXT parameter) that identifies it as a temporary receiver for disaster recovery.

```
CRTJRNRCV JRNRCV(library-name/TEMPnn)
      TEXT('temporary journal receiver for journal xxx')
```

b. To detach the current receiver and attach the new TEMPnn receiver, type the following and press the Enter key.

```
CHGJRN JRN(library-name/journal-name) JRNRCV(library-name/TEMPnn)
```

c. Delete the detached journal receiver (identified in the step where you wrote down the name of the journal and journal receiver) using the Delete Journal Receiver (DLTJRNRCV) command. (This allows the journal receivers on the SAVCHGOBJ media to be restored successfully.

```
DLTJRNRCV JRNRCV(library-name/journal-receiver)
```

If you receive message CPA7025 *Receiver never fully saved*, enter an **I** to ignore and press Enter to continue the delete.

Restoring Changed Objects

Load the SAVCHGOBJ tape and enter the following to restore changed objects.

```
RSTOBJ OBJ(*ALL) DEV(tape-device) SAVLIB(library-name)
         OBJTYPE(*ALL) ENDOPT(*LEAVE) MBROPT(*ALL) ALWOBJDIF(*ALL)
```

You must repeat this step for every library saved using the SAVCHGOBJ command. If you specified SAVCHGOBJ LIB(*ALLUSR), type the following to determine the libraries that were saved:

```
DSPTAP *SAVRST
```

If you are using journaling, perform the steps in "Applying Journal Changes" for each journal you wish to apply journal changes to. Otherwise, continue with "Restoring Changed Documents and Folders."

Applying Journal Changes

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily saves using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform this step for the system supplied journal QUSRSYS/QAOSDIAJRN.

1. Determine the name of the last journal receiver (the last receiver restored) by entering the following:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

2. Press F15 (Work with receiver directory) to display the last receiver with a status of SAVED or PARTIAL. Write down the name of the receiver.

3. Create a receiver that follows the same naming convention as the last receiver but assign it a number of one greater.

   ```
   CRTJRNRCV JRNRCV(library-name/journal-receiver-nameNN)
   ```

   By doing this, you are doing what the CHGJRN command would normally do if the receiver was the current receiver being detached with a new receiver name being created. This allows your naming convention for journal receivers to continue.

4. Use the CHGJRN command to detach the temporary receiver and attach the new receiver you just created.

```
CHGJRN JRN(library-name/journal-name)
      JRNRCV(library-name/journal-receiver-nameNN)
```

5. Determine the chain of receivers to be used in the APYJRNCHG command by entering the following command:

```
WRKJRNA JRN(library-name/journal-name)
```

Press F15 (Work with receiver directory) to display the receivers. Write down the first and last receiver that you restored (last receiver is prior to the TEMPnn receiver). Notice that the first and last is the same receiver if only one journal receiver was restored.

6. When applying journal changes and the ending receiver has a status of PARTIAL (saved while attached), the TOENT parameter requires that a sequence number be specified on the APYJRNCHG command. Determine the last entry to be applied by entering the following command for the last receiver (identified in previous step):

```
DSPJRNRCVA JRNRCV(library-name/last-journal-receiver-name)
```

Write down the value for the *Last Sequence Number* field.

7. Enter the following command to apply the journaled changes using the first and last journal receivers identified on the Work with Receiver Directory display.

```
APYJRNCHG JRN(library-name/journal-name)
         FILE((library-name/*ALL))
         RCVRNG(lib-name/first-receiver lib-name/last-receiver)
         FROMENT(*LASTSAVE) TOENT(last-entry)
```

Attention

Do not apply journal changes to the document and folder search index database files (QAOSSH10 through QAOSSH19) for journal QAOSDIAJRN in library QUSRSYS. You must specify individual files on the FILE parameter instead of *ALL.

```
APYJRNCHG JRN(QUSRSYS/QAOSDIAJRN)
         FILE((QUSRSYS/QAOSAH05) (QUSRSYS/QAOKLY02)
              (QUSRSYS/QAOSAH07) (QUSRSYS/QAOKLY03)
              (QUSRSYS/QAOKDYX4) (QUSRSYS/QAOKNY06)
              (QUSRSYS/QAOKDY01) (QUSRSYS/QAOKDY04)
              (QUSRSYS/QAOKDY05) (QUSRSYS/QAOKDY07)
              (QUSRSYS/QAOKDY08) (QUSRSYS/QAOKDY09))
         RCVRNG(lib-name/first-receiver lib-name/last-receiver)
         FROMENT(*LASTSAVE) TOENT(last-entry)
```

Restoring Changed Documents and Folders

If you performed daily save operations for documents and folders, do the following steps. Otherwise, continue with the RSTAUT command.

1. Load the last daily SAVDLO tape.
2. If you performed daily save operations to backup all new folders and new and changed documents since the last complete SAVDLO operation, type the following and press the Enter key.

```
RSTDLO DLO(*ALL) DEV(TAP01) SAVFLR(*ANY) ALWOBJDIF(*ALL)
```

If you saved all documents referred to by the mail using SAVDLO DLO(*MAIL), type the following and press the Enter key.

```
RSTDLO DLO(*MAIL) DEV(TAP01)
```

3. Restore users' authority by entering:

   ```
   RSTAUT
   ```

   The time it takes for the RSTAUT command to complete can vary significantly. The time depends on the number of user profiles and private authorities that were saved during the save operation.

4. This completes the restore operation.

   Power down the system by entering the following:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES)
   ```

   to perform a normal IPL and return the system to normal operations.

   Note: If you turned auto configuration off during the restore of the operating system, you must perform an IPL with the keylock switch in the **Normal** position.

Save and Restore

Restoring Storage from a Non-IPL Tape Unit

The save storage operation can be used to write to a tape device that is not the IPL tape unit. For example, the 9404 has an optional 9348 tape unit. (The 6346 tape unit is the IPL tape unit on the 9404.)

The tape used for the save storage operation on the 9348 tape unit cannot be used to restore the licensed internal code. However, it can be used to restore the remaining parts of the system.

When the save storage operation is done using a tape unit other than the IPL tape unit, you should perform a SAVSYS operation using the tape unit used for the IPL. It is important that the operations occur without applying PTFs for the licensed internal code between the time of the SAVSTG operation and the SAVSYS operation. Otherwise, the restore operation will cause the PTF index to reflect the time of the SAVSYS and any PTFs applied since the last SAVSYS will have to be installed again.

To recover the entire system, and restore the licensed internal code from the IPL tape unit, use the SAVSYS tapes to restore the licensed internal code from the IPL tape unit. Sign on to DST after the restore of the licensed internal code is complete and request the restore storage function. Specify the non-IPL tape device for the restore storage operation.

Procedure for Resuming the Restore Storage Operation

You can use this procedure to resume the restore storage operation that ended before the entire restore operation of the disk unit data was complete.

To start the restore storage operation again, do the following:

1. From the Dedicated Service Tools (DST) menu, select option 9 (Work with save storage and restore storage) and press the Enter key.
2. Select option 2 (Resume restore storage) and press the Enter key.

3. If the following display is shown, load the tape that is indicated and press the Enter key.

```
                         Resume Restore Storage

Do the following:

 1.  Locate the tape to resume the restore on.  The tape
     that was being read when the restore storage was
     interrupted has the following identification:
        Volume identifier  . . . . . . . . :  ________
        Sequence number  . . . . . . . . . :  ____
 2.  Insert the tape in the tape device.
 3.  Make the tape device ready, if necessary.

Note:
  If the restore storage was interrupted because of a media
  error on a tape, you may want to resume the restore
  storage on the tape following the failing tape.  If you
  resume the restore storage on that tape, the system will
  have damaged objects, and the system might not be able to
  perform and IPL to OS/400 when the restore storage is complete.

Press Enter to continue.

F3=Exit        F12=Cancel
```

4. If the wrong volume is loaded, the following display appears:

```
                      Device Intervention Required

Device type. . . . . . . . . . . . . . . . . . . :      _____
Device model . . . . . . . . . . . . . . . . . . :      _____
Device serial number . . . . . . . . . . . . . . :      ____________
Device address . . . . . . . . . . . . . . . . . :      ____-________
I/O manager code . . . . . . . . . . . . . . . . :      _________
Reference code . . . . . . . . . . . . . . . . . :
   I/O controller  . . . . . . . . . . . . . . . :      _____
   Device  . . . . . . . . . . . . . . . . . . . :      _____
Volume or file loaded  . . . . . . . . . . . . . :      _____

If the wrong volume was loaded, type change, press Enter.

Type choice, press enter
  New volume or file  . . . . . . . . . . . . . .       _____________________




F3=Exit             F12=Cancel
Wrong volume loaded
```

Type the name of the correct volume or file, and press the Enter key.

5. The restore storage operation starts again.

If the restore storage operation continues to fail on the same tape with a tape media failure, you have three options:

- Use a previous copy of your save storage tapes to completely restore storage.

- Use the most recent copy of your save system tapes used to save the entire system.
- Start the restore storage operation again on the tape following the tape with the media error. If the tape that has the media error is the last tape to restore in the set, option 3 (Force end of an interrupted restore storage) on the Restore Storage menu should be selected.

  Warning: In either instance, some disk unit data is not restored. There may also be many objects damaged on the system when the restore operation completes. An initial program load of the operating system may not occur. You should restore the operating system again.

Part 3. Journaling and Commitment Control

Chapter 12. Database Recovery

This chapter describes the journal management function and how you use it to recover and back up database files. For information on journaling access paths, refer to topic "Access Path Recovery" on page 12-26.

Journal Management

Two objects are associated with journal management. The following is a description of the objects and how they are used:

- A **journal receiver** (*JRNRCV). The journal receiver is an object that contains the entries (called **journal entries**) written when you make a change. These entries include:
  - The after-image of each record changed
  - Optionally, the before-image of each record changed
  - System-created entries
  - Any user-created entries
- A **journal** (*JRN). The journal is an object that identifies the files and access paths being protected. The system also uses the journal to record information about the journal receivers and the database files being journaled.

Journal receivers are attached to a journal on the create journal (CRTJRN) and change journal (CHGJRN) commands. Journal entries are written to the attached receivers. Journal receivers that have been attached to a journal and are still known to the system are considered to be associated with that journal. Use the Work with Journal Attributes (WRKJRNA) command to see a list of receivers associated with a journal.

The system creates an entry in the attached journal receiver when you change a database file member. Each entry is sequentially numbered and contains information that identifies:

- Type of change
- Record that has been changed
- Change that has been made to the record
- Information about the change (such as the job being run and the time of the change)

Changes to physical files (whether made directly to the physical file or through a logical file) and access paths are written to the journal. The system does not journal data that you retrieved but did not change. If the logical file record format does not contain all the fields that are in the dependent physical file record format, the journal entry still contains all the fields of the physical file record format. In addition, if you are journaling access paths, entries for those access paths are written to the journal.

Figure 12-1 on page 12-2 shows journal processing. Files A and B are being journaled; file C is not. Programs PGMX and PGMY use file B. When you make a change to a record in file A or B, the following occurs:

- The change is written in the active journal receiver.
- The journal receiver is written to auxiliary storage.
- The record is written to the file.

File C record changes are written directly to the main storage copy of the file because it is not being journaled. Only the changes made to the journal receiver are written immediately to auxiliary storage. Changes against the physical file may stay in main storage until the file is closed.

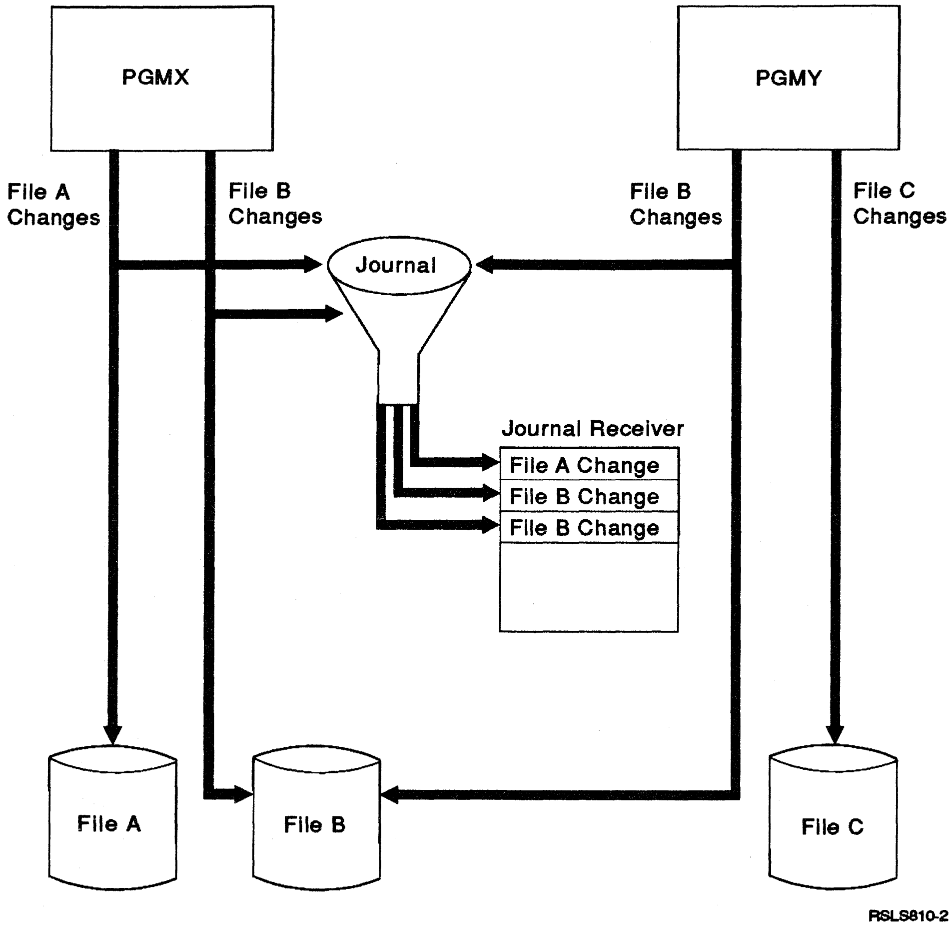


Figure 12-1. Journaling Overview

Journal Management Commands

The following commands are used for journal management. These commands are used to create journal objects and use the functions of journal management. For a complete description of these commands, see the *Programming: Control Language Reference*, SC41-0030.

Journal: Use these commands for journals:

Command	Meaning	Description
CHGJRN	Change Journal	Changes the attributes of a journal and attaches new journal receivers to the journal.
CRTJRN	Create Journal	Creates a journal.
DLTJRN	Delete Journal	Deletes a journal.
WRKJRNA	Work with Journal Attributes	Displays the attributes of a journal.

Journal Receiver: Use these commands for journal receivers:

Command	Meaning	Description
CRTJRNRCV	Create Journal Receiver	Creates a journal receiver.
DLTJRNRCV	Delete Journal Receiver	Deletes a journal receiver.
DSPJRNRCVA	Display Journal Receiver Attributes	Displays the attributes of a journal receiver.

Journal Entries: Use these commands for journal entries:

Command	Meaning	Description
CMPJRNIMG	Compare Journal Images	Compares the before-images and the after-images of record-level changes in a file member and indicates where differences occur.

Command	Meaning	Description
DSPJRN	Display Journal	Displays the journal entries that are in the journal receivers associated with the specified journal.
RTVJRNE	Retrieve Journal Entry	Retrieves a journal entry and places it in CL program variables.
RCVJRNE	Receive Journal Entry	Allows a specified user program to continuously receive journal entries one at a time. Can be used to provide backup on another system.

Files: Use these commands to start and end journaling a file or access path. For more information on journaling access paths, see the topic "Access Path Recovery" on page 12-26.

Command	Meaning	Description
STRJRNPF	Start Journal Physical File	Starts journaling for the physical file.
ENDJRNPF	End Journal Phyical File	Ends journaling for the physical file.
STRJRNAP	Start Journal Access Path	Starts journaling access paths.
ENDJRNAP	End Journaling Access Path	Ends journaling access paths.

Database File Member: Use these commands to recover a database file member using the journaled changes:

Command	Meaning	Description
APYJRNCHG	Apply Journaled Changes	Applies the changes to the designated physical file member that were recorded in a journal receiver associated with the journal.
RMVJRNCHG	Remove Journaled Changes	Removes the changes from the designated physical file member that were recorded in a journal receiver associated with the journal (allowed only if before-images and after-images are journaled for the file).

User-Created Entries: Use this command to add user-created entries to a journal:

Command	Meaning	Description
SNDJRNE	Send Journal Entry	Places a user-created entry in the journal.

Journal Functions: Use this command to perform journal functions, particularly, recovery functions:

Command	Meaning	Description
WRKJRN	Work with Journals	Displays the journal menu from which you can perform system assisted recovery and other journal operations.

Journaling and Commit

Using One or More Journals

Using one journal allows you to establish standard operational procedures for files with similar backup and recovery needs, and makes it easier for you to manage the journal and the journal receivers. If you are journaling to a user auxiliary storage pool (ASP), a single journal receiver for each ASP provides the best performance. For more information about user ASPs, see "User ASPs" on page 14-8.

You can specify after-images for some files, and simultaneously specify both before-images and after-images for other files, even though they use the same journal. All files under commitment control in the same commitment control environment must be journaled to the same journal. For more information on commitment control, see the topic "Commitment Control" on page 12-28.

To start journaling for a file, the library of the file and the library of the journal must be in the same ASP. The journal receivers do not need to be in the same ASP as the journal and the files being journaled.

Use more than one journal if one or more of the following are true:

- Some files have different backup and recovery requirements. For example, if the files for your major application programs are in separate libraries and the libraries are saved on different schedules, journal the files for each application program to separate journals.

- The security of certain files requires that you exclude their recovery and backup procedures from the procedures for other files.
- Some files must be treated separately from others because they need to be transferred from one system to another or replaced on the system periodically.
- You are journaling the files for different reasons, such as some for audit or activity trail purposes, and some for debugging purposes.
- You have many application programs that use different files. Journaling all the files to one journal decreases performance because of journal entries being written to the journal at the same time. However, if you have a journal receiver isolated on a single ASP, you may achieve better performance with a single journal.
- Job accounting must be done to the QACGJRN journal in library QSYS. You should not mix database file journaling with the accounting journal.
- Security auditing must be done to the QAUDJRN in library QSYS. You should not mix database journaling with the security auditing journal. For more information about security auditing, see the manual *Security Concepts and Planning*, SC41-8083. For more information about job accounting, see the manual *Programming: Work Management Guide*, SC41-8078.

Naming Journal Receivers

Use generic names to help identify the use of each receiver. When you use the CHGJRN command, you can specify that the system create and name the new receivers by specifying JRNRCV(*GEN). In this case, the system adds a number to the name or increases by one the number already in the attached journal receiver name.

The system uses the following rules to append a number to the name of a currently attached journal receiver:

- If the name of the current journal receiver does not end in a number and does not exceed 6 characters, the system appends 0001.
- If the name of the current journal receiver exceeds 6 characters, and the last four characters are not numeric, the system truncates the name to 6 characters and appends 0001.
- If the name of the current journal receiver ends in a number and does not exceed 6 characters, the system adds 1.

For an example, see Table 12-1.

Table 12-1. System-Created Journal Receiver Names

Current Journal Receiver Name	System-Created Journal Receiver Name	Comments
A	A0001	
ABCDEF	ABCDEF0001	
ABCDEFG	ABCDEF0001	(The last character of the current name is dropped because it exceeds 6 characters.)
A0001	A0002	
A1	A2	
A9	A10	
ABCDEF1	ABCDEF0001	(The last character of the current name is dropped because it exceeds 6 characters.)
ABCDEF9999	Error	Adding 1 to 9999 causes an overflow condition.

Using Dual Journal Receivers

You can specify that a journal have two journal receivers attached to it at the same time. When you use two receivers, all entries are written to both receivers. The use of two receivers provides more protection from damage and should be considered if the changes to your files are not easily reconstructed. If one receiver is damaged, the system continues to write changes to the other receiver.

The following rules are used to name two journal receivers if JRNRCV(*GEN *GEN) is specified.

- If you currently have two journal receivers, the name of the first journal receiver is used for the first name, and the name of the second journal receiver is used for the second name. For example, if the names of the current journal receivers are A1 and B5, the names of the new receivers are A2 and B6.
- If you currently have a single journal receiver attached, the system uses the same naming convention for both new journal receivers. For example, if the current journal receiver is A1, the new journal receivers are A2 and A3.

If your journal entries are critical, you can have one receiver in one ASP and the other receiver in a different ASP. If a single disk unit fails, one receiver would still exist.

Journal Receiver Chains

Journal receivers associated with a journal (presently or previously attached to the journal) are linked in one or more chains. Each journal receiver, except the first one, has a previous receiver that was detached when the current receiver was attached. Each journal receiver, except the one currently attached, also has a next receiver. If you use two receivers, each journal receiver, except the first, has two previous receivers and each journal receiver, except the last, has two next receivers.

Figure 12-2 illustrates the process by which journal receivers are created. If you leave the previously attached receivers RCVA7 through RCVA9 online, you can use them to apply changes, to remove changes, or to display journal entries without restoring them first.

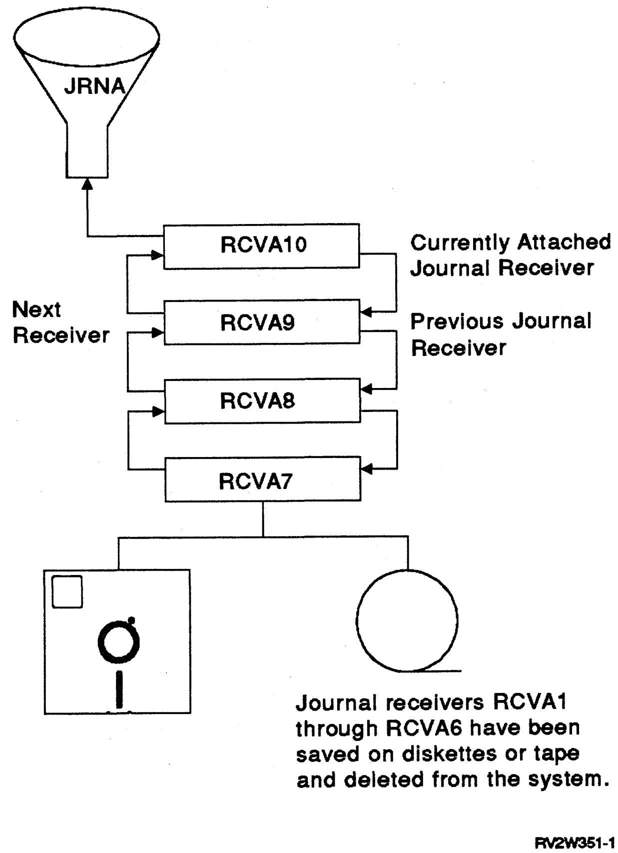


Figure 12-2. Creating a Journal Receiver

The APYJRNCHG, RMVJRNCHG, RCVJRNE, DSPJRN, RTVJRNE, and CMPJRNIMG commands cannot be used across multiple receiver chains. (If multiple receiver chains exist, run a separate command for each receiver chain.)

A logical break in a receiver chain is a receiver chain break. Receiver chain breaks should be avoided. A receiver chain break indicates that any changes made between the last receiver in one chain and the first receiver in the next chain are not available in any journal receiver on the system.

A set of receivers for a journal that has one or more receiver chain breaks has multiple receiver chains. Receiver chain breaks result from the following:

- You restored an old journal receiver and its next receiver is not on the system.
- A journal receiver was saved while it was attached, a partial receiver is restored, and no complete copy of the receiver is on the system or restored.

- A receiver that has not had its storage freed by a save operation is restored, and the next receiver has had its storage freed by a save operation.
- Both receivers in a set of two receivers are unusable, destroyed, or damaged.
- The journal is restored.
- A damaged or destroyed journal receiver was deleted from the middle of a chain, and dual receivers do not exist on the system.
- A journal receiver from another system is restored. The journal receiver will be associated with a journal at restore time if the associated library and journal on the source system had the same library name and journal name as the library and journal on the target system.

Changing Journal Receivers

The Change Journal (CHGJRN) command is used to make the following changes:

- Change receivers for a journal.
- Reset the sequence number.
- Change the journal threshold message queue.
- Change text for the journal.

Use the CHGJRN command to change the journal receiver for a given journal at any time, even if the files being journaled are open. As a new journal receiver is attached to the journal, the previously attached journal receiver is automatically detached. You can periodically change the journal receiver without interrupting normal system functions and without affecting the use of the files being journaled. No database changes are lost. However, a significant performance penalty is associated with each open file. The overhead is similar to the cost of closing a file. The system creates journal entries in both receivers indicating that a new receiver has been attached. Changing the journal receivers allows you to save and delete the detached journal receiver.

Avoid changing receivers during peak production periods because of the possible effect on other jobs. Changing to a new receiver can cause the system to force many database file changes to auxiliary storage. Thus, the time to change receivers can depend on the status of the files. It is best to schedule receiver changes during low production periods.

When you change from one receiver to another and the SEQOPT is *CONT, the sequence number for the first journal entry in the new receiver is one greater than the last sequence number in the detached receiver. Use SEQOPT(*RESET) to reset the starting sequence so that the first journal entry in the new receiver is 1. You cannot reset the sequence number unless any of the following are true:

- All journaled files are closed.
- All files under commitment control are at a commitment control boundary.
- The files are open and all changes to the files have been forced to auxiliary storage (by using the force end of data in an RPG program or the FRCRATIO of 1 on each file journaled).

In general, you should reset the sequence number only when the sequence number becomes extremely large. Resetting the sequence number has no effect on the naming of the journal receivers. See "Naming Journal Receivers" on page 12-4.

When you use JRNRCV(*GEN) on the Change Journal (CHGJRN) command, the system creates the new receiver with the same attributes as the currently attached receiver, and in the same library. These attributes include the owner, public authority, ASP identifier, threshold, and text.

Use the CHGJRN command to change from two receivers to a single receiver, or from a single to two receivers. If two receivers are used, additional protection is provided against the loss of entries. If you place one of the receivers in a separate user ASP, you are also protected against a disk device media failure.

Note: Entries are placed in both receivers if you are using dual receivers. Best performance is achieved by placing each receiver in a separate user ASP.

When a journal receiver is detached from a journal, the receiver cannot be reattached to any journal. The receiver can be used only with system functions such as the save and restore operations, or on the following commands:

- Apply Journal Changes (APYJRNCHG)
- Remove Journal Changes (RMVJRNCHG)
- Display Journal Receiver Attributes (DSPJRNRCVA)
- Compare Journal Image (CMPJRNIMG)
- Retrieve Journal Entry (RTVJRNE)
- Receive Journal Entry (RCVJRNE)
- Work with Journal (WRKJRN)
- Display Journal (DSPJRN)

Deleting Journals and Journal Receivers

If you no longer need a journal receiver, delete it using the Delete Journal Receiver (DLTJRNRCV) command. If you delete a journal receiver, you must restore it from a saved copy before the recovery functions Apply Journal Changes (APYJRNCHG) command or Remove Journal Changes (RMVJRNCHG) command) can use that receiver. The system does not allow you to delete a journal receiver attached to a journal.

The following command deletes the journal receiver JRNDST1 from the library DSTJRN:

```
DLTJRNRCV DSTJRN/JRNDST1
```

You can also delete a journal receiver using the Work with Journal Attributes display to select the Work with Journal Receiver Directory display.

Journal receivers must be deleted in the same order they were attached to a journal except that a damaged or inoperable receiver can be deleted regardless of this restriction. If an attached receiver is damaged, you must detach it (CHGJRN command) before you delete it. You can also delete a journal receiver in the middle of a chain if the corresponding dual receiver still exists and is usable.

You should ensure that the journal receivers are not deleted without first being saved. Display the journal receiver directory using the Work with Journal Attributes (WRKJRNA) command to determine which receivers have been saved. A date of 00/00/00 in the *Save date* column indicates that a journal receiver has not been saved.

If you attempt to delete a receiver that has not been saved, the system sends the inquiry message CPA7025. If the reply to the message is ignore (I), the journal receiver is deleted. If the reply is cancel (C), the operation is not completed. You can use the system reply list to specify the reply the system sends for a particular job.

If you no longer need a journal, you should delete it using the Delete Journal (DLTJRN) command. Each journal on the system causes additional time and resource to be used at each IPL after an abnormal end. If you delete a journal, you must restore it from a saved copy or create a new journal with the same name in the same library. Then restore all the needed receivers before using the APYJRNCHG or RMVJRNCHG command to apply or remove changes with that journal. Receivers already on the system need to be saved and restored to become associated with the journal.

The system does not allow a journal to be deleted if files are being journaled to it, or if commitment control is active and the journal is associated with it.

To delete a journal, perform the following steps:

1. End journaling of all logical files with the ENDJRNAP command.
2. End journaling of all files associated with the journal using the ENDJRNPF command.
3. If commitment control is active, and the journal is associated with it, end commitment control with the ENDCMTCTL command.
4. Delete the journal.

   The following command deletes the journal JRNLA from the library DSTJRN:

   ```
   DLTJRN DSTJRN/JRNLA
   ```

Saving Journals and Journal Receivers

To save or restore a journal, use these commands:

- Save Object (SAVOBJ)
- Save Changed Object (SAVCHGOBJ)
- Save Library (SAVLIB)
- Save Storage (SAVSTG)
- Save System (SAVSYS)
- Restore Object (RSTOBJ)
- Restore Library (RSTLIB)

Journal receivers attached to a journal can be saved without being detached from the journal. Journal receivers no longer attached can be saved with the storage freed, restored, or deleted.

Inoperable Journal Receivers

If you have specified journaling for any files, the system ensures that you have corrected problems that affect journaling before continuing with operations on those files. If the attached journal receiver becomes inoperable, the database operation that writes a journal entry is interrupted and the system sends an inquiry message notifying the system operator. The operator can use the CHGJRN command to change the journal receiver. The user can then respond to the inquiry message. A receiver can become inoperable if the receiver is damaged, the maximum sequence number has been reached, or there is no more space.

If dual receivers are attached to a journal and one receiver becomes inoperable, the system continues placing journal entries in the operable receiver and discontinues placing them in the inoperable receiver. The system sends a message notifying the operator that one of the receivers is inoperable. The system operator can use the CHGJRN command to replace the inoperable receiver, which causes both currently attached receivers to be replaced.

Planning Journal Management

Consider the following before you decide to use journal management:

- You can use journal management to do the following:
  - Recover a file member from some form of damage to the member.
  - Recover access paths after an abnormal system end.
  - Provide an audit trail of file or file member activity.
  - Analyze problems or for testing tools.
  - Provide an activity trail.
  - Review the security plans for the files.
- Do the following depending on the importance of the files being journaled:
  - Use two journal receivers for the journal.
  - Use online backup for the files. (The operation of copying a file using the CPYF command is not journaled in the journal, but the save operation to a save file is.)
  - Back up the journal, the journal receivers, and the journaled files on tape or diskette.
- Determine who is responsible for the following:
  - Recovery functions
  - Authorization to use the journal
  - Use of the journal
- Other items to consider include:
  - Legal considerations or restraints, such as who is allowed to look at the data
  - Business practices that influence your use of journal management, such as auditing or security requirements
  - Amount of system space to use journal management
  - Effect of journal management on the performance of your system

After considering the above, you may come to these conclusions:

- The recovery functions provided by the save and restore operations are adequate for your needs.
- You will use journal management for your recovery needs on some files.
- You will use journal management for reasons other than recovery.

Consider the following when you use journal management:

- Which physical files will be journaled?
- Which access paths for database files will be journaled?
- What attributes will you use to journal those files?
- Do you need multiple journals?
- Will you journal multiple files to the same journal?

- Should you place the journal receiver in a user ASP?
- How often will you save the journal?
- How often will you save the journal receivers?
- How often will you attach new journal receivers to the journal?
- How often will you save the physical files?
- What information will you make available for journal users?
- How will physical files be recovered?
- Will application programs place entries directly into the journal?
- What information should you record with user-created journal entries?

Managing Journal Receivers

Following are examples of how to manage journal receivers:

- Create all the physical files that are journaled in a particular application, the journal, and the associated journal receivers in the same library.

  At the end of each day, run a CL program that:

  1. Creates a new journal receiver.
  2. Attaches it to the journal.
  3. Saves the entire library with the SAVLIB command.

  Only one receiver is used throughout the day. Leaving the detached journal receivers online provides an easier recovery because the receivers do not have to be restored. However, if you have insufficient space, you can delete the detached receivers after you have saved them.

- Create all the physical files that are journaled in a particular application, the journal, and the associated journal receivers in the same library, and use the SAVCHGOBJ command instead of the SAVLIB command.

  At the end of each day, run a CL program that:

  1. Creates a new journal receiver.
  2. Attaches it to the journal.
  3. Saves the previous receiver and the journal, and saves the changed objects (SAVCHGOBJ command) in the library excluding the journaled files and the active receivers.

  You can then save the journaled files periodically (for example, weekly) by using the SAVCHGOBJ command and specifying OBJJRN(*YES) so that you have a starting point for recovery of the files.

- Create all the physical files that are journaled in a particular application in the same library, and create the journal and journal receivers in another library.

  At the end of each day, run a CL program that:

  1. Creates a new receiver.
  2. Attaches it to the journal.
  3. Saves the library with the journal and journal receiver.

  Then save the files on a regular, but not a daily basis. (Use the SAVLIB command to save the library that contains the files.)

  If you have sufficient space, leave the journal receivers online for other uses, such as audit trails or debugging. If you do not have the space, and do not need the receivers for any other purposes, save the detached receivers (specify STG(*FREE)) or delete them.

- If you have applications with a critical need to get a copy of the journal receivers offline as soon as possible, change one of the preceding approaches to save the currently attached receiver many times throughout the day, or run a CL program (or the CL commands) to save the attached journal receiver. If you use this approach, you should save the journal receiver as soon as it is detached and then discard all but the last back-level copy of the receiver. Keep the last back-level copy as a backup.

- If you change the journal several times a day, you should completely save the receiver. Use the Change Journal (CHGJRN) command only when system activity is at a minimum. By changing the journal receivers more frequently, the impact or decrease in performance is reduced. This is especially true when journaling access paths.

- If you wish to protect the journal from loss of data due to a disk device failure, create a user ASP for the journal receiver. Creating a journal receiver in a library on a user ASP can also improve performance of the journal receiver. For information on creating user ASPs, see "General Information about Auxiliary Storage Pools" on page 14-7.
- You should also place journal receivers in a different ASP if your journal entries are critical.
- If the data in your files is extremely critical and difficult to re-create, attach two receivers to the journal and place each receiver in a different ASP. If one of the journal receivers becomes inoperable, the system continues to use the other one. The CL programs to control the journal receivers are the same as described in the preceding examples except that two journal receivers are attached and detached with each CHGJRN command. These journal receivers can be saved on the same volume or on different volumes for more reliability.

  You can then perform file recovery using either one of the journal receivers if they are both usable.

In deciding how you are going to manage your journal receivers, you should also keep the following points in mind:

- If it is seldom necessary for you to recover a file, save the file less frequently, and save the journal receivers on a regular basis.
- If you use the CHGJRN command and specify JRNRCV(*GEN), you do not have to explicitly create a receiver.
- You do not need to delete the journal receivers as soon as they are saved. If you have sufficient space on your system for more than one receiver, keep the receivers online so they are available for use in producing audit trails, for problem analysis, and for recovery purposes. In this case, you do not have to restore the receivers when you need to do a file recovery.

  However, if your storage space is limited, save the journal receivers with the storage freed rather than deleting them to save space. Saving journal receivers with storage freed also facilitates restore and recovery procedures. (The system keeps track of all receivers that have not been deleted.) If you do not have enough space on your system, delete the receivers after they have been saved.
- If you anticipate transferring an application to another AS/400 system, saved journals and journal receivers can be restored only to the libraries they were originally created in. Move the application using the Save Library (SAVLIB) and Restore Library (RSTLIB) commands. However, in this situation, if the library becomes damaged, the entire library must be restored from a previously saved copy.

Managing System-Created Journals and Journal Receivers

It is recommended that periodically (monthly), large journal receivers should be detached, saved, and then deleted to free storage.

The following journals are shipped with the operating system. With some licensed programs, the OS/400 licensed program automatically adds entries to the journals. For example, OfficeVision/400, QSNADS, and QDSNX automatically add entries to the journals. The following is a list of journals used:

- QDSNXJRN (distributed services network journal)
- QAOSDIAJRN (document library and directory journal)
- QSNADS (Systems Network Architecture distribution services journal)
- QSXJRN (problem database journal)

Two journals can be optionally created:

- QAUDJRN (security auditing journal)
- QACGJRN (job accounting journal)

For more information about what each journal is used for, and a management strategy for each journal, see "Strategies for IBM-Supplied Journals" on page 3-12.

The journal receivers attached to the journal can become quite large because of the activities performed by certain products.

Before a journal receiver can be deleted, you must first detach it from the journal. Use the

Change Journal (CHGJRN) command to detach a journal receiver for a given journal.

As a new journal receiver is attached to the journal, the previously attached journal receiver is automatically detached. You can periodically change the journal and delete the journal receiver when the journal receiver becomes too large without affecting normal system operations and without affecting the use of the files. However, it is recommended that this operation not be performed during the time the system is at maximum use.

When you use JRNRCV(*GEN) on the CHGJRN command, the system creates the new receiver with the same values as the currently attached receiver, and in the same library. These values include the owner, public authority, ASP identifier, threshold, and text.

When you change from one receiver to another (when JRNRCV(*GEN) is specified), the last number in the new receiver name is one greater than the last number in the detached receiver. For example, if you detach journal receiver QAOSDI0001, the system created journal receiver is QAOSDI0002.

In the following example, the journal receiver QAOSDI0001 is being detached from journal QAOSDIAJRN. See "Naming Journal Receivers" on page 12-4.

You can display all journals on the system by using the Work with Journals (WRKJRN) command.

If you know the name of the journal, you can use the Work with Journal Attributes (WRKJRNA) command to determine which receivers are currently attached, which files are being journaled, and to provide a list of all receivers associated with the journal.

To display all journals on the system, do the following:

1. Type the following on a command line and then press the Enter key:

   ```
   WRKJRN
   ```

2. The Specify Journal Name display is shown. *ALL is the default for the journal. Specify *ALL for the *Library* field and press the Enter key.

3. The Work with Journals display is shown. A list of all the journals on the system are shown.

   All system-supplied journals start with the letter *Q*.

4. If you want to display the status of a specific journal, type **5** (Display journal status) in the *Opt* column and press the Enter key.

5. Use the following command (where Qxxxxxxxxx is the name of the journal) to detach a journal receiver from the journal and create a new journal receiver.

   ```
   CHGJRN JRN(Qxxxxxxxxn) JRNRCV(*GEN)
   ```

6. It is recommended that you save the journal receiver. If you want to save the detached journal receiver (where Qxxxxxxxxn is the name of the journal receiver), do the following:

   ```
   SAVOBJ OBJ(Qxxxxxxxxn) OBJTYPE(*JRNRCV) LIB(QUSRSYS)
          DEV(TAP01)
   ```

7. If you *do not* want to save the journal receiver but do want to delete it, do the following:

   ```
   DLTJRNRCV JRNRCV(QUSRSYS/QAODI0001)
   ```

   If the journal was not saved, enter I when the following message is shown.

   ```
   Receiver not fully saved. ( C I )
   ```

   Press the Enter key.

   This completes saving and deleting the journal receiver.

Managing a Journal

The frequency with which you perform the following steps and the methods you use depend on how you decide to manage journaling.

1. Attach a new journal receiver.
2. Save the detached journal receiver.
3. Periodically save the journal.
4. Periodically save the files and all associated logical files.

Do the above steps after you complete a recovery operation using journaling. Save a file after you add a member so that it can be recovered if needed. You should periodically save the journal and all logical files. Save the detached journal receiver after you run a CHGJRN command.

Journaling a Physical File

When you specify that a physical file is to be journaled, the system ensures that an active journal receiver exists, that all changes to members of the file are journaled through the same journal, and that new members added to the journaled file are automatically journaled. You use the Start Journaling Physical File (STRJRNPF) command once per file to have those changes journaled for every job that subsequently uses the file. When journaling of a file ends, an entry is recorded in the journal unless the receiver is damaged.

Changes for several physical files can be journaled to the same journal, but changes for any one physical file can be journaled to only one journal at a time. If a logical file is dependent on several physical files, it is recommended that all the physical files be journaled to the same journal.

In general, database source files should not be journaled. If you use the Start Source Entry Utility (STRSEU) command to update a member, every record in that member is considered changed and every record is journaled to the journal. However, if changes to a source file are critical, you can journal the file in the same manner as data files.

It is recommended that you do not journal changes to IBM-supplied database files. These files are sometimes created and managed differently than user-created files. The system does not assure the recovery of these files even though all recovery activity normally succeeds.

Saving Journaled Files

Save physical files after you start journaling them. Logical files dependent on the physical files, should also be saved in case the physical file becomes damaged. You do not need to save access paths being journaled in order for the system to recover those access paths after an abnormal system end. If you save the access paths being journaled (when saving files), the system does not have to rebuild the access paths after restoring the files and access paths. You can use the Save Changed Objects (SAVCHGOBJ) command and specify OBJTYPE(*FILE) OBJJRN(*YES), or you can use the Save Object (SAVOBJ) or Save Library (SAVLIB) command to save files.

Journaled changes are associated with files by a mark placed on the file at the time journaling was started. You should save the file after journaling is started because this allows the journaled changes to be applied to the file after the file is restored. When you save the files, ensure that no changes can be made to the files by using the Allocate Object (ALCOBJ) command, or by stopping activity on the files when they are being saved.

You should also save a journaled file after you add a member to the file so the member can be recovered in case it becomes damaged.

Journal Entries

Every journal entry is stored internally in a compressed format and must be converted by the operating system to an external form before it can be shown to the user.

You can use the Display Journal (DSPJRN) command to display or print the journal entries or write them to a database output file. You cannot directly access the journal entries. Not even the security officer can remove or change journal entries in a journal receiver. You can use these journal entries to help you recover your files or analyze changes made to the files.

You can write journal entries to tape or send them to another system using the Receive Journal Entry (RCVJRNE) command. This allows a user-defined program, called an **exit program**, to receive journal entries from the RCVJRNE command. The program writes the entries to tape or to an OS/400 intersystem communications function file (ICF) file that sends them to a backup system. When written to a backup source, you can use the entries to update the primary database or a backup copy. You cannot use these retrieved entries with system-supplied recovery commands (APYJRNCHG and RMVJRNCHG) to update your files because the RCVJRNE command converts the entries to their external form. You must write your own program to apply the changes contained in the entries to the database files.

The RCVJRNE command supports most of the parameters provided by the DSPJRN command. This allows you to specify which entries go to the exit program.

You can use the RTVJRNE command in a CL program to retrieve a journal entry and place it in program variables. See the topic "Retrieving a Journal Entry" on page 12-23 for more information.

You can use the CMPJRNIMG command to compare and list the difference between the before- and after-image of a record or between the current after-image of a record and the previous after-image of the record. See the topic "Comparing Journal Images" on page 12-24 for more information.

The format of the converted journal entry depends on the value specified for the OUTFILFMT parameter if the DSPJRN command is used, or the ENTFMT parameter if the RCVJRNE or RTVJRNE command is used. Each converted journal entry has a fixed length prefix portion followed by a variable length portion containing entry-specific data and associated information. The fields that make up the fixed length prefix of every converted journal entry are common to all entries and are described in the following pages.

The entry-specific data is either data for system-created entries or user-created data recorded in the journal by the SNDJRNE command. If the entry is system-created, the entry-specific data varies with the type of the journal entry. If the entry is a user-created entry, the entry-specific data is the data specified on the ENTDTA parameter of the SNDJRNE command. System-created entries are described in the following pages.

Contents of a Journal Entry

The entries created when a change is made to a member of a journaled file contain:

- Information identifying the type of change.
- Information identifying the record that was changed.
- The after-image of the record.
- Optionally, the before-image of the record (this is a separate entry in the journal).
- Information identifying the job, the user, the time of change, and so on.
- Information that identifies whether the file was opened, closed, reorganized, cleared or saved.

The system also places entries in the journal that are not for a particular file member. These entries contain information about the operation of the system and the control of the journal receivers.

Each journal entry is sequentially numbered without any breaks in the numbers until the Change Journal (CHGJRN) command resets the sequence number. However, when journal entries are converted and shown to the user, there may be breaks in the sequence numbers. The system uses some journal entries only internally and combines some entries into one during conversion.

When the system exceeds the largest sequence number (2 147 483 647), a message is sent to the system operator identifying the condition and requesting action. No other journal entries can be added to the journal until the journal receivers are changed and the sequence number is reset.

Fixed Length Portion of a Journal Entry: Table 12-2 on page 12-14 shows the information in the fixed length prefix portion of a converted journal entry if OUTFILFMT(*TYPE1) is requested on the DSPJRN command, or ENTFMT(*TYPE1) is requested on the RCVJRNE or RTVJRNE command. The field names shown in parantheses in the table are the names of the fields in the system-supplied output file QSYS/QADSPJRN.

Table 12-2 (Page 1 of 2). Field Descriptions of the Fixed Length Portion of a Journal Entry *TYPE1		
Field	**Format**	**Description**
Entry length (*JOENTL*)	Zoned decimal (5,0)	Specifies the length of the journal entry including the entry length field, all subsequent positions of the journal entry, and any portion of the journal entry that was truncated if the length of the output record is less than the length of the record created for the journal entry. This field only appears when the journal entries are written to the database output file.
Sequence number (*JOSEQN*)	Zoned decimal (10,0)	Assigned by journal management to each journal entry. It is initially set to 1 for each new or restored journal and is incremented by 1 until you request that it be reset when you attach a new receiver. There are occasional gaps in the sequence numbers because the system uses internal journal entries for control purposes.
Journal code (*JOCODE*)	Character (1)	Identifies the primary category of the journal entry: **A** = Job accounting entry **C** = Information about the commitment control environment **F** = File level information **J** = Information about journal receivers and operation of the journal **P** = Performance entry **R** = Information about a change to a specific record **S** = SNA distribution services (SNADS) entry **T** = Security-related events **U** = User-created entry (added to the journal by the SNDJRNE command)
Entry type (*JOENTT*)	Character (2)	Further identifies the type of user-created or system-created entry. See the following tables for more information.
Date stamp (*JODATE*)	Character (6)	Specifies the system date when the change was made to the file and is in the format of the job attribute DATFMT. The system cannot assure that the date stamp is always in ascending order for sequential journal entries because you can change the value of the system date.
Time stamp (*JOTIME*)	Zoned decimal (6,0)	Corresponds to the system time when the change was made to the file in the format hhmmss. The system cannot assure that the time stamp is always in ascending order for sequential journal entries because you can change the value of the system time.
Job name (*JOJOB*)	Character (10)	Specifies the name of the job that created the entry.
User name (*JOUSER*)	Character (10)	Specifies the user profile name of the user that started the job.
Job number (*JONBR*)	Zoned decimal (6,0)	Specifies the job number of the user that started the job.
Program name (*JOPGM*)	Character (10)	Specifies the name of the program that created the entry. If an application or CL program did not create the entry, the field contains the name of a system-supplied program such as QCMD or QPGMMENU.
Object name (*JOOBJ*)	Character (10)	Specifies the name of the object for which the journal entry was created. It is the name that was last known when the object was journaled. This is blank for some entries.
Library name (*JOLIB*)	Character (10)	Specifies the name of the library containing the object.

Table 12-2 (Page 2 of 2). Field Descriptions of the Fixed Length Portion of a Journal Entry *TYPE1

Field	Format	Description
Member name (*JOMBR*)	Character (10)	Specifies the name of the physical file member or is blank if the object is not a physical file.
Count/relative record number (*JOCTRR*)	Zoned decimal (10,0)	Contains a value specified on a parameter of the command for which the journal entry was created or the relative record number of the record in the physical file member (see Table 12-5 on page 12-18 for more information).
Indicator flag (*JOFLAG*)	Character (1)	Contains an indicator for the operation (see Table 12-5 on page 12-18 for more information).
Commit cycle identifier (*JOCCID*)	Zoned decimal (10,0)	Contains a number that identifies the commit cycle. A commit cycle is from one commit or rollback operation to another.
Reserved field (*JORES*)	Character (8)	Always contains zeros. Contains hexadecimal zeros in the output file.

If OUTFILFMT(*TYPE2) is requested on the DSPJRN command, or ENTFMT(*TYPE2) is requested on the RCVJRNE or RTVJRNE command, then the prefix portion of each converted journal entry fixed length is the same as the format in Table 12-2 on page 12-14, except for the information that follows the commit cycle identifier field. The fields of the prefix that follow the commit cycle identifier are shown in Table 12-3. The field names shown in parantheses in the table are the names of the fields in the system-supplied output file QSYS/QADSPJR2.

Table 12-3. Field Descriptions of the Fixed Length Portion of a Journal Entry *TYPE2

Field	Format	Description
User profile (*JOUSPF*)	Character (10)	Specifies the name of the user profile under which the job was running when the entry was created.
System name (*JOSYNM*)	Character (8)	Specifies the name of the system on which the outfile was created.
Reserved field (*JORES*)	Character (20)	Always contains zeros. Contains hexadecimal zeros in the output file.

For V2R1.1

A third value, *TYPE3 is supported on the OUTFILFMT parameter for the DSPJRN command, and the ENTFMT parameter for the RCVJRNE and RTVJRNE commands. If OUTFILFMT(*TYPE3) is specified on the DSPJRN command, or ENTFMT(*TYPE3) is specified on the RCVJRNE or RTVJRNE command, the information in the prefix portion of a converted journal entry is shown in Table 12-4. The field names shown in parantheses in the table are the names of the fields in the system-supplied output file QSYS/QADSPJR3.

*Table 12-4 (Page 1 of 2). Field Descriptions of the Fixed Length Portion of a Journal Entry *TYPE3*

Field	Format	Description
Entry length (*JOENTL*)	Zoned decimal (5,0)	Specifies the length of the journal entry including the entry length field, all subsequent positions of the journal entry, and any portion of the journal entry that was truncated if the length of the output record is less than the length of the record created for the journal entry. This field only appears when the journal entries are written to the database output file.
Sequence number (*JOSEQN*)	Zoned decimal (10,0)	Assigned by journal management to each journal entry. It is initially set to 1 for each new or restored journal and is incremented by 1 until you request that it be reset when you attach a new receiver. There are occasional gaps in the sequence numbers because the system uses internal journal entries for control purposes.
Journal code (*JOCODE*)	Character (1)	Identifies the primary category of the journal entry: **A** = Job accounting entry **C** = Information about the commitment control environment **F** = File level information **J** = Information about journal receivers and operation of the journal **P** = Performance entry **R** = Information about a change to a specific record **S** = SNA distribution services (SNADS) entry **T** = Security-related events **U** = User-created entry (added to the journal by the SNDJRNE command)
Entry type (*JOENTT*)	Character (2)	Further identifies the type of user-created or system-created entry. See the following tables for more information.
Time stamp (*JOTMST*)	Character (26)	Corresponds to the system date and time when the journal entry was logged in the journal receiver. The time stamp is in SAA format. The system cannot assure that the time stamp is always in ascending order for sequential journal entries because you can change the value of the system time.
Job name (*JOJOB*)	Character (10)	Specifies the name of the job that created the entry.
User name (*JOUSER*)	Character (10)	Specifies the user profile name of the user that started the job.
Job number (*JONBR*)	Zoned decimal (6,0)	Specifies the job number of the user that started the job.
User profile (*JOUSPF*)	Character (10)	Specifies the name of the user profile under which the job was running when the entry was created.

*Table 12-4 (Page 2 of 2). Field Descriptions of the Fixed Length Portion of a Journal Entry *TYPE3*

Field	Format	Description
Program name (*JOPGM*)	Character (10)	Specifies the name of the program that created the entry. If an application or CL program did not create the entry, the field contains the name of a system-supplied program such as QCMD or QPGMMENU.
Object name (*JOOBJ*)	Character (10)	Specifies the name of the object for which the journal entry was created. It is the name that was last known when the object was journaled. This is blank for some entries.
Library name (*JOLIB*)	Character (10)	Specifies the name of the library containing the object.
Member name (*JOMBR*)	Character (10)	Specifies the name of the physical file member or is blank if the object is not a physical file.
Count/relative record number (*JOCTRR*)	Zoned decimal (10,0)	Contains a value specified on a parameter of the command for which the journal entry was created or the relative record number of the record in the physical file member (see Table 12-5 on page 12-18 for more information).
Indicator flag (*JOFLAG*)	Character (1)	Contains an indicator for the operation (see Table 12-5 on page 12-18 for more information).
Commit cycle identifier (*JOCCID*)	Zoned decimal (10,0)	Contains a number that identifies the commit cycle. A commit cycle is from one commit or rollback operation to another.
System name (*JOSYNM*)	Character (8)	Specifies the name of the system on which the outfile was created.
Reserved field (*JORES*)	Character (20)	Always contains zeros. Contains hexadecimal zeros in the output file.

System-Created Journal Entries

This section describes the system-created journal entries by journal code.

Journal Code A: Journal entries with a journal code of A contain information about job accounting. Refer to the manual *Programming: Work Management Guide*, SC41-8078, for a detailed description of the contents of converted journal entries with journal code A.

Journal Code C: Journal entries with a journal code of C contain information about the commitment control information. Additional information about journal entries with journal code C follows in "Entry Types by Journal Code" on page 12-18.

Journal Code F: Journal entries with a journal code of F contain file level information about changes for a physical file member that are being journaled to this journal. (If you use a logical file in a program, the file level information reflects the physical file on which the logical file is based.) Journal entries with journal code F can also contain file level information for access paths associated with physical or logical file members that are being journaled to this journal. Additional information about journal entries with journal code F follows in "Entry Types by Journal Code" on page 12-18.

Journal Code J: Journal entries with a journal code of J contain information about the journal and the journal receivers. Additional information about journal entries with journal code J follows in "Entry Types by Journal Code" on page 12-18.

Journal Code P: Journal entries with a journal code of P contain information about performance. For the description of the layout of these entries, refer to the manual *Programming: Work Management Guide*, SC41-8078.

Journal Code R: Journal entries with a journal code of R contain information about a change to a specific record in the physical file member that is being journaled to the journal. For a given physical file memeber, the record-level journal entries appear in the journal in the order that the changes were made to the file. Additional information about journal entries with journal code R follows in "Entry Types by Journal Code" on page 12-18.

Journal Code S: Journal entries with a journal code of S contain information about SNA distribution services (SNADS). For the description of the layout of these entries, refer to the manual *Communications: Distribution Services Network Guide*, SC41-9588.

Journal Code T: Journal entries with a journal code of T contain auditing information. For the description of the layout of audit journal entries, see the manual *Security Concepts and Planning*, SC41-8083.

Entry Types by Journal Code: Table 12-5 describes, by journal code, the contents of the *Entry Type* field in the journal entries.

Table 12-5 (Page 1 of 3). Journal Entries by Journal Code and Type

Journal Code	Entry Type	Operation Description	See Note
A	DP	Direct print information.	1
A	JB	Job resource information.	1
A	SP	Spooled print information.	1
C	BC	Start commitment control (STRCMTCTL).	
C	CM	Set of record changes committed (COMMIT).	2
C	EC	End commitment control (ENDCMTCTL).	
C	RB	Set of record changes rolled back (ROLLBACK).	3
C	SC	Commit cycle started.	
F	AY	Journaled changes applied to a physical file member (APYJRNCHG).	4
F	CE	Change end of data for physical file member.	5
F	CL	Physical file member closed (for shared files, a close entry is made for the last close operation of the file).	6
F	CR	Physical file member cleared (CLRPFM).	
F	EJ	Journaling for a physical file member ended (ENDJRNPF).	
F	EP	Journaling access path for a database file member ended (ENDJRNAP).	
F	FD	Physical file member forced (written) to auxiliary storage.	7
F	IU	Physical file member in use at the time of abnormal system end.	8,9
F	IZ	Physical file member initialized (INZPFM).	10
F	JM	Journaling for a physical file member started (STRJRNPF).	11
F	JP	Journaling access path for a database file member started (STRJRNAP).	
F	MD	Physical file member deleted. This entry is created when you remove the member (RMVM) or delete the file (DLTF) containing the member.	
F	MF	Physical file member saved with storage freed (SAVOBJ, SAVCHGOBJ, or SAVLIB).	
F	MM	Physical file containing the member moved to a different library (MOVOBJ or RNMOBJ OBJTYPE(*LIB).	12
F	MN	Physical file containing the member renamed (RNMM or RNMOBJ).	12
F	MR	Physical file member restored (RSTOBJ or RSTLIB).	13
F	MS	Physical file member saved (SAVOBJ, SAVLIB, or SAVCHGOBJ).	13
F	OP	Physical file member opened (for shared files, an open entry is added for the first open of the file).	6
F	PD	Database file member's access path deleted (this entry is created when you remove the member (RMVM) or delete the file (DLTF) containing the member).	
F	PM	The logical owner of a journaled access path was moved (MOVOBJ or RNMOBJ OBJTYPE(*LIB)).	12

Table 12-5 (Page 2 of 3). Journal Entries by Journal Code and Type

Journal Code	Entry Type	Operation Description	See Note
F	PN	The logical owner of a journaled access path was renamed (RNMOBJ or RNMM).	12
F	RC	Journaled changes removed from a physical file member (RMVJRNCHG).	4
F	RG	Physical file member reorganized (RGZPFM).	14
F	SA	The point at which the APYJRNCHG command started running.	
F	SR	The point at which the RMVJRNCHG command started running.	
J	IA	System IPL after abnormal end.	8
J	IN	System IPL after normal end.	8
J	NR	Identifier for the next journal receiver (the receiver that was attached when the indicated receiver was detached).	15
J	PR	Identifier for the previous journal receiver (the receiver that was detached when the indicated receiver was attached).	15
J	RD	Delete of a journal receiver (DLTJRNRCV).	
J	RF	Storage for a journal receiver freed (SAVOBJ, SAVCHGOBJ, or SAVLIB).	
J	RR	Restore of a journal receiver (RSTOBJ or RSTLIB).	13
J	RS	Save of a journal receiver (SAVOBJ, SAVCHGOBJ, or SAVLIB).	13
P	TP	Performance shared pool change.	1
R	BR	Before-image of record updated for rollback operation.	16,17
R	DL	Record deleted in the physical file member.	16,17
R	DR	Record deleted for rollback operation.	16,17
R	PT	Record added to a physical file member.	17
R	PX	Record added directly by RRN to a physical file member.	17
R	UB	Before-image of a record that is updated in the physical file member (this entry is present only if IMAGES(*BOTH) is specified on the JRNPF command).	16,17
R	UP	After-image of a record that is updated in the physical file member.	17
R	UR	After-image of record updated for rollback operation.	17
S	CF	A change was made to the next system table through the CFGDSTSRV command.	18
S	ER	An error was detected by a SNADS process (this includes, but is not limited to, the sender, receiver, and router).	18
S	LG	An operation, such as sending or receiving a distribution queue entry, was successfully performed.	18
S	RT	A change was made to the routing table through the CFGDSTSRV command.	18
S	XE	DSNX error entry.	18
S	XL	DSNX logging entry.	18
T	AF	All authority failures.	19
T	PW	Passwords used that are not valid.	19
T	RP	Restore of programs that adopt their owner's authority.	19
T	RJ	Restore of job descriptions that contain user profile names.	19
T	RO	Restore of objects when ownership information changes.	19

Table 12-5 (Page 3 of 3). Journal Entries by Journal Code and Type

Journal Code	Entry Type	Operation Description	See Note
T	RA	Restore of objects when authority changes.	19
T	RU	Restore of authority for user profiles.	19
T	CA	Changes to object authority (authorization list or object).	19
T	CP	Create, change, delete, display, restore of user profiles.	19
T	DS	DST security officer password reset.	19
T	OW	Changes to object ownership.	19
T	PA	Changes to programs (CHGPGM) that will now adopt the owner's authority.	19
T	PS	Profile swap.	19
T	SV	Changes to system values.	19
T	NA	Changes to network attributes.	19
T	SE	Changes to subsystem routing.	19
T	DO	All delete operations on the system.	19
T	JD	Changes to the USER parameter of a job description.	19
U	User-created	User specified	20

Notes:

1. Refer to the *Programming: Work Management Guide*, SC41-8078, for the layout of the converted journal entry.
2. The *Count* field contains the length of the commit ID, and the *Entry-specific data* field contains the commit ID specified on the operation (character data).
3. The *Job name* and *Program name* fields do not appear if the entry was added during an IPL. The *Flag* field indicates whether the rollback operation was successful (hex F0 indicates that all changes were rolled back; hex F1 indicates that not all changes were successfully rolled back).
4. The *Count* field contains the number of journal entries applied or removed. The *Flag* field indicates the completion status:

   Hex F0 = command completed normally
   Hex F1 = command completed abnormally

   The *Entry-specific data* contains the following:

   Sequence number of the first entry applied or removed, zoned decimal (10, 0)
   Sequence number of the last entry applied or removed, zoned decimal (10, 0)

   ```
   Starting receiver name   Character (10)
   Library name             Character (10)
   Ending receiver name     Character (10)
   Library name             Character (10)
   ```

5. The *Count* field contains the relative record number of the last record retained in the physical file member.

6. The *Entry-specific data* field contains the following information on the file opened or closed. (This is a different name than the *Object Name* if a logical file was opened.)

```
File Name      Character (10)
Library name   Character (10)
Member name    Character (10)
Open options   Character (4) (not applicable to control language (CL)
                              entry type)
               Byte   Contents
               1      I = File opened for input
                      Blank = Input not specified
               2      O = File opened for output
                      Blank = Output not specifed
               3      U = File opened for update
                      Blank = Update not specified
               4      D = File opened for delete
                      Blank = Delete not specified
```

7. If the FD entry occurs at IPL, the *Job name*, *Job number*, and *Program name* fields are blank.

8. The time stamp created at IPL is read from the battery-powered clock. If the battery-powered clock cannot be read, the time is that of the system power down, not the time of the IPL because the system time has not been updated yet at the time of the journal entry being logged.

9. The *Flag* field indicates whether the file was synchronized with the journal:

   Hex F0 = file was synchronized with journal
   Hex F1 = file was not synchronized with journal

10. The *Flag* field indicates the type of record:

    Hex F0 = *DFT (default)
    Hex F1 = *DLT (delete)

    The *Count* field contains the number of records specified on the TOTRCDS parameter of the Initialize Physical File member (INZPFM) command.

    For V2R1.1

    If the member is initialized with default records, the *Entry-specific data* in the entry contains the default record image.

11. The *Flag* field indicates the type of images:

    Hex F0 = after-images
    Hex F1 = before- and after-images

    The *Entry-specific data* field contains an indicator of the value specified on the OMTJRNE parameter of the STRJRNPF command:

    Hex F0 = no entries are omitted from being journaled
    Hex F1 = open and close entries are omitted

12. The *Entry-specific data* field contains the name of the member before a move or a rename operation in the following form:

```
File Name      Character (10)
Library name   Character (10)
Member name    Character (10)
```

13. The *Entry-specific data* field contains the following save or restore information:

Media type (DKT, TAP or SAV)	Character (3)
First Volume ID	Character (6)
Date of save or restore operation	Character (6)
Time of save or restore operation	Zoned decimal (6, 0)

14. The *Entry-specific data* field contains the name of the file used in the Reorganize Physical File Member (RGZPFM) command. This field is blank if you specified KEYFILE(*NONE). If you specified other than KEYFILE(*NONE), for example, if you specified KEYFILE(library-name/filename member-name), this field contains the name of the logical file.

File Name	Character (10)
Library name	Character (10)
Member name	Character (10)

15. The *Count* field contains the number of receivers attached or detached. The *Entry-specific data* field contains the receiver name (10 characters) and the library name (10 characters) for each receiver included in the number returned in the *Count* field.

16. The *Flag* field indicates whether a record image is present in the journal entry:

    Hex F0 = before-image is not present
    Hex F1 = before-image is present

    If before-images are being journaled, and the *Flag* field indicates that a before-image is not present, this implies that an update or a delete operation is being requested for a record that has already been deleted.

17. The *Entry-specific data* field contains the image of the physical record. For entry types PT, PX, UP or UR, this field contains the after-image of the record. For entry types UB, DL, BR or DR, this field contains the before-image of the record if IMAGE(*BOTH) was specified and if the record was not previously deleted.

18. Refer to the *Communications and Systems Management Guide (Alerts and Distributed Systems Node Executive)*, SC41-9661, for the layout of the converted journal entry.

19. Refer to the *Security Concepts and Planning*, SC41-8083, for the layout of the converted journal entry.

20. The *Entry-specific data* is the value specified on the ENTDTA parameter of the SNDJRNE command.

Variable Length Portion of a Journal Entry

If you use DSPJRN with *TYPE1 or *TYPE2 specified for the OUTFILFMT parameter, or if you use RCVJRNE or RTVJRNE with *TYPE1 or *TYPE2 specified for the ENTFMT parameter, the variable length portion of the journal entry includes just the entry-specific data. The contents of the entry-specific data depends on the journal entry code and entry type as was explained in the notes to Table 12-5 on page 12-18.

For V2R1.1

If OUTFILFMT(*TYPE3) is requested on a DSPJRN command, the variable length portion of the converted journal entry will have two fields.

The first field is the *Null Value Indicators* and the second field is the *Entry-specific data*. The *Null Value Indicators* contains relevant information only for entries with journal code R. The *Null Value Indicators* is a character string with one character for each field in the physical file whose record image appears in the journal. Each character has the following interpretation:

Hex F0 = corresponding field in the physical file is not NULL.
Hex F1 = corresponding field in the physical file is NULL.

The second field in the variable length portion of the journal entry is the *Entry-specific data* in the journal entry.

The *Null Value Indicators* and *Entry-specifc data* fields have been defined as variable length character fields in the system-supplied output file QSYS/QADSPJR3. See the DSPJRN command in the *Programming: Control Language Reference*, SC41-0030, for additional details regarding OUTFILFMT(*TYPE3).

If ENTFMT(*TYPE3) is requested on a RCVJRNE or a RTVJRNE command, the variable length portion of the converted journal entry contains three pieces of information. The first field is a *DEC LEN(5,0) field which has the length the user requested be used for the *Null Field Indicator* field. The second field is the *Null Field Indicator* and the third field is the *Entry-specific data*.

User-Created Journal Entries

Use the Send Journal Entry (SNDJRNE) command to add your own entries to a journal. The system places these entries on the journal's attached journal receiver along with the system-created journal entries. To help identify your entries, you can associate each entry with a particular physical file member.

You may add entries to the journal to identify significant events (such as a checkpoint) or to help in the recovery of your applications. The data specified on the ENTDTA parameter becomes the *Entry-Specific Data* field in the journal entry, and the TYPE parameter value becomes the *entry type*.

Retrieving a Journal Entry

Use the Retrieve Journal Entry (RTVJRNE) command in a CL program to retrieve a journal entry and place it in variables in the program. You can retrieve the following:

- Sequence number
- Journal code
- Entry type
- Journal receiver name
- Library name for the journal receiver
- Journal entry

For example, you can use this command to automate your recovery procedures or to change the journal receivers and then save them. The manual *Programming: Control Language Reference*, SC41-0030, has a description of the format of the journal entry. For an example of using the RTVJRNE command in a program, see the topic "Using the Retrieve Journal Entry (RTVJRNE) Command in a Program" on page B-5.

Using Journaling to Provide an Audit Trail

You can use journal management to provide an audit trail of changes made to your database files. You can determine which program or user made changes to files by using the journal entries, and you can determine what changes were made to specific records by using the CMPJRNIMG command.

Journal management can be used to provide an audit trail because:

- The journal entries cannot be removed or changed, even by the security officer.
- The journal entries represent a chronological sequence of events.
- Each journal entry in the system is sequentially numbered without gaps until the sequence number is reset by the CHGJRN command. If the sequence number is reset, a journal entry is written. When you display the journal entries, there can be gaps in the sequence numbers because some journal entries are only used internally by the system and some entries are combined when they are displayed.
- The journal contains entries indicating when each journal receiver was changed and the name of the next journal receiver in the chain.
- Entries are written whenever journaling for a file is ended or whenever a file is restored.

Remember that the date and time recorded in the journal entries depends on the date and time entered during an IPL and, therefore, may not represent the actual date and time. Also, if you use shared files, the program name that appears in the journal entry is the name of the program that first opened the shared file.

Comparing Journal Images

Use the Compare Journal Images (CMPJRNIMG) command to compare and list the differences between the before-image of a record and the after-image of that record, or the after-image of a record with the previous after-image of that record.

The printed output from the CMPJRNIMG command shows the before- and after-images of a record followed by a line that indicates (with asterisks) the specific change in the record on a character-by-character basis. If you compare the after-images, the output shows the previous after-image of the record and the current after-image of the record, followed by a line indicating the changes.

Performance and Space Considerations

The following considerations apply to system performance and space:

- Space requirements increase if both before- and after-images are journaled, but performance is minimally affected.
- Space requirements increase if you journal access paths. The actual increase is application-dependent. The minimum increase occurs when you make all changes to the same primary value of the access path (for example, when the access path uses the date added field as its primary value and you add many records on the same day). Journaling access paths has a minimal effect on performance. The system packages before- and after-record images and any access path changes into a single write operation to disk.
- To save space, you can use the OMTJRNE parameter on the Start Journal Physical File (STRJRNPF) command to remove the open and close entries that are written to the journal when a file is being opened or closed. Depending on the application, this can be a space and a performance advantage. When you remove the open and close entries, you cannot:
  - Use the TOJOBO and TOJOBC parameters on the APYJRNCHG and RMVJRNCHG commands.
  - Use the journal to determine which users have opened the file but have not made changes.

  The manual *Programming: Control Language Reference*, SC41-0030, has more information about these commands.
- Space requirements also increase when the number of files and/or the number of file access paths being journaled increases. As the number of files being journaled increases, the general performance of the system, including the IPL, is slower. (This depends to some extent on the number of changes being made to the journaled files.) The effect on performance of the IPL occurs primarily after an abnormal system end and depends on the number of journal entries, the number of files being journaled, and the

number of access paths being journaled when the abnormal system end occurred.

- Space requirements increase when you use two receivers. The second receiver does not require the same overhead as the first receiver because the write operations are done at the same time. If the receivers are in the same ASP, there could be a decrease in performance if the receivers are on the same storage unit.

- Using the force-write ratio (FRCRATIO) parameter on physical files to force data to be physically written to the disk decreases performance. You can increase or delete the force-write ratio on physical files being journaled (and for the logical files over the physical files) because the change within the journal is written to auxiliary storage before it is written to the database file.

  To minimize the time required during an abnormal IPL to synchronize the database files being journaled, specify a reasonable force-write ratio on the physical files. An IPL after an abnormal end takes longer depending on the number of unforced changes needed to synchronize the journaled files. For a typical environment in which a file is updated in either interactive or batch or both, a force ratio value of approximately 500 can be used on the physical file being journaled. This prevents a large number of unforced changes from accumulating in the journal and, at the same time, minimizes the number of times that a force would disrupt production jobs.

  A force-write ratio of 1 writes every add, update, and delete operation to a file member immediately to disk. You specify the force-write ratio on the FRCRATIO parameter on the Create Physical File (CRTPF), Change Physical File (CHGPF), Create Logical File (CRTLF), Change Logical File (CHGLF), Create Source Physical File (CRTSRCPF), Change Source Physical file (CHGSRCPF) or Override with Database File (OVRDBF) command. If you are journaling a file, typical application use would ignore the FRCRATIO parameter.

- If sequential-only processing (SEQONLY(*YES)) is specified for files being journaled, records added or inserted in the file are blocked. When the block is full, the records are written to the journal with one write operation and to the database file. The records appear in the journal as a connected series of entries.

  You can decrease the number of records in a block or delete blocking altogether for recovery reasons using the Override with Database File (OVRDBF) command. However, this may make the application program run slower. Sequential-only processing is specified by the OVRDBF command, or, in some cases, the high-level language implicitly specifies sequential-only processing. For a description of when high-level languages specify sequential-only, see the appropriate high-level language manual.

To determine the approximate size of a journal receiver, estimate that the system-created portion of a journal entry is 78 bytes. If the record size in your files is, for example, 115 bytes, the journal entry would be approximately 193 bytes. (These values are approximate because the system-created portion of the journal entry varies according to the journal entry type. The number of bytes used for the storage of a journal entry in a receiver is not the same as the number of characters required when the journal entry is shown.) Use the following formula to estimate the size of a receiver:

```
Journal receiver size per day =
         (78 bytes + average record size)
         x (number of changes per day)
```

If you specify only after-images and if you have 10 000 changes per day and an average record length of 115, the journal receiver could reach approximately 1 930 000 bytes. (This size does not include any user-created entries.) Remember that any formula to determine the size of a receiver provides only an estimate of the size. You can determine the actual size of a receiver by using the Display Object Description (DSPOBJD) or the Display Journal Receiver Attributes (DSPJRNRCVA) command. As with other objects, the system allocates space as needed in increasingly larger allocations. If the size of a journal receiver is shown immediately after creation, it shows the initial allocation, even though no entries exist yet.

If you use before- and after-images, the journal receiver size does not necessarily double because not all journal entries contain a record image and not all records have both before- and after-images. For example, if all operations to the journaled files are write operations, the journal receiver size does not double (no before-images exist). If the operations include delete operations, the size increases but does not double. If the operations are all update operations, the receiver could double in size.

Journaling access paths requires more auxiliary storage than journaling physical files with after-images only. Several journal entries that cannot be displayed are also written in the journal when journaling access paths. The amount of additional storage needed depends on how many access paths are being journaled, and how the journaled access paths are updated by various applications. Applications that update, delete, or add keys in an ascending or descending order use less auxiliary storage than applications that update, delete, or add keys in random order.

Access Path Recovery

This section describes how to recover access paths more quickly after the system ends abnormally using access path journaling. If you use access path journaling, you do not need to completely rebuild access paths after an abnormal system end.

Unless you choose to journal your access paths, the system can spend a significant amount of time rebuilding access paths during the IPL following an abnormal system end.

Journaling Access Paths

You can use access path journaling to recover access paths during an IPL, or you can delay the recovery to be performed after the IPL. Recovery for the immediate maintenance files (files created with the MAINT(*IMMED) parameter value) takes place during the IPL. Recovery for delayed maintenance files (files created with the MAINT(*DLY) parameter value) takes place during the next open operation that uses that access path.

Access paths for files that are specified with a parameter value of MAINT(*REBLD) cannot be journaled. Recovery for rebuilt maintenance files, created with the MAINT(*REBLD) parameter value, takes place during the next open operation that uses that access path.

Before you begin journaling access paths, consider the following:

- Before you can journal an access path, you must first journal the physical files over which that access path is built.
- All physical files over which the access path is built must be journaled to the same journal as the access path.
- Any physical file specified for access path journaling must have a keyed access path. Any logical files that use the physical file as a dependent file are not implicitly journaled. You must specify each access path that is to be journaled by naming a specific file.
- All physical files must remain journaled while the access path is being journaled.

Considerations for Storage and Performance: Space requirements can increase significantly if you journal access paths. The minimum requirement occurs when you make all changes to the same point of the access path (for example, when the access path uses the date-added field as its primary value and you add many records on the same day). When anticipating the effect of access path journaling on system space and performance, consider the following:

- Because access paths can be shared, you can journal the same access path from several files. For example, file B shares the access path of file A. Both files can request journaling of their access paths and there is no additional overhead. If file A ends journaling its access path, the access path remains journaled until file B ends journaling the access path. Journaling can end only through files that have access paths journaled.
- Access path journaling does not cause additional synchronous write operations to the disk beyond those required to journal the underlying physical file. Access path information is written at the same time as the after-image of the underlying changed database record. The affect on system performance due to access path journaling is kept to a minimum.
- Access path journaling can significantly increase the amount of auxiliary storage

used. For more information, see "Performance and Space Considerations" on page 12-24.

- To improve the performance of journaling, you should isolate the journal receiver to a user ASP. The amount of additional auxiliary storage required varies, based on the access path characteristics and the types of changes that occur. However, access path journaling normally requires an increase in the amount of auxiliary storage required by the journal receivers. For more information on user ASPs, refer to "General Information about Auxiliary Storage Pools" on page 14-7.

  Several journal entries that cannot be displayed are also created in the journal when journaling access paths. The amount of additional storage needed depends on how many access paths are being journaled, and how the journaled access paths are updated by various applications. Applications that update, delete, or add keys in an ascending or descending order use less auxiliary storage than applications that update, delete, or add keys in random order.

  Note: If you do not put journal receivers in a separate user ASP, consider saving them regularly on tape or diskette to protect them from loss due to disk failure.

- The system journals information in addition to the before-image and after-image of the database records. The system uses this additional information only after an abnormal system end to recover the access paths.

Space requirements also increase when the number of files and/or the number of file access paths being journaled increases.

Considerations for Journaling Database Files: Consider the following when using access path journaling for database files:

- Access paths for both physical and logical files can be journaled. Designate the physical and logical files that will have their access paths journaled using the Start Journal Access Path (STRJRNAP) command. When you specify journaling for database file access paths, all access paths for all members are journaled and new members added to the file automatically have their access paths journaled.
- If a database file is saved while access paths are journaled, and the file is deleted and restored, the restore operation attempts to begin journaling the access paths for the file.

Considerations for Using the Create and Change Commands with Access Path Journaling: Consider the following command requirements when using access path journaling:

1. All journaled access paths must have a maintenance attribute of *IMMED or *DLY specified on the MAINT parameter on the Create Physical File (CRTPF), Create Source Physical File (CRTSRCPF), and Create Logical File (CRTLF) commands as well as the Change Physical File (CHGPF), Change Logical File (CHGLF), or Change Source Physical File (CHGSRCPF) command. MAINT(*REBLD) cannot be specified.
2. All access paths that are to be journaled must have a force attribute of *NO (the FRCACCPTH parameter on the CRTPF, CRTLF, CRTSRCPF, CHGPF, CHGLF, or CHGSRCPF command).

Access Path Journal Entries

Several entry types are used to describe journal activity for journaled access paths. These entries are:

JP Start journaling an access path. For join logical files, more than one JP entry can be created for the same logical file member: one for the primary access path and one for each of the secondary access paths. If a file member's access path is already being journaled when access path journaling is started for the file, no JP entry is created for that access path. This condition can arise if the access path is being shared by several file members, and one of the sharing files has already journaled the access path.

EP End journaling access path for member. For join logical files, more than one EP entry can be created for the same logical file member: one for the primary access path and one for each of the secondary access paths. If a file member's access path is being shared and one of the other sharing files is still journaling the access path, journaling is not ended for that access path and no EP entry is written.

PD Access path deleted. A journaled access path was deleted by deleting a file journaling its access paths or by removing a member from a file journaling its access paths.

PM The logical owning member of a journaled access path was moved to a different library.

PN The logical owning member of a journaled access path was renamed.

Other system-created entries are recorded in the journal for journaled access paths. You cannot display or use them in an apply or remove journal changes operation.

The system makes entries to the journal to record all necessary information about the changes made to the access paths. For example, you have the option to journal information about the access paths changed because of a change in the physical file member. These entries are made only if one or more access paths, built over the physical file member, are changed and the access paths are changed. The system uses these entries only when recovering the access paths after the system abnormally stops. These entries cannot be shown, or used to apply or remove journal changes.

For a complete description of the system-created journal entries for a file member, see "System-Created Journal Entries" on page 12-17.

Access Path Recovery Actions

No explicit recovery actions initiated by the user are required when journaling access paths. After an abnormal system end, the system automatically uses the journal to recover journaled access paths, if necessary.

Commitment Control

Commitment control is an extension of the journal function on the AS/400 system. It allows you to define and process a number of changes to database files as a single transaction.

For commitment control, a **transaction** is defined as a group of changes that appear as a single change to the work station user. An example of changes that can be grouped together is the transfer of funds from a savings to a checking account. To the work station user, this is a single transaction. However, more than one transaction occurs because both savings and checking accounts are updated. User transactions can be:

- Inquiries in which no file changes occur
- Simple transactions in which one file is changed each time the Enter key is pressed
- Complex transactions in which one or more files are changed each time the Enter key is pressed
- Complex transactions in which each time the Enter key is pressed one or more files are changed, but these changes represent only a part of a logical group of transactions

Using commitment control allows you to:

- Ensure that all changes within a transaction are completed for all files affected.
- Ensure that all changes within a transaction are rolled back if processing is interrupted.
- Remove changes made during a transaction when the work station user determines that a transaction is in error.
- Design an application so that it can be started again if a job or system fails.

Two operations affect changes made under commitment control:

- Commit

  All changes made to a database file since the previous commit or rollback operation are applied permanently to the file and all records are unlocked.

- Rollback

  All changes made since the previous commit or rollback operation are removed from the file and all records are unlocked.

The commit and rollback operations outlined are available in several AS/400 programming languages including RPG III, COBOL/400* control language (CL), and Structured Query Language/400 (SQL/400*).

Starting Commitment Control

To use commitment control, all database files that need to be under commitment control in a routing step must be journaled to the same journal. Both before-images and after-images are journaled for files under commitment control. If you specified only after-images when you started journaling your files, the system automatically journals before-images while those files are under commitment control. You start commitment control by using the Start Commitment Control (STRCMTCTL) command. On the STRCMTCTL command, specify the following:

- The LCKLVL for the records in database files under commitment control. For more information about lock levels, see the topic "Lock Level Parameter."
- The notify object (NFYOBJ) to which an entry (the commit ID) is sent to identify the last transaction successfully committed by the job. The system places the commit ID in the notify object if an abnormal system or job end occurs or if uncommitted entries exist during the normal routing step end. For more information about notify objects, see the topic "Notify Object Parameter" on page 12-31.

When commitment control is started, the system creates a commitment definition to save internal control information. This information includes:

- The parameters given on the Start Commitment Control (STRCMTCTL) command
- The current status of the commitment control environment
- Information about the files that contain changes made during the current transaction

The commitment definition is maintained by the system until commitment control is ended. When commitment control is ended, the commitment definition is destroyed.

You cannot start commitment control if it is already started in the routing step. When you end the commitment control environment, you can start commitment control again. For the new commitment control environment, you can place different files under commitment control. You can journal these files to a journal other than the one used in the preceding commitment control environment.

The database files placed under commitment control are specified at the time the files are opened by a high-level language program. For details on how to place a file under commitment control, refer to the appropriate language reference manual.

If you specify that a shared file be placed under commitment control, all subsequent uses of that file in the same routing step must be placed under commitment control. If SEQONLY(*YES) is specified for a file opened for read only (either implicitly by the high-level language program or explicitly by the OVRDBF command), SEQONLY(*YES) is ignored and SEQONLY(*NO) is used.

Lock Level Parameter: Specify the lock level according to your needs, the wait periods allowed, and the release procedures used most often. The following section refers only to files opened under commitment control. Specify one of the following lock levels:

- *CHG

  Use the value *CHG if you want only changed records to be protected from changes by other jobs running at the same time.

- *CS

  > **For V2R1.1**
  >
  > The *CS lock level is used if you want both changed and retrieved records to be protected from changes by other jobs running at the same time. Retrieved records are protected only until they are released, or a different record is retrieved.

- *ALL

  Use the value *ALL if you want both changed records and retrieved records to be protected from changes by other jobs running at the same time. Records that are retrieved or changed are protected until the next commit or rollback operation.

Table 12-6 on page 12-30 shows the duration of record locks for files under and not under commitment control.

Table 12-6 (Page 1 of 2). Lock Duration by Lock Level Parameter

Request	LCKLVL Parameter	Duration of Lock	Lock Type
Read only	No commitment control	No lock	
	LCKLVL(*CHG)	No lock	
	LCKLVL(*CS)	From read to next read, commit, or rollback	*READ
	LCKLVL(*ALL)	From read to commit or rollback	*READ
Read for update then update or delete	No commitment control	From read to update or delete	*UPDATE
	LCKLVL(*CHG)	From read to update or delete From update or delete to next commit or rollback	*UPDATE *UPDATE
	LCKLVL(*CS)	From read to update or delete From update or delete to next commit or rollback	*UPDATE *UPDATE
	LCKLVL(*ALL)	From read to update or delete From read to update or delete to next commit or rollback	*UPDATE *UPDATE
Read for update then release	No commitment control	From read to release	*UPDATE
	LCKLVL(*CHG)	From read to release	*UPDATE
	LCKLVL(*CS)	From read to release From release to next read, commit, or rollback	*UPDATE *READ
	LCKLVL(*ALL)	From read to release From release to next commit or rollback	*UPDATE *UPDATE
Add	No commitment control	No lock	
	LCKLVL(*CHG)	From add to commit or rollback	*UPDATE
			*UPDATE
	LCKLVL(*CS)	From add to commit or rollback	*UPDATE
	LCKLVL(*ALL)	From add to commit or rollback	*UPDATE

Table 12-6 (Page 2 of 2). Lock Duration by Lock Level Parameter			
Request	**LCKLVL Parameter**	**Duration of Lock**	**Lock Type**
Write direct	No commitment control	For duration of write direct	*UPDATE
	LCKLVL(*CHG)	From write direct to commit or rollback	*UPDATE
	LCKLVL(*CS)	From write direct to commit or rollback	*UPDATE
	LCKLVL(*ALL)	From write direct to commit or rollback	*UPDATE
Note: LCKLVL(*CS) is new for V2R1.1			

The lock level specified for commitment control applies only to the routing step for which commitment control is active. A *READ type of record lock is obtained on records that are not read for update when the lock level is *ALL. This type of lock prevents other jobs from reading the records for update but does not prevent the records from being accessed from a read-only operation. A *UPDATE type of record lock is obtained on records that are updated, deleted, added, or read for update. This type of lock prevents other jobs from reading the records for update, and prevents jobs running under commitment control with a record lock level of *ALL from accessing the records for even a read-only operation.

The following conditions apply to the lock levels:

- The *CHG lock level ensures programs under commitment control are not able to read a record for update that is locked by another job. This is normal locking protocol with or without commitment control.

For V2R1.1

The *CS lock level ensures programs under commitment control are not able to read a record for update that is locked by another job. This is normal locking protocol with or without commitment control. In addition, they are not able to read records for update that have been accessed by another routing step until that routing step accesses a different record.

- The *ALL lock level ensures that programs under commitment control are not able to access a record locked for update by another routing step. This is different from normal locking protocol. When the lock level is specified as *ALL, even a record that is not read for update cannot be accessed if it is locked with a *UPDATE type of lock in another job.
- Programs that are not using commitment control can read records locked by another job, but cannot read records for update, regardless of the LCKLVL setting.

Notify Object Parameter

A **notify object** can optionally be specified when starting commitment control. If specified, the notify object is updated with the last successful transaction (commit ID) when:

- A job ends abnormally
- A job ends normally with uncommitted changes
- The system ends abnormally

This information can be used by the workstation user or user-programs to start the application again from the last successful commitment boundary.

You can specify the notify object as a database file, data area, or message queue on the STRCMTCTL command.

You can specify the commit ID that is sent to the notify object on the CL COMMIT command, the RPG COMMIT operation code, or on the PL/I PLICOMMIT subroutine. The commit ID or notify object function is not supported on the COBOL

COMMIT verb or the SQL COMMIT statement. The maximum size of the commit ID area within the notify object is 4 000 characters. If the commit ID is larger than 4 000 characters, the ID is truncated.

When a notify object is updated with the commit ID, it is updated as follows:

- Database file

  If a database file is used as the notify object, the commit ID is added to the end of the file. Any existing records are left in the file. Because several users or jobs can be changing records at the same time, each commit ID in the file should contain a unique identification to associate the data with the job that failed. The file serving as a notify object can be journaled.

- Data area

  If a data area is used as the notify object, the entire contents of the data area is replaced when the commit ID is placed in the data area. If more than one user or job is using the same program, only the commit ID from the last job that ends will be in the data area. Consequently, a single data area notify object does not produce the correct information for starting again in this situation. To solve this problem, use a separate data area for each workstation user or job.

- Message queue

  If a message queue is used as a notify object, the message CPI8399 is sent to the message queue. (The commit ID is placed in the Message Help.) Define the contents of the message to associate the information with the job that ended.

Completion: The notify object reacts to the following types of completion:

- If a routing step ends normally and no uncommitted entries exist, the system does not place the commit ID in the notify object. Instead, the ID appears in the journal.
- If uncommitted entries exist when a routing step ends, the commit ID is placed in the notify object and the uncommitted entries are rolled back. For more information on the functions done during the normal ending of a routing step, see the topic "Commitment Control during Routing Step End" on page 12-35.
- If the job ends abnormally before the first successful commit operation, the system does not write to the notify object. If you want to differentiate between this condition and a normal program completion, your program must write a specific entry into the notify object.
- If an abnormal system or job end occurs after a successful commit operation, the system places the commit ID in the notify object. You can then use the commit ID to start the program again.

Committing Database Changes

A commit operation (COMMIT) permanently writes changes made to a database since the last commit or rollback operation. Records are unlocked and available to other users. When a successful commit operation runs, the system does the following:

- Adds information to the commitment definition to show that a new transaction has been started.
- Saves the commit ID provided by the program for use at recovery time. The commit ID is not placed in the notify object unless a failure occurs or uncommitted entries exist when a routing step is normally ended.
- Places an entry on the journal to indicate that changes made under commitment control have been committed. Before the commit, the before-images and the after-images of the changed records were placed on the journal.
- Sets the positions of the files under commitment control.
- Does not change the input/output feedback area and buffers.
- Releases all locks on records made during the processing of the transaction.

All of the above must perform correctly for the commit operation to be successful.

If you attempt a commit operation when no files have been opened under commitment control, the commit operation is ignored. If you want a specific entry in your notify object at the beginning of your program before files are opened, you must explicitly write the entry to the notify object yourself.

Use the CL COMMIT command to perform the commit operation (in RPG, use the COMMIT operation code; in COBOL, use the COMMIT verb; in PL/I, use the PLICOMMIT subroutine; in SQL, use the COMMIT statement).

The following CL command commits changes to database files and specifies a commit ID if the job or system fails. The commit ID includes the user profile name, the program name, the order number, and the item number as follows:

COMMIT 'JONES ORDENT AA12345 L2635'

If records are being added to a file under commitment control and SEQONLY(*YES) is requested (a buffer of records), a commitment operation causes the buffer to be cleared.

Certain system functions automatically cause a transaction to end or start (commit boundary to be established) as shown in Table 12-7.

Table 12-7. Automatic System Functions

System Function	Commitment Boundary Established?	System Action	Comment
Abnormal job end	Yes	Rollback[2]	
Normal routing step end	Yes	Rollback[2]	Any uncommitted changes are rolled back.
End servicing of program start request by a prestart job	Yes	Rollback[2]	Acts like a normal routing step end.
Transfer job (TFRJOB or TFRBCHJOB)	Yes	Rollback[2]	Transfer job is treated as a job end.
Reroute job (RRTJOB)	Yes	Rollback[2]	Reroute job ends the current routing step.
System failure	Yes	Rollback[2]	
First open operation of any file under commitment control	Yes	No action	The initial transaction.
File close operation	No	Changes remain uncommitted	Locks are kept on any data records that were locked.
Start commitment control (STRCMTCTL)	No	Set up commitment control environment	The action is performed before any commit or rollback operation.
End commitment control (ENDCMTCTL)	Yes	Rollback	Any uncommitted changes are rolled back.[1]
Commit (COMMIT)	Yes	Commit	
Rollback (ROLLBACK)	Yes	Rollback	

Notes:

1 **For V2R1.1**, if there are any uncommitted changes for an interactive job, inquiry message CPI8350 is sent to ask the user whether the changes should be committed or rolled back before the commitment control environment is ended. For a batch job, the changes are rolled back.

2 The system performs an implicit rollback operation for these functions.

Rolling Back Database Changes

A rollback (ROLLBACK) operation removes changes made to files under commitment control since the last commit or rollback operation. Records are unlocked and available to other users. The commit ID currently saved by the system remains the same as the commit ID provided with the last commit operation.

A rollback operation has the following effect on the job and the system:

- If a record is deleted from a file, the record is added back in.
- Any changes to records since the last transaction are removed, and the original records (the before-images) are placed in the file.
- If any records were added to the file, they remain in the file as deleted records.
- The commitment boundary is established as the last successful commitment boundary (or the start of commitment control).
- Entries are placed on the journal indicating that a rollback operation occurred. The journal also contains changed images of the changes that were rolled back.
- The system positions the open files under commitment control at the last accessed record in the previous transaction or at the open position if no commit operation has been issued for the file. This is an important consideration if you are doing sequential processing.
- The I/O feedback area and I/O buffers remain unchanged.
- Closed files are not reopened or repositioned.
- Operations on files, other than positioning, are not rolled back. For example, opened files are not closed and cleared files are not restored.
- All database records locked under commitment control are unlocked and made available.

Use the CL ROLLBACK command to perform the rollback operation (in RPG, use the ROLBK operation code; in COBOL, use the ROLLBACK verb; in PL/I, use the PLIROLLBACK subroutine; and in SQL use the ROLLBACK statement).

Displaying the Status of Commitment Control

You can select an option on the Work with Jobs display to display the status of commitment control for a particular job. The following information on the Display Job Commitment Control Status display identifies information that can be helpful in debugging a program:

- Journal to which the files under commitment control are being journaled
- System-generated identifier of the current commit cycle
- Default lock-level specified on the STRCMTCTL command

  For V2R1.1

  Whether there are any resources under commitment control, and if so, whether they are local or remote.

- Name of the notify object
- Total number of commit operations run since the last STRCMTCTL command
- Total number of rollback operations run since the last STRCMTCTL command

From the Display Job Commitment Control Status display, you can select the option Display Job Commitment Control Files.

For V2R1.1

Display Job Commitment Files display is changed to Display Record Level Status.

This display lists the following:

- Files currently under commitment control
- Total number of committed changes to the files since the last time the file was opened
- Total number of changes to the files that were rolled back since the last time the file was opened
- Total number of changes to the files waiting for a commit or rollback operation
- Level of record locking for the file
- Status of the file (either open or closed)

For a description of the commitment control status displays, use the online help information.

Ending Commitment Control

End commitment control at any time in a job by using the End Commitment Control (ENDCMTCTL) command. All files opened under commitment control must be closed before the ENDCMTCTL command is run.

If there are uncommitted changes when the ENDCMTCTL command is run, escape message CPF8356 is sent, uncommitted changes are rolled back, and the commit ID is placed in the notify object for recovery use. There is probably an error in your application.

When the commitment control environment ends successfully, a commitment boundary is established. Once you reach this stage, no lingering transaction recovery is required during the next IPL. The system does not place the commit ID in the notify object, and does not perform any special functions during the next IPL because the changes were committed to the files.

After commitment control ends, you can establish another commitment control environment by issuing the STRCMTCTL command. The new commitment control environment is independent of the previous environment. Different files might be under commitment control in the new environment, and the files might be journaled to a different journal.

Although you can start and end commitment control more than once in a job, the amount of system resources required for this repeated start/end pair degrade performance.

Commitment Control during Routing Step End: The system performs a rollback operation on uncommitted changes when a routing step ends. When this happens, the following functions are also performed:

- If changes are rolled back at the end of a normal routing step, a message is placed on the job log.
- If a notify object is used, the system places the commit ID in the notify object for recovery use.

A routing step is normally ended by one of the following:

- A normal return.
- A normal sign-off.
- An End Job (ENDJOB) command specifying a controlled cancel (*CNTRLD). If a controlled end is issued, runs out of time, and becomes an immediate end, the following can occur:
  - Commitment control ends before the time-out occurs. This appears as a normal end, and no commit ID is placed in the notify object.
  - Commitment control has not ended when the time-out occurs. This appears as an abnormal end. If a notify object is used, the commit ID is placed in the notify object.
- An End Subsystem (ENDSBS) command specifying a controlled end (*CNTRLD).

Any other end of a routing step is abnormal.

Rerouting or Transferring Jobs: The Reroute Job (RRTJOB), Transfer Job (TFRJOB), and Transfer Batch Job (TFRBCHJOB) commands end the current routing step and start a new routing step. If commit control is active in a routing step and the job is rerouted or transferred, uncommitted changes are rolled back, commitment control is ended, and the commit ID is placed in the notify object.

Commitment Control during Abnormal System or Job End: The system performs the following after an abnormal system (during the next IPL) or job end:

- All transactions with uncommitted changes are rolled back, including those changes being committed at the time of the abnormal end.
- The commit ID that identifies the last successful commit boundary is placed in the notify object if a notify object is being used.
- All database changes committed before the abnormal system or job end are assured to be in the database.
- No uncommitted changes appear in the database.

Considerations for Using Commitment Control

The following sections describe programming considerations for using commitment control.

Determining the Size of a Transaction

For this discussion, a transaction is interactive. (Commitment control can also be used for batch applications, which often can be considered a series of transactions. Many of the same considerations apply to batch applications, which are discussed in "Starting Application Programs Using a Notify Object" on page 13-35.)

When choosing the lock level for your records, consider the size of your transactions. Size should determine how long records are locked before a transaction ends. You have to decide if a commit or rollback operation for commitment control is limited to a single use of the Enter key, or if the transaction consists of many uses of the Enter key.

The maximum number of records that can be locked in a transaction is 32 768. (A transaction is from one commit and rollback operation to another.)

For example, for an order entry application, a customer might order several items in a single order requiring an order detail record and an inventory master record update for every item in the order. If the transaction is defined as the entire order and each use of the Enter key orders an item, all records involved in the order are locked for the duration of the entire order. Therefore, often-used records (such as inventory master records) could be locked for long periods of time, preventing other work from progressing. If all items are entered with a single Enter key using a subfile, the duration of the locks for the entire order is minimized.

In general, the number and duration of locks should be minimized so several work station users can access the same data without long waiting periods. You can do this by not holding locks while the user is entering data on the display. Some applications may not require more than one work station user accessing the same data. For example, in a cash posting application with many open item records per customer, the typical approach is to lock all the records and hold them until a work station user completes posting the cash for a given receipt.

If the work station user presses the Enter key several times for a transaction, it is possible to perform the transaction in a number of segments. For example:

- The first segment is an inquiry in which the work station user requests the information.
- The second segment is a confirmation of the work station user's intent to complete the entire transaction.
- The third segment is retrieval and update of the affected records.

This approach allows record locking to be restricted to a single use of Enter.

This inquiry-first approach is normally used in applications where a decision results from information displayed. For example, in an airline reservation application, a customer may want to know what flight times, connecting flights, and seating arrangements are available before making a decision on which flight to take. Once the customer makes a decision, the transaction is entered. If the transaction fails (the flight is now full), the rollback function can be used and a different request entered. If the records were locked from the first inquiry until a decision is made, another reservation clerk would be waiting until the other transaction is complete.

For more information on locking considerations and transaction size, see "Considerations for Using Commitment Control."

Record Locking

When a job holds a record lock and another job attempts to retrieve that record for update, the requesting job waits and is removed from active processing until one of the following occurs:

- The record lock is released.
- The specified wait time ends.

More than one job can request a record locked by another job. When the record lock is released, the first job to request the record receives that record. When waiting for a locked record, specify the wait time in the WAITRCD parameter on the following create, change, or override command:

Create Physical File (CRTPF)
Create Logical File (CRTLF)
Create Source Physical File (CRTSRCPF)
Change Physical File (CHGPF)
Change Logical File (CHGLF)
Change Source Physical File (CHGSRCPF)
Override Database File (OVRDBF)

When specifying wait time, consider the following:

- If you do not specify a value, the program waits 60 seconds for a record.
- If the record cannot be allocated within the specified time, a notify message is sent to the high-level language program.
- If the wait time for a record is exceeded, the message sent to the job log gives the name of the job holding the locked record that caused the requesting job to wait. If you experience record lock exceptions, you can use the job log to help determine which programs to alter to not hold locks for long durations. Programs keep record locks over long durations for one of the following reasons:
  - The record remains locked while the work station user is considering a change.
  - The record lock is part of a long commitment control operation. Consider making smaller transactions so a commit operation can be performed more frequently.
  - An undesired lock has occurred. For example, assume a file is defined as an update file with unique keys, and the program updates and adds additional records to the file. If the workstation user wants to add a record to the file, the program may attempt to access the record to determine whether the key already exists. If it does, the program informs the work station user that the request made is not valid. When the record is retrieved from the file, it is locked until it is implicitly released by another read operation to the same file, or until it is explicitly released.

Note: For more information about how to use each high-level language interface to release record locks, see the appropriate high-level language reference manual.

The duration of the lock is much longer if LCKLVL(*ALL) is specified because the record that was retrieved from the file is locked until the next commit or rollback operation. It is not implicitly released by another read operation and cannot be explicitly released.

Minimizing Locks

A typical way to minimize record locks is to release the record lock. (This technique does not work if LCKLVL(*ALL) has been specified.) For example, a single file maintenance application typically does the following:

- Displays a prompt for a record identification to be changed
- Retrieves the requested record
- Displays the record
- Allows the work station user to make the change
- Updates the record

In most cases, the record is locked from the access of the requested record through the update. The record wait time may be exceeded for another job that is waiting for the record. To avoid locking a record while the work station user is considering a change, release the record after it is retrieved from the database (before the record display appears). You then need to access the record again before updating. If the record was changed between the time it was released and the time it was accessed again, you should inform the work station user. The program can determine if the record was changed by saving one or more fields of the original record and comparing them to the fields in the same record after it is retrieved, as follows:

- Use an update count field in the record and add 1 to the field just before an update. The program saves the original value and compares it to the value in the field when the record is retrieved again. If a change has occurred, the work station user is informed and the record appears again. The update field is changed only if an update occurs.

The record is released while the work station user is considering a change. If you use this technique, you must use it in every program that updates the file.

- Save the contents of the entire data record and compare it to the record the next time it is retrieved.

In both cases above, the sequence of operations prevents the simple use of externally described data in RPG where the same field names are used in the master record and in the display file. Using the same field names (in RPG) does not work because the work station user's changes are overlaid when the record is retrieved again.

You can solve this problem by moving the record data to a data structure or continue to use externally described data if you use the DDS keyword RTNDTA. The RTNDTA keyword allows your program to reread data on the display without the operating system having to move data from the display to the program. This allows the program to do the following:

1. Prompt for the record identification.
2. Retrieve the requested record from the database.
3. Release the record.
4. Save the field or fields used to determine if the record was changed.
5. Display the record and wait for the work station user to respond.

If the work station user changes the record on the display, the program uses the following sequence:

1. Retrieves the record from the database again.
2. Compares the saved fields to determine if the database record has been changed. If it has been changed, the program releases the record and issues a message when the record appears.
3. Retrieves the record from the display by issuing a read operation with the RTNDTA keyword and updates the record in the database record.
4. Proceeds to the next logical prompt because there are no additional records to be released if the work station user cancels the request.

The parameter LCKLVL(*CHG) works in this situation.

For V2R1.1

LCKLVL(*CHG) and LCKLVL(*CS) work in this situation.

If LCKLVL(*ALL) is used, you must release the record lock by using a commit or rollback operation.

Using Commitment Control for Batch Applications

Batch applications may or may not need commitment control. In some cases, a batch application can perform a single function of reading an input file and updating a master file. However, you can use commitment control for this type of application if it is important to start it again after an abnormal end.

The input file is an update file with a code in the records to indicate that a record was processed. This file and any files updated are placed under commitment control. When the code is present in the input file, it represents a completed transaction. The program reads through the input file and bypasses any records with the completed code. This allows the same program logic to be used for normal and starting again conditions.

If the batch application contains input records dependent on one another and contains switches or totals, a notify object can be used to provide information on starting again. The values held in the notify object are used to start processing again from the last committed transaction within the input file. See the topic "Notify Object Parameter" on page 12-31.

If input records are dependent on one another, they can be processed as a transaction. A batch job is not permitted to exceed the 32 768 lock limit. Any commit cycle that exceeds 2000 locks will probably slow down system performance noticeably. Otherwise, the same locking considerations exist as for interactive applications, but the length of time records are locked in a batch application is less important than in interactive applications.

Commitment Control Errors

When you use commitment control, it is important to understand which conditions cause errors and which do not. In general, errors occur when commitment control functions are used inconsistently, such as opening a file under commitment control and not specifying the STRCMTCTL command, or issuing an End Commitment Control (ENDCMTCTL) command when uncommitted records exist.

Error Conditions: Following are some typical errors that occur with commitment control.

If an error occurs, an escape message is sent that you can monitor for in a CL program.

- Uncommitted records exist at the end of the job or when an ENDCMTCTL command is issued.

> **For V2R1.1**
>
> This is not an error condition for the ENDCMTCTL command if all files are closed and the job is interactive. Instead, inquiry message CPA8350 is sent to ask the user whether the changes should be committed or rolled back before ending the commitment control environment.

- A commit, rollback, or ENDCMTCTL command is run, and a STRCMTCTL command was not run.
- An ENDCMTCTL command is issued with files still open under commitment control.
- Consecutive STRCMTCTL commands are run without an intervening ENDCMTCTL command.
- Files are opened under commitment control, but no STRCMTCTL command was run.
- Files opened under commitment control are not journaled.
- The files opened under commitment control are not journaled to the same journal.
- The program issues a read operation, a commit operation, and a change to the same record. The read operation must be issued again after the commit operation because the commit operation has freed the lock on the record.
- The first open operation of a shared file places the file under commitment control, but subsequent open operations of the same shared file do not.
- The first open operation of a shared file does not place the file under commitment control, but subsequent open operations of the same shared file do.
- No more than 32 768 records can be locked in a single transaction.

Non-Error Conditions: Following are some situations in commitment control in which no errors occur:

- A commit or rollback operation is run and there are no uncommitted records. This allows you to include commit and rollback operations within your program without considering whether there are uncommitted records.
- A file under commitment control is closed and uncommitted records exist. This situation allows another program to be called to perform the commit or rollback operation. This occurs regardless of whether or not the file is shared. This function allows a subprogram to make database changes that are part of a transaction involving multiple programs.
- A program ends, either normally or abnormally, with uncommitted changes. The changes remain uncommitted. See the discussion in "Monitoring for Errors after a CALL Command" on page 12-40.
- A program accesses a changed record again that has not been committed. This allows a program to:
  - Add a record and update it before specifying the commit operation.
  - Update the same record twice before specifying the commit operation.
  - Add a record and delete it before specifying the commit operation.
  - Access an uncommitted record again by a different logical file (under commitment control).
- You specify LCKLVL(*CHG) on the STRCMTCTL command and open a file with a commit operation for read only. In this case, no locks occur on the request. It is

treated as if commitment control is not in effect, but the file does appear on the WRKJOB menu option of files under commitment control.

- You issue the STRCMTCTL command and do not open any files under commitment control. In this situation, commitment control does not handle any changes made to these files.

Monitoring for Errors after a CALL Command: When a program that uses commitment control is called, you should monitor for unexpected errors and perform a rollback operation if an error occurs. For example, uncommitted records can exist when a program encounters an unexpected error such as an RPG divide-by-zero error. Depending on the status of the inquiry message reply (INQMSGRPY) parameter for a job, the program sends an inquiry message or performs a default action. If the operator response or the default action ends the program, uncommitted records still exist waiting for a commit or rollback operation.

If another program is called and causes a commit operation, the partially completed transaction from the previous program is committed.

To prevent partially completed transactions from being committed, monitor for escape messages after the CALL command. For example, if it is an RPG program, use the following coding:

```
CALL RPGA
MONMSG MSGID(RPG9001)
     EXEC(ROLLBACK) /*Rollback if pgm is canceled*/
```

If it is a COBOL program:

```
CALL COBOLA
MONMSG MSGID(CBE9001)
       EXEC(ROLLBACK) /*Rollback if pgm is canceled*/
```

Performance Considerations for Commitment Control

Using commitment control requires resources that can affect system performance. Several factors affect performance of commitment control:

- Journaling

  Journaling a file requires system resources. If you specify only after-images, commitment control changes this to both before- and after-images while commitment control is in effect. Usually this is a space, not a performance, consideration.

- Journal entries caused by commit or rollback operation

  Each commit or rollback of a transaction writes two entries to the journal (a begin and an end entry) whether or not the user has made changes to the database. The number of entries written can increase significantly for a large volume of small transactions. You may want to place the journal receivers on a user auxiliary storage pool (ASP). For detailed information on user ASPs, see "General Information about Auxiliary Storage Pools" on page 14-7.

- Rollback operation

  Because commitment control must rollback the pending changes recorded in the database, additional system resources are required whenever a rollback occurs. A rollback operation occurring when there are no records to be rolled back causes begin and end entries to be written.

- Start Commitment Control (STRCMTCTL) and End Commitment Control (ENDCMTCTL) commands

  Each time a commitment control environment is established by the STRCMTCTL command, the system creates a commitment definition to save internal control information. The ENDCMTCTL command destroys the commitment definition Therefore, avoid using the STRCMTCTL and ENDCMTCTL commands for each transaction. Use them only when necessary.

  If you open a file without specifying the commit open option within a commitment control environment, no additional system resource is used. For more information about specifying the commit open option, see the appropriate high-level language reference manual.

  You can establish a commitment control environment at the beginning of an interactive job and use it for the duration of the job. Additional programming and system resources for those programs that share files yet do not need the functions of commitment control must be considered. In these cases, try to minimize the use of the STRCMTCTL command.

- Changing the journal used for commitment control

  The same journal must be used for all files under commitment control. If all files under commitment control are closed, and a commit or rollback operation is performed, then files that are journaled to a different journal can be placed under commitment control without ending the commitment control environment. The journal cannot be changed if there are files open under commitment control or a transaction has not been completed.

- Record locking

  Record locking can effect other applications. The number of records locked within a particular job increases the overall system resources used for the job. Applications needing to access the same record must wait for the transaction to end.

- SEQONLY

  If you request the SEQONLY(*YES) option (by using the OVRDBF command or the application program implicitly attempts to use SEQONLY(*YES)) and the file is opened for input only under commitment control, the option is changed to SEQONLY(*NO). This option can affect the performance of input files because records will not be blocked.

Example of Using Commitment Control

The advantages of using commitment control are illustrated in the following example. Assume that the following application program does not use commitment control. The records read for updating are locked by the system. The following steps describe how the application program transfers funds from a savings account to a checking account:

- Program A locks and retrieves the savings record. (This action could require a wait if the record is locked by another program.)

- Program A locks and retrieves the checking record. (This could also require a wait.) Program A now has both records locked, and no other program can change them.

- Program A updates the savings record. This causes the record to be released so it is now available to be read for update by any other program.

- Program A updates the checking record, which causes the record to be released so it is now available to be read for update by any other program.

Without using commitment control, a problem needs to be solved to make this program work properly in all circumstances. For example, a problem occurs if program A does not update both records because of a job or system failure. In this case, the two files are not consistent — funds are removed from the savings account, but they are not added to the checking account. Using commitment control allows you to ensure that all changes involved in the transaction are completed or that the files are returned to their original state if the processing of the transaction is interrupted.

If commitment control is used, the preceding example is changed as follows:

1. Commitment control is started.

2. Program A locks and retrieves the savings record. (This action could require a wait if the record is locked by another program.)

3. Program A locks and retrieves the checking record. (This could also require a wait.) Program A now has both records locked, and no other program can change them.

4. Program A updates the savings record, and commitment control keeps the lock on the record.

5. Program A updates the checking record, and commitment control keeps the lock on the record.

6. Program A commits the transaction. The changes to the savings record and the checking record are made permanent in the files. The changes are recorded in the journal, which assumes they will appear on disk. Commitment control releases the locks on both records. The records are now available to be read for update by any other program.

Because the locks on both records are kept by commitment control until the transaction is committed, a situation cannot arise in which one record is updated and the other is not. If a routing step or system failure occurs before the transaction is committed, the system removes (rolls back) the changes that have been made so that the files are updated to the point where the last transaction was committed.

When using commitment control, ensure that all database files to be placed under commitment control in a single environment are first journaled to the same journal.

For each routing step in which files are to be under commitment control, the steps shown in Figure 12-3 occur:

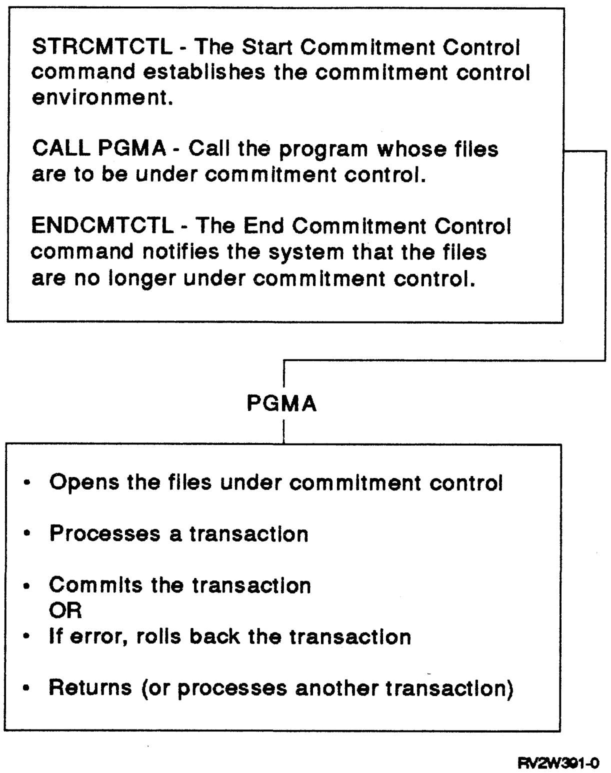


Figure 12-3. Routing Steps with Files under Commitment Control

The operations that are performed under commitment control are journaled to the journal. Using the transfer of funds from a savings account to a checking account as an example, the major entries shown in Figure 12-4 are placed in the journal. (For a description of all the journal entries and their specific content, see the manual *System Operator's Guide*, SC41-8082.)

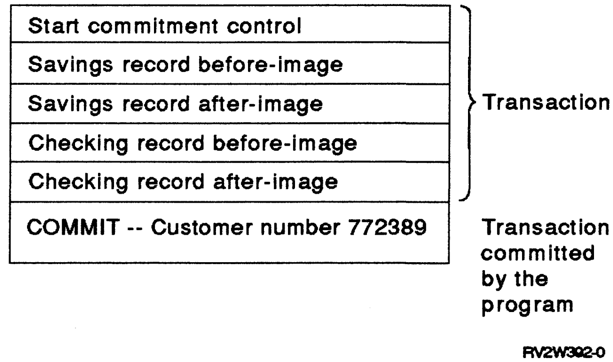


Figure 12-4. Journal Entries for Transfer of Funds from Savings to Checking

The start commitment control journal entry appears after the first file open entry under commitment control. This is because the first file open entry determines what journal is used for commitment control. The journal entry from the first open operation is then used to check subsequent open operations to ensure all files are using the same journal.

When a job failure or system failure occurs, the database base files are updated to a commitment boundary. If a transaction is started but is not completed before a routing step ends, that transaction is rolled back by the system and does not appear in the file after the routing step end, and the record images are placed in the journal as if a commitment will occur. If the system abnormally ends before a transaction is completed, that transaction is rolled back by the system and does not appear in the file after a subsequent successful initial program load (IPL) of licensed internal code. Anytime a rollback occurs, reversing entries are placed in the journal.

For example, assume you have a balance of $100.00 and a customer takes out $20.00, for a new balance of $80.00. The database update causes both before-image ($100.00) and after-image ($80.00) journal entries. The journal entries appear as in the example in Figure 12-5 on page 12-43

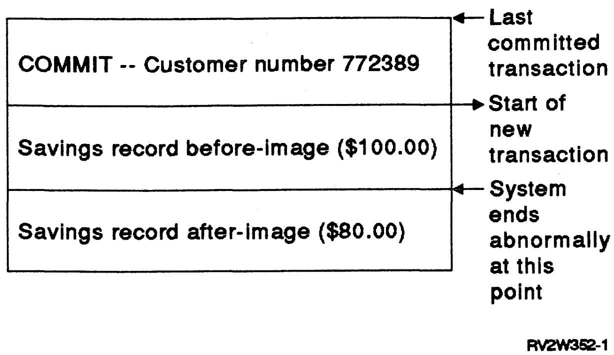


Figure 12-5. Journal Entries for a System That Ended Abnormally

Assume the system abnormally ended after journaling the entries, but before reaching the commitment point or rollback point. After the IPL, the system reads the journal entry and updates the corresponding database record. This update produces two journal entries that reverse the update: the first entry is the before-image ($80.00) and the second entry is the after-image ($100.00) as shown in Figure 12-6.

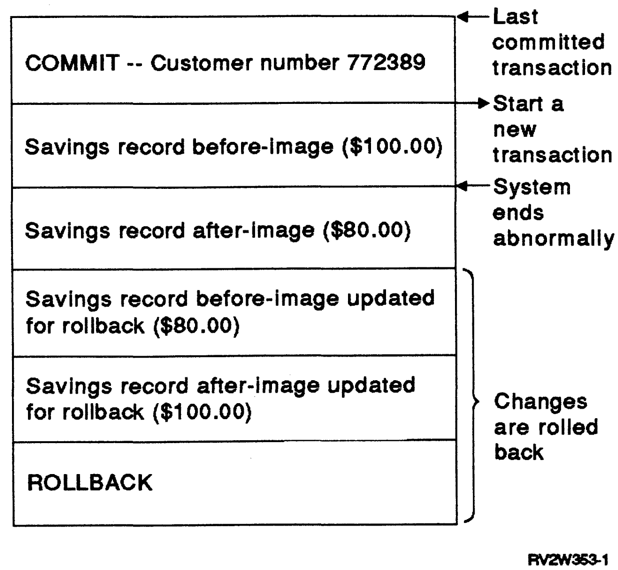


Figure 12-6. Journal Entries for Rollback Changes

When the IPL is successfully completed after the abnormal end, the system removes (or rolls back) any database changes that are not committed. In the preceding example, the system removes the changes from the savings record because a commit operation is not in the journal for that transaction. In this case, the before-image of the savings record is placed in the file. The journal contains the rolled back changes, and an indication that a rollback operation occurred, as in the example shown in Figure 12-6.

Chapter 13. Using Data Recovery Operations

This chapter contains information on:

- Using the Work with Journals command for recovery options
- Examples of recovery operations

Configuring Your Journal

Before you configure your journal, decide whether to:

- Use one or more journals
- Use one or two journal receivers for each journal
- User after-images, or both before-images and after-images of database records
- Put your journal receivers in a user ASP

Isolating your journal receivers in a user ASP other than the ASP that contains the journals and database files provides:

- Better performance of the journal function
- Protection against loss of data on auxiliary storage devices

For additional information on user ASPs, refer to "General Information about Auxiliary Storage Pools" on page 14-7.

Use the following steps to configure your journal:

1. Create the journal receivers using the Create Journal Receiver (CRTJRNRCV) command.
2. Create the journals using the Create Journal (CRTJRN) command.

   Note: All journals and journal receivers on the system should have unique names. For example, try not to have JRNA in library A and library Y. If both journals become damaged and a reclaim storage (RCLSTG command) operation is required, the duplicate journals will be renamed when they are placed in in library QRCL. The system will not allow you to rename those journals or journal receivers placed in library QRCL.
3. Use the Start Journaling Physical File (STRJRNPF) command to start journaling all members of a physical file. The library for the journal and the library for the files must be in the same ASP before journaling can be started.
4. Optionally, start journaling the access paths of one or more physical or logical files using the Start Journaling Access Paths (STRJRNAP) command. All members for each file have their access path journaled. See "Access Path Recovery" on page 12-26 for a complete discussion of journaling access paths and access path recovery.
5. Save the files and members after you start journaling. You cannot apply journaled changes to files and members using FROMENT (*LASTSAVE) on the Apply Journal Change (APYJRNCHG) command that were not being journaled when they were saved. If you do save access paths (when saving the

files), the system does not rebuild them after restoring the files and access paths. Saving access paths does not affect IPL recovery (or delayed recovery) of access paths.

Run the applications that use those journaled files when you have completed these steps.

Creating a Journal Receiver

You must create journal receivers before you create the journal to which they will be attached. To create a journal receiver, use the CRTJRNRCV command and specify the name of the receiver when you create it. The following command creates a journal receiver RCVDST1:

```
CRTJRNRCV JRNRCV(DSTJRN/RCVDST1) THRESHOLD(10000)
          TEXT('RECEIVER FOR DSTJRN JOURNAL')
```

In the above example, the receiver is placed in the library DSTJRN.

To create the journal receiver in a user ASP, first create the library for the journal receiver in the user ASP, or if the desired library is in the system ASP, specify a user ASP number for the ASP parameter on the CRTJRNRCV command. If you do use the ASP parameter, only journals, journal receivers, and save files will be allowed in the user ASP. You can use the THRESHOLD parameter to have a message sent to a message queue when the receiver reaches a specified size. In the above example, if the size of the receiver exceeds 10000KB (the THRESHOLD parameter value), the system sends a message to the message queue specified on the CRTJRN or CHGJRN command and journaling continues. You may want to change the receiver when this occurs.

The UNIT parameter on the CRTJRNRCV command is no longer supported. The UNIT parameter is kept only for syntax compatibility with previous releases of the Operating System/400.

Creating a Journal

Journals are created using the Create Journal (CRTJRN) command after the journal receiver has been created. Specify the name of the journal and one or two journal receivers. For example, the following command creates a journal with the name JRNLA in library DSTJRN and attaches a journal receiver (RCVDST1) to the journal.

```
CRTJRN JRN(DSTJRN/JRNLA) JRNRCV(DSTJRN/RCVDST1)
     MSGQ(QSYS/QSYSOPR) TEXT('Distribution Journal')
```

To specify that the journal be created in a user ASP, first create the library for the journal in the user ASP. If the library for the journal is in the system ASP, it is possible to specify a user ASP number for the ASP parameter on the CRTJRN command. However, it is not recommended. New journal receivers are attached to the journal when you run the Change Journal (CHGJRN) command.

Display Journal Status

The Work with Journal Attributes display lets you display information about the journal and related journal receivers. Use the WRKJRNA command. This display identifies:

- The journal receivers currently attached to the journal
- A directory of the journal receivers still on the system that are associated with the journal
- The names of all physical files being journaled through the journal
- The names of all database files that are having their access paths journaled

Journal Receiver Directory

Figure 13-1 shows an example of the Work with Receiver Directory display.

```
                         Work with Receiver Directory

Journal . . . . . . :   JRNLA          Library . . . . . . :   DSTJRN

Total size of receivers . . . . . . . . . . . . . . . . . . :   155648

Type options, press Enter.
  4=Delete    8=Display attributes

                                    Attach                        Save
Opt  Receiver    Library     Number  Date      Status             Date
 _   RCVDST1     DSTJRN      02001   06/08/90  ONLINE             00/00/90
 _   RCVDSTA1    DSTJRN      03001   06/09/90  PARTIAL            06/90/90
 _   RCVDSTA2    DSTJRN      03001   06/09/90  ONLINE             00/00/90
 _   RCVDSTB1    DSTJRN      03002   06/10/90  PARTIAL            06/11/90
 _   RCVDSTB2    DSTJRN      03003   06/10/90  SAVED              06/11/90
 _   RCVDSTC1    DSTJRN      04001   06/11/90  ATTACHED           06/12/90



                                                                      Bottom
F3=Exit    F5=Refresh    F11=Display size    F12=Cancel
```

Figure 13-1. Work with Receiver Directory

The *PARTIAL* status of the journal receiver on this display indicates the following:

- A journal receiver was saved while it was attached to the journal. This means that additional entries will be recorded in the journal attached to this receiver after the save operation has occurred.
- The receiver was later restored, and no complete version is available.
- A partial receiver does not contain all the entries recorded in the journal while this receiver was attached. It does contain entries recorded up to the last save operation.

You can use partial receivers to apply or remove changes from a file. If you attempt to restore a saved receiver while a more current version of the receiver is on the system, an escape message is sent to prevent you from restoring the receiver. The system makes sure the most complete version is preserved.

You can use a partial receiver as the last receiver in the receiver chain for an APYJRNCHG command only if you specify a sequence number for the TOENT parameter. You can use a partial receiver as the first receiver in the receiver chain for a RMVJRNCHG command only if you specify a sequence number for the FROMENT parameter.

The system does not allow you to delete a journal receiver from the middle of the receiver chain. This ensures logical recovery. However, if a journal receiver is damaged, you can delete it from the middle of the chain. If an attached journal receiver is damaged, you must detach the damaged receiver (CHGJRN command) before you can delete it.

The system does not prevent you from deleting a receiver that was once attached and is not saved or that is required to provide adequate recovery. If you try to delete a journal receiver that was once attached but has not been saved, the system issues an inquiry message. You can then continue or cancel the delete operation. You may use the system reply list to specify the reply the system is to send for this inquiry message (rather than explicitly responding to each inquiry message).

You must ensure that the journal receivers are not deleted until you no longer need them. You can display the journal receiver directory from the Journal Attributes display to determine which journal receivers have been saved. A date of 00/00/00 in the *Saved* column indicates that a journal receiver has not been saved.

Start Journaling Physical File

When using the Start Journal Physical File (STRJRNPF) command, you can specify the OMTJRNE (*OPNCLO) parameter to omit the open and close operations for specific files from being journaled. This can be done to reduce the number of entries in a journal. Be aware that without open and close entries in the journal, the history of file access (open for input, output, update, or delete) does not exist.

The use of the TOJOBO and TOJOBC parameters on APYJRNCHG and RMVJRNCHG commands are affected because these parameters specify that the apply or remove operation continue until the specified job first opens or last closes any physical file member.

An alternative way to reduce the number of open and close entries in a journal is to specify SHARE (*YES) for the file. The system writes a single open and close entry regardless of how often the shared open data path (ODP) is opened or closed within a routing step.

The following command starts journaling for the physical file ORDENTP:

```
STRJRNPF  FILE(DSTPRODLIB/ORDENTP) JRN(DSTJRN/JRNLA)
       IMAGES(*BOTH)
```

After you have run this command, any changes made to the physical file ORDENTP are recorded in the journal JRNLA whether the changes are made directly to the physical file ORDENTP or through logical files of that physical file. Both before- and after-images will be journaled.

End Journaling Physical File

Use the End Journal Physical File (ENDJRNPF) command to end journaling of changes for a specific physical file and all its members. Journaling of a member stops if you remove it from a journaled file. Journaling of a file stops if you delete it.

Changing Journaling to a Different Journal

You can change the journal that a file is journaled to. To change journals for a physical file, do the following:

1. Allocate the file and its members using the ALCOBJ command to prevent changes to the file while you are changing from one journal to another.
2. End journaling of the physical file with the ENDJRNPF command.
3. Start journaling the file to the new journal using the STRJRNPF command.

Saving Files

Save physical files after you start journaling them. Logical files dependent on the physical files, should also be saved in case the physical file becomes damaged. If you save access paths (when saving the files) the system does not have to rebuild the access paths after restoring the files and access paths. You can use the Save Changed Objects (SAVCHGOBJ) command and specify OBJTYPE(*FILE) OBJJRN(*YES), or you can use the Save Object (SAVOBJ) or Save Library (SAVLIB) command to save files. Saving access paths does not affect IPL recovery.

Journaled changes are associated with files by a mark placed on the file when journaling of the file began. Thus, the version of a file saved before journaling began is not marked, and journaled changes cannot be applied to that version of the file. When you save the files, make sure that no changes can be made to the files by using the Allocate Object (ALCOBJ) command to allocate each member of the file or by ceasing all activity on the files while they are being saved.

You should also save a journaled file after you add a member to the file so the member can be recovered in case it becomes damaged.

Start Journaling Access Paths

The Start Journal Access Path (STRJRNAP) command is used to start journaling, to a specific journal, an access path or access paths for all members of a data-base file. Any new members that are added to the file will also have their access paths journaled.

If the physical file is keyed, journaling can be started for the file's access path. When access path journaling is started for the physical file, only the access paths for the physical file members are journaled. Journaling for any logical files is started only when access path journaling is started for the logical files.

The journal entries created for access path journaling cannot be used in any operation that applies or removes journal changes. These entries are used only to recover access paths after the system ends abnormally.

Follow these steps to begin access path journaling:

1. Identify those access paths you wish to journal. Consider journaling your large access paths (those that would take a long time to rebuild if the system ended abnormally).
2. Use the Start Journal Physical File (STRJRNPF) command to journal all physical files associated with the access paths you want protected (if you are not already journaling the physical files).
3. Start journaling the selected access paths using the Start Journal Access Path (STRJRNAP) command. Remember that you can journal both physical file and logical file access paths. Specify the file whose access path you want to journal on the STRJRNAP command. The following command begins journaling the access path for the logical file ORDENTL:

   ```
   STRJRNAP FILE(DSTPRODLIB/ORDENTL) JRN(DSTJRN/JRNLA)
   ```

Ending Access Path Journaling

To end journaling the access paths for a physical or logical file, use the End Journal Access Path (ENDJRNAP) command. In addition, if you remove a member from a file that is journaling its access path, journaling for that member's access path is implicitly ended, unless the access path is shared and journaled by another file member. If a file that is journaling its access paths is deleted, journaling is implicitly ended for all the file member's access paths unless a particular access path is being shared and journaled by another file.

A physical file cannot have journaling ended for its members if the physical file or any logical file based on that physical file is journaling its access path. All physical files under a logical file must be journaled if that logical file is journaling its access path.

Work with Journal (WRKJRN) Command Options

The Work with Journal (WRKJRN) command shows a display that provides you with options to display journal status and guides you through various types of recovery involving journals, journal receivers, and journaled files. The options are shown on the following display:

```
                        Work with Journals

Type options, press Enter.
  2=Forward recovery       3=Backout recovery   5=Display journal status
  6=Recover damaged journal   7=Recover damaged journal receivers

Opt  Journal     Library     Text
     JRNACC      DSTA1       JOURNAL FOR ACCOUNTS

                                                                  Bottom
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```


Recovery Options

When it is necessary to restore an object during recovery procedures, the system prompts for the values needed with a prompt of the RSTOBJ command. The values known for OBJ, SAVLIB, and OBJTYPE parameters appear, but they cannot be changed. If the values for DEV, VOL, SAVF, SAVDATE and SAVTIME can be determined by the system (depending on the amount of damage), they are also filled in. If the values are not shown, you must enter the values.

If more than one object is restored, the system groups the objects according to the library specified in the SAVLIB parameter. Within each group, the system further orders the objects according to the values specified for the DEV, VOL, SAVF, SAVDATE, and SAVTIME parameters. This grouping results in a minimum number of restore object prompts as a result of information known to the system.

Work with Forward Recovery

Option 2 on the Work with Journals display shows the Work with Forward Recovery display and a list of the file members being journaled.

```
                         Work with Forward Recovery

Journal . . . . . . :   JRNACC        Library . . . . . . :   DSTA1

Position to . . . . .
  Library . . . . . .

Type options, press Enter.
  1=Add member to list   2=Apply journaled changes   3=Restore
  4=Remove member from list

Opt     File          Library       Member        Status




                                                                  Bottom
F3=Exit   F12=Cancel
```

The Work with Forward Recovery display contains a status field for each file member. The status field for each member indicates the following:

- NOT FOUND. The system cannot locate the specified database file.
- DAMAGED. The member is damaged and needs to be restored.
- NOT SYNCHRONIZED. The journal receiver used for this member is damaged and needs to be restored.
- RESTORE COMPLETE. The restore of the member is complete.
- RECOVERED. Recovery has completed successfully.
- NOT JOURNALED. The member is not journaled to any journal and cannot be recovered.
- DIFFERENT JOURNAL. The member is not journaled to the journal you are working with.
- Blank. The member and the journal are usable and everything is synchronized.

Option 1 (Add member to list) on the display allows you to add a member to the list. You may want to do this if you want to restore those members.

Option 2 (Apply journaled changes) shows the APYJRNCHG command prompt, which applies the journaled changes to the file members, and changes the status to RECOVERED (if the apply operation was successful). If the apply operation was not successful, messages appear indicating why, and the status remains the same.

If any required receivers are missing or damaged while running the APYJRNCHG command, the system displays prompts for the restore procedures for the missing or damaged receivers.

If any of the members in the list have a status of DAMAGED when option 2 is selected, you are prompted with the command necessary to recover the file member. For files that are damaged, recovery involves the restore of the last save followed by the Apply Journal Changes (APYJRNCHG) command. The system guides you through recovery as follows:

1. The system identifies all the logical files dependent on the specified damaged file. The Dependent Logical Files display appears identifying these files.
2. The dependent logical files are deleted.
3. The files to be recovered (restored) are deleted by the system.
4. The system displays prompts for the restore of files to be recovered. After all restores are completed successfully, the files to be recovered are allocated exclusively to prevent any other processing. This allocation is maintained until the recovery procedures are complete.
5. The system displays prompts for the restores of the dependent logical files.
6. An APYJRNCHG command is prompted with FROMENT(*LASTSAVE) and TOENT(*LASTRST).
7. If the APYJRNCHG command encounters a required journal receiver that is not online, the system prompts for the restore of the required receiver and again starts the APYJRNCHG command.

When the recovery process is complete, the status field for the member indicates RECOVERED (if the operation was successful). If the operation failed, the status field remains unchanged, and messages appear indicating why the operation failed.

Option 3 (Restore) prompts you for the files to restore. Use this option if any members have a status of NOT FOUND. Members that are restored successfully have a status of RESTORE COMPLETE. Members that are not restored keep their old status. A message is sent indicating that the restore did not complete successfully. All members restored are included in the list of members to recover.

Option 4 (Remove member from list) on the display removes file members from the list of members to be recovered.

Work with Backout Recovery

The same options on the Work with Forward Recovery display are available on the Work with Backout Recovery display (option 3 on the Work with Journals display) with the exception of allowing you to prompt for the Remove Journal Changes(RMVJRNCHG) command, rather than the APYJRNCHG command. In addition, the option to restore the file is not valid for backout recovery. The status field shown on the Work with Backout Recovery display is either blank or it indicates the same status as for forward recovery, except for RESTORE COMPLETE.

```
                         Work with Backout Recovery

Journal . . . . . . :   JRNACC          Library . . . . . . :   DSTA1

Position to . . . . .
  Library . . . . . .

Type options, press Enter.
  1=Add member to list   2=Remove journaled changes
  4=Remove member from list

Opt     File            Library         Member          Status









                                                                      Bottom
F3=Exit   F12=Cancel
```

Option 1 (Add member to list) on the display allows you to add a member to the list.

Option 2 (Remove journaled changes) on this display shows the Remove Journal Changes (RMVJRNCHG) prompt, removes the journaled changes, and changes the status to RECOVERED (if the operation was successful). If any required journal receivers are missing or damaged while the RMVJRNCHG command is running, the system displays prompts for the necessary restore procedures for the missing or damaged receivers. If the remove operation was not successful, messages appear indicating why the status remains the same.

If any members in the list have a status of NOT FOUND or DAMAGED when on the Work with Backout Recovery display, the operation is not allowed. These members must be recovered in a forward fashion after they have been restored. Forward recovery of specific files must be used for this type of recovery.

Option 4 (Remove member from list) on this display allows you to remove file members from the list.

Display Journal Status

Option 5 on the Work with Journal display shows the current status of the journal. It shows if the last system end was NORMAL or ABNORMAL, and if the journal is damaged. The damage status is NONE or FULL.

```
                          Display Journal Status

Journal . . . . . . :   QAUDJRN        Library . . . . . . :   QSYS

Last system end status . . . . . . . . . . . . . :   NORMAL
Journal damage status . . . . . . . . . . . . . . :   NONE
All objects synchronized . . . . . . . . . . . . :   YES

Attached                      Damage
Receiver      Library         Status
QAUDRC0002    QGPL            NONE

                                                                Bottom
Press Enter to continue.

F3=Exit   F12=Cancel
```


If the last system end was abnormal, this display indicates whether objects being journaled were synchronized (that is, whether each object in use during the abnormal end was corrected to match the entries in the attached journal receivers during the previous IPL).

If the last system end was normal, the display indicates that all objects are synchronized with the journal. If the journal is damaged, the display indicates that the system was unable to determine whether or not all objects are synchronized.

The display also presents information about the currently attached receivers and their damage status. The damage status of the receivers can be NONE, PARTIAL, or FULL. If the journal is damaged to the extent that the status of the attached journal receivers cannot be determined, there are no attached receivers shown on the display.

If some objects are not synchronized or damage has been detected, a message appears indicating the form of recovery that you should perform.

Recover Damaged Journal

Option 6 on the Work with Journals display verifies that the journal is damaged before proceeding with recovery. If the journal is not damaged, an information message appears.

Recovery for a damaged journal guides you through the following steps:

1. The system attempts to determine which files are currently being journaled to the indicated journal. If the system cannot successfully build this list, a message appears before the recovery operation begins.
2. Journaling is ended for all access paths currently being journaled to the specified journal.
3. Journaling is ended for all files currently being journaled to the specified journal.
4. The system deletes the journal.

5. The system presents the Recover Damaged Journal display, which asks you whether to restore or create the journal:
   a. If the journal will be restored, the system prompts for the values needed for the restore operation.
   b. If the journal will be created, the system prompts for the receiver names and attributes with the CRTJRNRCV command prompt. The system prompts for values needed to create the journal with the CRTJRN command prompt, with known values shown.
6. The list of files for which journaling is to be started again is shown. When you press the Enter key, journaling is started for all files listed.
7. The list of files containing access paths for which journaling is to be started again appears. When you press the Enter key, journaling for the access paths is started for the files listed.
8. The system associates all applicable receivers with the re-created or restored journal so that a restore of these receivers is not necessary.

As the recovery of a damaged journal proceeds, the Display Journal Recovery Status display appears. The information on this display is updated as the operation progresses to indicate which steps have been completed, which steps have been bypassed, and which step will be run next. Whenever a user action is required, the status display is replaced by the appropriate prompt display.

The status field indicates the following operation status:

- PENDING. The step has not been started.
- NEXT. The step will be performed next (after the Enter key is pressed).
- BYPASSED. The step was not performed. (It was not necessary).
- COMPLETE. The step has been performed.

The first display you usually see after the first status display is the Recover Damaged Journal display. Use this display to choose whether the journal is to be created or restored.

When the last step of the recovery process is complete, a message appears indicating that all files for which journaling was started should be saved to establish a new recovery point.

Recover Damaged Journal Receivers

Option 7 on the Work with Journals display checks to determine which journal receivers associated with the specified journal are damaged. If none are damaged, a message appears.

If there are damaged journal receivers associated with the specified journal, the Recover Damaged Journal Receivers display appears and lists those receivers.

The status fields initially show a value of DAMAGED. After recovery has been successfully completed, the status shows a value of RECOVERED (receiver recovered).

Recovery for a damaged journal receiver guides you through the following steps:

1. If the attached receivers are damaged, a CHGJRN command must be run before they can be recovered.

Indicate whether existing (empty) receivers will be used or if new ones will be created. If new receivers are to be created, the CRTJRNRCV command prompts are presented for receiver name and attributes. After the new receivers are created, the CHGJRN command prompt is shown.

If the attached receiver is not damaged, the preceding step is omitted.

2. The damaged journal receiver is deleted.
3. Prompts for the restore of the damaged journal receivers are shown. Any of the values on the prompt can be changed except the receiver name. Save information in the prompt is provided by the system.

Recovery of a Physical File Using Journaled Changes

In addition to the file recovery information that follows, you can use the Work with Journals (WRKJRN) command to help recover journaled files.

You can recover from many types of damage to database file members using journaled changes. For example, a file member is damaged and becomes unusable, an error in an application program caused records to be improperly updated, or incorrect data was used to update a file. In each of these instances, simply restoring a saved version of the file may result in the loss of a significant amount of data. If you use the Apply Journal Changes (APYJRNCHG) command to apply journaled changes, significantly less data may be lost. You can use the Remove Journal Changes (RMVJRNCHG) command to recover from improperly updated records or incorrect data if before-images have been journaled. This command removes (or backs out) changes made to a file.

Note: To apply or remove journaled changes to or from a saved copy of the file, the file must have been saved while it was being journaled.

For an explanation of why files must be saved, see "Saving Files" on page 13-5.

Recovery after Abnormal System End

If the system abnormally ends while files are being journaled, the system does the following:

1. Brings all journals, journal receivers, and files being journaled to a usable and predictable condition during the IPL, including any access paths being journaled and in use at the time the system abnormally ended.
2. Checks all recently recorded entries in the journal receivers that were attached to a journal.
3. Places an entry in the journal to indicate that an abnormal system end occurred. When the system completes the IPL, all entries are available for processing.
4. Checks that the journal receivers attached to journals can be used for normal processing of the journal entries. If some of the files being journaled could not be synchronized with the journal, the system sends a message (CPF3172) to the history log (QHST) that identifies the journals that could not be synchronized. If a journal or a journal receiver is damaged, the system sends a message to the history log identifying the damage that occurred (CPF3171 indicates that the journal is damaged, and the message CPF3173 or CPF3174 indicates that the journal receiver is damaged).
5. Recovers each physical file member that was in use at the time the system ended abnormally, using the normal system recovery procedures for data-

base files. See the manual *Database Guide*, SC41-9659, for complete information on database files.

In addition, if a physical file being journaled (or a logical file defined over that physical file) was opened for output, update, or delete operations, the system performs the following functions so changes to that file will not be lost:

a. Ensures that the changes appearing in the journal receiver after the IPL also appear in the database file. Changes that do not appear in the journal receiver are not in the database file.

b. Places an entry in the journal receiver indicating whether the file was synchronized with the journal. If the file could not be synchronized with the journal, the system places a message (CPF3175) in the history log identifying the failure, and you must correct the problem.

A synchronization failure can occur if the data portion of the member is damaged, a journal receiver required to perform the synchronization is damaged, or the journal is inoperable.

If an abnormal system end occurs, follow these steps:

1. Perform an IPL.

2. Check the history log to determine if there are any damaged files, files that are not synchronized, or any damaged journals or journal receivers.

   An alternative is to use the WRKJRN command. See "Work with Journal (WRKJRN) Command Options" on page 13-6 for more information.

3. If necessary, recover the damaged journals or journal receivers as described in "Recovering When a Journal Is Damaged" on page 13-15 and "Recovering When a Journal Receiver Is Damaged" on page 13-16.

4. If there is a damaged file:

   a. Delete the file.
   b. Restore the file from the latest saved version.
   c. Allocate the file so no one else can access it.
   d. Restore the needed journal receivers if they are not online.
   e. Use the APYJRNCHG command to apply the changes to the file.
   f. Deallocate the file.

   WRKJRN can be used to recover damaged files. See "Work with Journal (WRKJRN) Command Options" on page 13-6.

5. If a file could not be synchronized, use the information in the history log and in the journal to determine why the file could not be synchronized and how to proceed with recovery. For example, you may need to use the DFU or a user-written program to bring the file to a usable condition.

6. Determine which applications or programs were active, and determine where to restart the applications from the information in the history log and in the journal.

If a journaled access path is in use during an abnormal system end, that access path does not appear on the access path recovery override display.

If the maintenance for the access path is immediate, the system automatically recovers the access path during IPL. A status message appears for each immediate maintenance access path as it is being recovered during IPL.

The system places a message (CPF3208) in the system history log for each access path recovered through the journal during IPL. If the maintenance for the access path is delayed, the system automatically recovers the access path during the next database file open operation using the access path entries recorded in a journal.

Recovering When a Journal Is Damaged

If a journal becomes damaged, the system sends the message CPF8135 to the system operator and to the job log. Use the Work with Journal (WRKJRN) command to help you in the recovery of a damaged journal. Select option 6 (Recover damaged journal) on the Work with Journals display to recover a damaged journal. For a description of the Work with Journals display, see "Work with Journal (WRKJRN) Command Options" on page 13-6, or the WRKJRN command in the manual *Programming: Control Language Reference*, SC41-0030.

It is recommended that you use the Work with Journals (WRKJRN) command to recover a damaged journal. The WRKJRN command performs all the steps described below except for saving the physical files and logical files. The WRKJRN command associates the receivers with the recovered journals without you having to delete and restore the receivers.

Use the following steps to recover a damaged journal without using the WRKJRN command:

1. End journaling for all access paths associated with the journal by using the ENDJRNAP command.
2. End journaling for all physical files associated with the journal by using the ENDJRNPF command.
3. Delete the damaged journal by using the DLTJRN command.
4. Create a journal receiver (CRTJRNRCV command), and create a journal with the same name and in the same library as the damaged journal (CRTJRN command), or restore the journal from a previously saved version.
5. Start journaling the physical files and, if desired, the access paths by using the Start Journal Physical File (STRJRNPF) and Start Journal Access Path (STRJRNAP) commands for the files, or by deleting and restoring all the files that were being journaled. Physical files and access paths that were journaled at the time of their save automatically begin journaling at restore time if the journal is online.
6. Save the physical files and all associated logical files to allow for later recovery.
7. Associate the old journal receivers with the new journal. Save the journal receiver that was attached to the damaged journal. Delete it and restore it and any previously attached journal receivers you need. You must restore the receivers after the journal is restored or re-created in order to associate the journal receivers with the journal. To display the names of the associated journal receivers, use the Work with Journal Attributes (WRKJRNA) command and take the option to Work with Receiver Directory.

Each time a journal is restored, a new receiver chain is started because the last journal receiver on the chain that existed prior to the restore process did not have the newly created receivers as its next receivers.

Recovering When a Journal Receiver Is Damaged

If a journal receiver becomes damaged, the system sends the message CPF8136 or message CPF8137 to the system operator and the job log. To recover from a damaged receiver, do the following:

- If the damaged receiver is currently attached to a journal, use the Change Journal (CHGJRN) command to attach a new receiver and detach the damaged receiver. (If dual receivers exist, the system continues to place entries on the receiver that is not damaged until a CHGJRN command is run.)
- If the journal receiver is not currently attached to a journal, delete the journal receiver using the Delete Journal Receiver (DLTJRNRCV) command and restore a previously saved copy.
- If the journal receiver was never attached to a journal, delete the receiver and create it again or restore it.

If the journal receiver is partially damaged, all journal entries except those in the damaged portion of the journal receiver can be listed using the Display Journal (DSPJRN) command. Using this list, you can determine what you need to do to recover your files. Applying or removing journal changes cannot be done with a partially damaged journal receiver.

It is recommended that you use the Work with Journals (WRKJRN) command to recover a damaged journal receiver. For a description of the journal menus, see the online information.

Journaling after a File Recovery

Use the following steps after you complete a recovery operation using journaling.

1. Attach a new journal receiver.
2. Save the detached journal receiver.
3. Save the journal.
4. Save the files and all associated logical views.

Save the file after you add a member to the file so the member can be restored if it becomes damaged.

Applying and Removing Journaled Changes

Two commands, Apply Journal Changes (APYJRNCHG) and Remove Journal Changes (RMVJRNCHG), are used to apply and remove changes made to a file member.

Applying Journaled Changes

If a file member becomes damaged or is not usable, you can recover the file using the Apply Journal Changes (APYJRNCHG) command directly, or by using the Work Journal (WRKJRN) command and following the prompts. For a description of the journal menus, see the online information or refer to the topic "Work with Journal (WRKJRN) Command Options" on page 13-6. You must first reestablish the physical file member to a condition that you know is undamaged.

- To reestablish the member, restore the last saved copy of the file. The file must have been saved while it was being journaled.

- Use the WRKJRN command options to help in the restore or apply operation.
- If you saved the file using the CPYF command, use the CPYF command to restore the member. For information about the CPYF command, see the manual *Programming: Control Language Reference*, SC41-0030.
- If the member was just initialized, initialize the member again using the Initialize Member (INZPFM) command or a user-created application program.
- If a member was just reorganized, reorganize the member again using the Reorganize Physical File Member (RGZPFM) command. The manual *Programming: Control Language Reference*, SC41-0030, has more information about the RGZPFM command.
- If the file or member was just created, create the file or member again.

If the journal receivers are deleted or saved with their storage freed since the file was last saved (or since some other point), you must restore the needed journal receivers. The system applies the changes to the file in the same order as they were originally made. When you use the APYJRNCHG command, the file cannot be in use by anyone else.

When the condition of the member has been established, use the APYJRNCHG command to apply the changes recorded in the journal to the file. On the APYJRNCHG command, specify the first journal entry to be applied to the member. This entry can be selected from any of the following points:

- After the last save of the member
- From the first journal entry
- From an identified sequence number that corresponds to a date and time stamp
- From an identified sequence number that corresponds to the start or end of a particular job's use of the member (if you did not specify OMTJRNE (*OPNCLO) when starting journaling for the file)
- From a commitment boundary

You can control which changes are applied to the file. For example, assume an application was updating entries incorrectly for a certain period of time. In this case, apply the changes for the member only to the point where the application first opened the member.

You can stop applying the journal entries at:

- The end of the data in the last journal receiver in the receiver range
- A particular entry in the journal
- A date and time stamp
- A commitment boundary
- The start or end of a particular job's use of the data in the member (provided you did not specify OMTJRNE (*OPNCLO))
- The journal entry indicating when the member was last restored

Use the Display Journal (DSPJRN) command to identify the desired starting and ending points. If you use a control language (CL) program for your recovery procedures, use the Retrieve Journal Entry (RTVJRNE) command to retrieve a journal entry and place it in program variables. See "Using the Retrieve Journal

Entry (RTVJRNE) Command in a Program" on page B-5 for more information. The Receive Journal Entry (RCVJRNE) command can also be used for recovery.

Apply Journal Changes (APYJRNCHG) Command Examples

The following command applies the changes in journal JRNA to the first member of all files in the library DSTPRODLIB that are being journaled to journal JRNA:

```
APYJRNCHG JRN(JRNLIB/JRNA) FILE((DSTPRODLIB/*ALL))
```

Because the RCVRNG parameter is not specified, the system determines the range of journal receivers to use as a result of the save information for the files. Because the FROMENT parameter is not specified, the system applies the changes beginning with the first journal entry after the member was last saved.

The following command applies the changes to the file from the journal receivers currently attached to the journal:

```
APYJRNCHG  JRN(JRNLIB/JRNA) FILE((LIBA/FILEA MBR1))
           RCVRNG(*CURRENT) FROMENT(*FIRST) TOENT(*LAST)
```

The system applies the changes from the first journal entry in the currently attached receiver to the last journal entry in the currently attached receiver. Changes are applied to member MBR1 of the file FILEA.

The following command applies the changes in the journal JRNA to all members of the file FILEA beginning with the first journal entry after the file member was last saved:

```
APYJRNCHG  JRN(JRNLIB/JRNA)   FILE((LIBA/FILEA *ALL))
           TOJOBC(000741/USERP/WORKSTP)
```

The operation continues until the last specified job closes any of the members in the file that it opened. The operation is not restricted only to those journal entries recorded by the specified job.

Note: This example works only if you do not specify OMTJRNE (*OPNCLO).

Removing Journaled Changes

Depending on the type of damage to the physical file and the amount of activity since the file was last saved, removing changes from the file can be easier than applying changes to the file. Use the Remove Journal Changes (RMVJRNCHG) command directly or the Work with Journal (WRKJRN) command and follow the prompts to remove (or roll back) changes from a file member if before-images were journaled. For a description of the journal menus, see the topic "Work with Journal (WRKJRN) Command Options" on page 13-6, or the online information for the WRKJRN command. The changes are removed in reverse chronological order from the order they were originally made to the file.

On the RMVJRNCHG command, you identify the first journal entry to be removed from the file member. This entry can be from:

- The last journal entry contained within the range of journal receivers specified
- A commitment boundary
- An identified sequence number

You can control the changes that are removed from the file. For example, assume an application updated entries incorrectly for a period of time. In this case, you could remove the changes from the member until that application first opened the member.

You can stop removing journaled changes at:

- The end of data in the journal receivers. (This corresponds to the first journal entry that was recorded on the range of journal receivers specified.)
- An identified sequence number that corresponds to a particular entry in the journal.
- The start of a particular job's use of the member (if you did not specify OMTJRNE (*OPNCLO)when starting journaling for the file).
- A commitment boundary.

Use the Display Journal (DSPJRN) command to identify the desired starting and ending points for removing the changes. If you use a control language (CL) program for your recovery procedures, use the Retrieve Journal Entry (RTVJRNE) command to retrieve a journal entry and place it in program variables. See "Retrieving a Journal Entry" on page 12-23 for more information about retrieving journal entries. The Receive Journal Entry (RCVJRNE) command can also be used for recovery.

Remove Journaled Changes (RMVJRNCHG) Command Examples

The following command removes the changes in journal JRNA from the first member of FILEA:

```
RMVJRNCHG JRN(JRNLIB/JRNA) FILE(DSTPRODLIB/FILEA)
          RCVRNG(*CURRENT)
```

The system starts removing the changes beginning with the latest entry for that member on the currently attached journal receiver and continues to the earliest entry for that member on the currently attached journal receiver.

The following command removes the changes in journal JRNA from the first member of FILEA:

```
RMVJRNCHG JRN(JRNLIB/JRNA) FILE(DSTPRODLIB/FILEA)
          RCVRNG(JRNLIB/RCVA10 JRNLIB/RCVA8)
```

The system starts removing the changes beginning with the last entry (the latest entry) for that member in journal receiver RCVA10 and continues to the first entry (the earliest entry) for that member on journal receiver RCVA8.

Actions of the APYJRNCHG or RMVJRNCHG Command by Journal Code

Table 13-1 on page 13-20 shows the actions taken by the APYJRNCHG or RMVJRNCHG command following actions by journal code and entry type.

Actions of the APYJRNCHG or RMVJRNCHG Command by Journal Code

Table 13-1 (Page 1 of 4). Actions by Journal Code and Entry Type

Journal Code	Entry Type	Operation	APYJRNCHG	RMVJRNCHG
A	DP	Direct print information	Ignores	Ignores
A	JB	Job resource information	Ignores	Ignores
A	SP	Spooled print information	Ignores	Ignores
C	BC	Start commitment control	Ignores	Ignores
C	CM	Commit record changes	Ignores	Ignores
C	EC	End commitment control	Ignores	Ignores
C	RB	Roll back changes	Ignores	Ignores
C	SC	Start of commitment cycle	Ignores	Ignores
F	AY	Journaled changes applied	Ends	Ends
F	CE	Change end of data	Member's end of data changed	Ends
F	CL	Member closed	Ignores	Ignores
F	CR	Member cleared	Member cleared of all records	Ends
F	EJ	End journaling	Ends	Ignores
F	EP	End journaling access paths	Ignores	Ignores
F	FD	Member forced to auxiliary storage	Ignores	Ignores
F	IU	Object synchronized (see note)	Ignores	Ignores
F	IU	Object not synchronized (see note)	Ends	Ends
F	IZ	Member initialized	Initialized records inserted in member	Initialized records deleted from member
F	JM	Start journaling member	Ignores	Ends
F	JP	Start journaling access paths	Ignores	Ignores
F	MD	Member deleted	Ends	Ends
F	MF	Member saved with storage freed	Ends	Ends
F	MM	Member moved	Ignores	Ignores
F	MN	Member renamed	Ignores	Ignores
F	MR	Member restored	Ends	Ends
F	MS	Member saved	Ignores	Ignores
F	OP	Member opened	Ignores	Ignores
F	PD	Access path deleted	Ignores	Ignores

Table 13-1 (Page 2 of 4). Actions by Journal Code and Entry Type

Journal Code	Entry Type	Operation	APYJRNCHG	RMVJRNCHG
F	PM	Logical owning member of access path moved	Ignores	Ignores
F	PN	Logical owning member of access path renamed	Ignores	Ignores
F	RC	Journaled changes removed	Ends	Ends
F	RG	Member reorganized	Ends	Ends
F	SA	Start of APYJRNCHG	Ends	Ends
F	SR	Start of RMVJRNCHG	Ends	Ends
J	IA	Abnormal IPL	Ignores	Ignores
J	IN	Normal IPL	Ignores	Ignores
J	NR	Receiver attached	Ignores	Ignores
J	PR	Receiver detached	Ignores	Ignores
J	RD	Receiver deleted	Ignores	Ignores
J	RF	Receiver saved, storage freed	Ignores	Ignores
J	RR	Receiver restored	Ignores	Ignores
J	RS	Receiver saved	Ignores	Ignores
P	TP	Performance shared pool change	Ignores	Ignores
R	BR	Before-image updated for rollback operation	Ignores	Record updated with before-image
R	DL	Record deleted	Record deleted	Record updated with before-image
R	DR	Record deleted for rollback operation	Record deleted	Record updated
R	PT	Record written to member	Record written to member	Record deleted from member
R	PX	Record added directly to member	Record added	Record deleted from member
R	UB	Record updated (before-image)	Ignores	Record updated with before-image
R	UP	Record updated (after-image)	Record updated with after-image	Ignores
R	UR	After-image updated for rollback operation	Record updated with after-image	Ignores
S	CF	Next system table changes	Ignores	Ignores
S	ER	Error detected by SNADS process	Ignores	Ignores

Table 13-1 (Page 3 of 4). Actions by Journal Code and Entry Type

Journal Code	Entry Type	Operation	APYJRNCHG	RMVJRNCHG
S	LG	SNADS operation performed successfully	Ignores	Ignores
S	RT	Routing table changed	Ignores	Ignores
S	XE	DSNX error entry	Ignores	Ignores
S	XL	DSNX logging entry	Ignores	Ignores
T	AF	All authority failures	Ignores	Ignores
T	PS	Profile swap	Ignores	Ignores
T	PW	Passwords used that are not valid	Ignores	Ignores
T	RP	Restore of programs that adopt their owner's authority	Ignores	Ignores
T	RJ	Restore of job descriptions that contain user profile names	Ignores	Ignores
T	RO	Restore of objects when ownership information changes	Ignores	Ignores
T	RA	Restore of objects when authority changes	Ignores	Ignores
T	RU	Restore of authority for user profiles		
T	CA	Changes to object authority (authorization list or object)	Ignores	Ignores
T	CP	Create, change, delete, display, restore of user profiles	Ignores	Ignores
T	DS	DST security officer password reset	Ignores	Ignores
T	OW	Changes to object ownership	Ignores	Ignores
T	PA	Changes to programs (CHGPGM) that will now adopt the owner's authority	Ignores	Ignores
T	SV	Changes to system values	Ignores	Ignores
T	NA	Changes to network attributes	Ignores	Ignores
T	SE	Changes to subsystem routing	Ignores	Ignores

Table 13-1 (Page 4 of 4). Actions by Journal Code and Entry Type

Journal Code	Entry Type	Operation	APYJRNCHG	RMVJRNCHG
T	DO	All delete operations on the system	Ignores	Ignores
T	JD	Changes to the USER parameter of a job description	Ignores	Ignores
U	User-specified	User entry	Ignores	Ignores

Note: The *Flag* field in the journal entry indicates whether the file is synchronized (hex F0 = file was synchronized; hex F1 = file was not synchronized).

In addition to the entries that cause the command to end, the system ends the Apply Journal Changes (APYJRNCHG) or Remove Journal Changes (RMVJRNCHG) command if any format error (such as an undefined entry for that file member) or logical error (such as updating a record that has not been inserted) is encountered when the command is run.

For entries that end the APYJRNCHG or RMVJRNCHG command, a message identifying the reason for the system end is placed in the job log, and the corresponding change is not made to the file member. The message contains the sequence number of the journal entry on which the failing condition was detected. Start applying the changes again by indicating a new starting point after you have analyzed the error, or create a CL program to analyze the error and start applying the changes again.

If the entry that caused the APYJRNCHG command to end is entry code F of type RG, you may want to reorganize the physical file member referred to in the journal entry. Use the same options that were originally specified when the journal entry was recorded in the journal receiver.

The APYJRNCHG and RMVJRNCHG commands send an escape message and end the journal operation if any required journal receiver defined by the RCVRNG parameter is not on the system and associated with the journal. Use the WRKJRNA command to select the journal receiver display, to see which journal receivers are on the system and associated with the journal. The escape message contains the name of the required journal receiver if the reason code of message CPF7053 is 1 or if message CPF9801 is issued.

When the processing of the APYJRNCHG or RMVJRNCHG command ends with an escape message, the members can be partially changed. To determine how many changes were applied or removed for each member, use the DSPJRN command to display the journal entries with an F journal code and an entry type of AY or RC. The *Count* field in the journal entry contains the number of journal entries applied or removed.

For more information on the journal codes and the journal entries, see "System-Created Journal Entries" on page 12-17.

Applying Journal Changes during a Total System Restore Operation

If you are using journaling and need to later apply journaled changes, continue with the following steps.

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily save operations using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform the steps in "Working with Journals" for the system supplied journal QUSRSYS/QAOSDIAJRN. If you specified OBJJRN(*YES) on the SAVCHGOBJ command, you do not need to apply journal changes.

Working with Journals

1. Type the following and press the Enter key:

   ```
   WRKJRN
   ```

2. The Specify Journal Name display is shown. Specify *ALL for the *Library name* prompt and press the Enter key.

3. The Work with Journals display is shown. Type a **5** in the *Opt* field for each journal that you want to apply journaled changes to; this will display the name of the currently attached journal receiver. Write down the name of the journal and journal receiver.

   Note: If you are using OfficeVision/400 or PC Support/400, you must apply journaled changes to the system supplied journal QUSRSYS/QAOSDIAJRN.

4. You cannot restore journal receivers from the SAVCHGOBJ media if active journal receivers are attached. To later apply all journaled changes that have occurred since the last complete save operation, you must restore the receivers to the system from the SAVCHGOBJ media.

   For each journal identified in the previous step, do the following steps:

   a. Create a journal receiver that will be used as a temporary receiver. Give it a name that will identify it as a temporary receiver, for example, TEMPnn. You can enter a description in the text (TEXT parameter) that identifies it as a temporary receiver for disaster recovery.

      ```
      CRTJRNRCV JRNRCV(library-name/TEMPnn)
            TEXT('temporary journal receiver for journal xxx')
      ```

   b. To detach the current receiver and attach the new TEMPnn receiver, type the following and press the Enter key.

      ```
      CHGJRN JRN(library-name/journal-name) JRNRCV(library-name/TEMPnn)
      ```

   c. Delete the detached journal receiver (identified in the step where you wrote down the name of the journal and journal receiver) using the Delete Journal Receiver (DLTJRNRCV) command. (This allows the journal receivers on the SAVCHGOBJ media to be restored successfully.)

      ```
      DLTJRNRCV JRNRCV(library-name/journal-receiver)
      ```

      If you receive message CPA7025 *Receiver never fully saved*, enter an **I** to ignore and press Enter to continue the delete.

Restoring Changed Objects

Load the SAVCHGOBJ tape and enter the following to restore changed objects.

```
RSTOBJ OBJ(*ALL) DEV(tape-device) SAVLIB(library-name)
       OBJTYPE(*ALL) ENDOPT(*LEAVE) MBROPT(*ALL) ALWOBJDIF(*ALL)
```

You must repeat this step for every library saved using the SAVCHGOBJ command. If you specified SAVCHGOBJ LIB(*ALLUSR), type the following to determine the libraries that were saved:

```
DSPTAP *SAVRST
```

If you are using journaling, perform the steps in "Applying Journal Changes" for each journal you wish to apply journal changes to. Otherwise, continue with "Restoring Changed Documents and Folders."

Applying Journal Changes

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily saves using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform this step for the system supplied journal QUSRSYS/QAOSDIAJRN.

1. Determine the name of the last journal receiver (the last receiver restored) by entering the following:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

2. Press F15 (Work with receiver directory) to display the last receiver with a status of SAVED or PARTIAL. Write down the name of the receiver.

3. Create a receiver that follows the same naming convention as the last receiver but assign it a number of one greater.

   ```
   CRTJRNRCV JRNRCV(library-name/journal-receiver-nameNN)
   ```

   By doing this, you are doing what the CHGJRN command would normally do if the receiver was the current receiver being detached with a new receiver name being created. This allows your naming convention for journal receivers to continue.

4. Use the CHGJRN command to detach the temporary receiver and attach the new receiver you just created.

   ```
   CHGJRN JRN(library-name/journal-name)
        JRNRCV(library-name/journal-receiver-nameNN)
   ```

5. Determine the chain of receivers to be used in the APYJRNCHG command by entering the following command:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

   Press F15 (Work with receiver directory) to display the receivers. Write down the first and last receiver that you restored (last receiver is prior to the TEMPnn receiver). Notice that the first and last is the same receiver if only one journal receiver was restored.

6. When applying journal changes and the ending receiver has a status of PARTIAL (saved while attached), the TOENT parameter requires that a sequence number be specified on the APYJRNCHG command. Determine the last entry to be applied by entering the following command for the last receiver (identified in previous step):

   ```
   DSPJRNRCVA JRNRCV(library-name/last-journal-receiver-name)
   ```

Write down the value for the *Last Sequence Number* field.

7. Enter the following command to apply the journaled changes using the first and last journal receivers identified on the Work with Receiver Directory display.

```
APYJRNCHG JRN(library-name/journal-name)
          FILE((library-name/*ALL))
          RCVRNG(lib-name/first-receiver lib-name/last-receiver)
          FROMENT(*LASTSAVE) TOENT(last-entry)
```

Attention

Do not apply journal changes to the document and folder search index database files (QAOSSH10 through QAOSSH19) for journal QAOSDIAJRN in library QUSRSYS. You must specify individual files on the FILE parameter instead of *ALL.

```
APYJRNCHG JRN(QUSRSYS/QAOSDIAJRN)
          FILE((QUSRSYS/QAOSSH10) (QUSRSYS/QAOKLY02)
               (QUSRSYS/QAOSAH07) (QUSRSYS/QAOKLY03)
               (QUSRSYS/QAOKDYX4) (QUSRSYS/QAOKNY06)
               (QUSRSYS/QAOKDY01) (QUSRSYS/QAOKDY04)
               (QUSRSYS/QAOKDY05) (QUSRSYS/QAOKDY07)
               (QUSRSYS/QAOKDY08) (QUSRSYS/QAOKDY09))
          RCVRNG(lib-name/first-receiver lib-name/last-receiver)
          FROMENT(*LASTSAVE) TOENT(last-entry)
```

Restoring Changed Documents and Folders

If you performed daily save operations for documents and folders, do the following steps. Otherwise, continue with the RSTAUT command.

1. Load the last daily SAVDLO tape and run the following two commands.

2. If you performed daily save operations to backup all new folders and new and changed documents since the last complete SAVDLO operation, type the following and press the Enter key.

   ```
   RSTDLO DLO(*ALL) DEV(TAP01) SAVFLR(*ANY) ALWOBJDIF(*ALL)
   ```

   If you saved all documents referred to by the mail using SAVDLO DLO(*MAIL), type the following and press the Enter key.

   ```
   RSTDLO DLO(*MAIL) DEV(TAP01)
   ```

3. Restore users' authority by entering:

   ```
   RSTAUT
   ```

   The time it takes for the RSTAUT command to complete can vary significantly. The time depends on the number of user profiles and private authorities that were saved during the save operation.

4. This completes the restore operation.

   Power down the system by entering the following:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES)
   ```

   to perform a normal IPL and return the system to normal operations.

   Note: If you turned auto configuration off during the restore of the operating system, you must perform an IPL with the keylock switch in the **Normal** position.

Displaying and Printing Journal Entries

Use the Display Journal (DSPJRN) command to display journal entries. These entries can be displayed at a work station, printed, or written to a database file. (You cannot directly access the journal entries in the form in which they are contained in the journal receivers.)

You can use the list of the journal entries to:

- Prepare for the recovery of a particular file. The list contains the information you need to specify the starting and ending points for applying and removing journaled changes:
  - The functions that have been performed on the physical files being journaled (such as save and restore, clear, reorganize)
  - The functions that have been performed on the journal (such as attaching new journal receivers)
  - The functions that have been performed on the associated journal receivers (such as save and restore)
- Review the activity that has occurred on a file.
- Analyze journal entries for debugging or problem analysis.
- Provide a report of a saved journal receiver.
- List the events that have occurred in the functioning of the journal (for example, when the files have been saved).

The DSPJRN command lists journal entries for a particular member of a file, or the entries for all files within a particular library. You can further identify journal entries to be listed by specifying only:

- User-created entries
- Journal entries created to control the journal (journal codes F and J)
- Journal entries for specific entry types or journal codes
- Journal entries for a particular job, program, or file
- Any combination of these

Output for Journal Entries Directed to a Work Station

If you direct the output from the DSPJRN command to the requesting work station, basic information about the journal entries appears. Use the roll key to display the next sequential set of entries. When TOENT(*LAST) is specified on the command and the last journal entries in the journal are displayed, press the Page Down key to display any new journal entries put in the journal since the last time the Page Down key was pressed.

Output for Journal Entries Directed to a Database File

If you direct the output from the DSPJRN command to a database output file, you can further restrict the journal entries you want to process by creating logical files over the database output file.

Each journal entry occupies one record in the output file. Each has a fixed length portion for standard files. Before-images and after-images occupy separate records. The ENTDTALEN parameter controls the length of the field used to contain the record image. If the journal entry is smaller than the output file record, the journal entry is padded with blanks. If the journal entry is larger than the output file record, the remainder of the journal entry is truncated, and the system issues a warning message. To avoid truncation, specify the maximum record length in your files for the ENTDTALEN parameter on the DSPJRN command.

If you write journal entries to a database output file, you can write applications programs that will process the data to:

- Write your own apply program. For an example, see the APYUSRCHG tool in library QUSRTOOL.
- Summarize the entries so there is only one single entry for each relative record. See the example SUMJRNENT tool in library QUSRTOOL.
- Correct data that has been incorrectly updated.
- Remove or review all changes made by a particular program.

  For example, if you remove all changes made by a particular program, you could remove some valid updates. For example, assume that two work station users are using the same program to update a file, and one user enters some invalid data. If you remove all changes made by that program to remove the invalid data, you also remove the valid data entered by the other work station user.

Format of Database Output Files

When you direct the output of the DSPJRN command to a database file, the system creates the converted journal entries in a standard format. The system creates the database file in one of two standard formats determined by the value specified for the OUTFILFMT parameter:

Type	Format
*TYPE1	The system uses the QJORDJE format in the model output file QADSPJRN to create the output file for the converted journal entries. These entries will not include the user profile and system name fields.
*TYPE2	The system uses the QJORDJE2 format in the model output file QADSPJR2 to create the output file for the converted journal entries. These entries will include the user profile and system name fields.

For V2R1.1

Type	Format
*TYPE3	The system uses the QJORDJE3 format in the model output file QADSPJR3 to create the output file for the converted journal entries. These entries will include the *Null Value Indicators* field.

You can create an output file to hold the output from the DSPJRN command but the format has to match the format of one of the IBM-supplied output files. See the DSPJRN command in the manual *Programming: Control Language Reference*, SC41-0030, for a list of the IBM-supplied output files.

Note: The length of the individual fields in the system-created output file is defined by the ENTDTALEN parameter on the DSPJRN command. The default length is 100 characters.

To process individual fields in the entry-specific data, you can use your high-level language (HLL) to subdivide the fields into subfields, or you can use the Retrieve Journal Entry (RTVJRNE) command and the substring built-in function.

Analyzing Your Journal Activity

You can use the output file created by the DSPJRN command to help analyze your journal entries. For example, you could determine how many of each type of entry (such as add or update) was done for a specific file or by a specific user.

An example of this type of analysis program is shown in the PRTJRNANL tool in the QUSRTOOL library.

Example of Transaction Logging File

A **transaction logging file** is used to start an application again after a system or job failure when a notify object is not used. A transaction logging file is often used in interactive applications to summarize the effects of a transaction.

For example, in an order entry application, a record is usually written to a transaction logging file for each item ordered. The record contains the item ordered, the quantity, and the price. In an accounts payable application, a record is written to a transaction logging file for each account number that is to receive a charge. This record normally contains such information as the account number, the amount charged, and the vendor.

In many of the applications where a transaction logging file already exists, a work station user can request information about the last transaction entered. By adding commit control to the applications in which a transaction logging file already exists, you can:

- Ensure that the database files are updated to a commitment boundary.
- Simplify the starting of the transaction again.

You must be able to uniquely identify the work station user if you use a transaction logging file for starting applications again under commit control. If unique user profile names are used on the system, that profile name can be placed in a field in the transaction logging record. This field can be used as the key to the file.

The following examples (Figure 13-2 through 13-6) assume that an order inventory file is being used to issue transactions and that a transaction logging file already exists. The program does the following:

1. Prompts the work station user for a quantity and item number.
2. Updates the quantity in the production master file (PRDMSTP).
3. Writes a record to the transaction logging file (ISSLOGL).

If the inventory quantity on hand is insufficient, the program rejects the transaction. The work station user can ask the program where the data entry was interrupted, since the item number, description, quantity, user name, and date are written to the transaction logging file.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7

  1.00      A          R PRDMSTR                   TEXT('Master record')
  2.00      A            PRODCT         3          COLHDG('Product' 'Number')
  3.00      A            DESCRP        20          COLHDG('Description')
  4.00      A            ONHAND         5  0       COLHDG('On Hand' 'Amount')
  5.00      A                                      EDTCDE(Z)
  6.00      A          K PRODCT
```

RV2W358-0

Figure 13-2. DDS for Physical File PRDMSTP

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7

  1.00      A          R ISSLOGR                   TEXT('Product log record')
  2.00      A            PRODCT         3          COLHDG('Product' 'Number')
  3.00      A            DESCRP        20          COLHDG('Description')
  4.00      A            QTY            3  0       COLHDG('Quantity')
  5.00      A                                      EDTCDE(Z)
  6.00      A            USER          10          COLHDG('User' 'Name')
  7.00      A            DATE           6  0       EDTCDE(Y)
  8.00      A                                      COLHDG('Date')
```

RV2W359-0

Figure 13-3. DDS for Physical File ISSLOGP Used by ISSLOGP

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7

  1.00      A                                      LIFO
  2.00      A          R ISSLOGR                   PFILE(ISSLOGP)
  3.00      A          K USER
```

RV2W360-0

Figure 13-4. DDS for Logical File ISSLOGL

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..
     A                                      REF(ISSLOGP)
     A          R PROMPT
     A                                      CA03(98 'End of program')
     A                                      CA02(97 'Where am I')
     A                                  1 20'ISSUES PROCESSING'
     A                                  3  2'Quantity'
     A            QTY       R        I   +1
     A  62                                  ERRMSG('Not enough +
     A                                      Qty' 62)
     A                                     +6'Product'
     A            PRODCT    R        I   +1
     A  61                                  ERRMSG('No Product +
     A                                      record found' 62)
     A  55                             15  2'No Previous record exists'
     A                                 24  2'CF2 Last transaction'
     A          R RESTART
     A                                  1 20'LAST TRANSACTION +
     A                                      INFORMATION'
     A                                  5  2'Product'
     A            PRODCT    R             +1
     A                                  7  2'Description'
     A            DESCRP    R             +1
     A                                  9  2'Qty'
     A            QTY       R             +1REFFLD(QTY)
```

RSLS848-1

Figure 13-5. DDS for Display File PRDISSD Used in the Program

This process is outlined in Figure 13-6 on page 13-32.

Example of Transaction Logging File

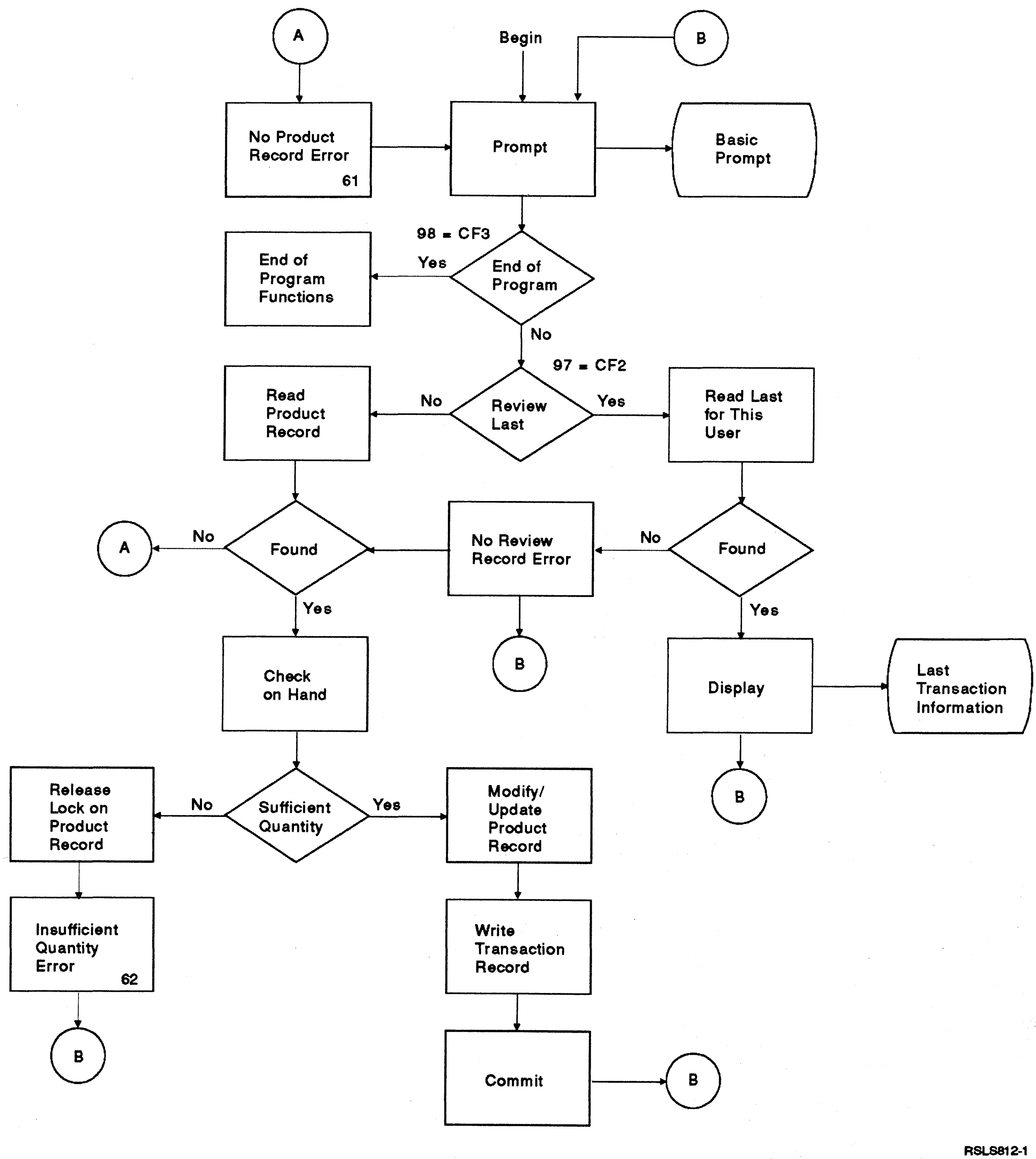


Figure 13-6. Program Flow

The RPG COMIT operation code is specified after the PRDMSTP file is updated and the record is written to the transaction logging file. Since each prompt to the operator represents a boundary for a new transaction, the transaction is considered a single Enter transaction.

The user name is passed to the program when it is called. The access path for the transaction logging file is defined in last-in-first-out (LIFO) sequence so the program can easily access the last record entered.

The work station user can start the program again after a system or job failure by using the same function that identified where data entry was stopped. No additional code needs to be added to the program. If you are currently using a transaction logging file but are not using it to find out where you are, add the user name to the transaction logging file (assuming user names are unique) and use this approach in the program.

Figure 13-7 on page 13-34 shows the RPG program used. Statements required for commit control are shown in boxes.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

     FPRDMSTP UF  E           K        DISK           KCOMIT
     FISSLOGL IF  E           K        DISK           KCOMIT        A
     FPRDISSD CF  E                    WORKSTN
     C           *ENTRY    PLIST
     C                     PARM           USER   10
     C*
     C*  Initialize fields used in Trans Log Rcd
     C*
     C                     MOVE UDATE     DATE
     C*
     C*  Basic processing loop
     C*
     C           LOOP      TAG
     C                     EXFMTPROMPT
     C   98                GOTO END                        End of pgm
     C   97                DO                              Where am I
     C                     EXSR WHERE
     C                     GOTO LOOP
     C                     END
     C           PRODCT    CHAINPRDMSTR            61      Not found
     C   61                GOTO LOOP
     C           ONHAND    SUB QTY        TEST    50  62   Less than
     C   62                DO                              Not enough
     C                     EXCPTRLSMST                     Release lck
     C                     GOTO LOOP
     C                     END
     C*
     C*  Update master record and output the Transaction Log Record
     C*
     C                     Z-ADDTEST      ONHAND
     C                     UPDATPRDMSTR
     C                     WRITEISSLOGR
     C                     COMIT
     C                     GOTO LOOP
     C*
     C*  End of program processing
     C*
     C           END       TAG
     C                     SETON                   LR
     C*
     C*  WHERE subroutine for 'Where am I' requests
     C*
     C           WHERE     BEGSR
     C           USER      CHAINISSLOGL            55      Not found
     C  N55                EXFMTRESTART
     C                     ENDSR
     OPRDMSTR E                RLSMST
```

RSLS813-2

Figure 13-7. RPG Program

To use commit control in this program, a lock level of *CHG would normally be specified. The record is locked by the change until a commit operation is run. Note that if there is an insufficient quantity of inventory, the record is explicitly released. (If the record were not explicitly released in the program, it would be released when the next record is read for update from the file.)

In this example, there is no additional advantage to using the lock level *ALL. If *ALL were used, a rollback or commit operation would have to be used to release the record when an insufficient quantity existed.

Figure 13-8 is a CL program that calls the RPG program PRDISS. Note the use of STRCMTCTL/ENDCMTCTL commands. The unique user name is retrieved (RTVJOBA command) and passed to the program. The use of the MONMSG command to cause a rollback is described in "Standard Commit Processing Program" on page 13-44.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

  1.00        PGM
  2.00        DCL          &USER *CHAR LEN(10)
  3.00        STRCMTCTL    LCKLVL(*CHG)
  4.00        RTVJOBA      USER(&USER)
  5.00        CALL         PRDISS PARM(&USER)
  6.00        MONMSG       MSGID(RPG9001) EXEC(ROLLBACK)
  7.00        ENDCMTCTL
  8.00        ENDPGM
```

RSLS827-0

Figure 13-8. CL Program Used to Call RPG Program PRDISS

Starting Application Programs Using a Notify Object

When a program is started after an abnormal end, it can look for an entry in the notify object. If one exists, the program can start a transaction again. After the transaction has been started again, the notify object is cleared by the program to prevent it from starting the same transaction yet another time.

Following are ways you can use a notify object:

- If the commit ID is placed in a database file, query this file to determine where to start each application or work station job again.
- If the commit ID is placed in a message queue for a particular work station, a message can be sent to the work station users when they sign on to inform them of the last transaction committed.
- If the commit ID is placed in a database file that has a key or user name, the program can read this file when it is started. If a record exists in the file, start the program again. The program can send a message to the work station user identifying the last transaction committed. Any recovery is performed by the program. If a record existed in the database file, the program deletes that record at the end of the program.
- For a batch application, the commit ID can be placed in a data area that contains totals, switch settings, and other status information necessary to start the application again. When the application is started, it accesses the data area and verifies the values stored there. If the application ends normally, the data area is set up for the next run.
- For a batch application, the commit ID can be sent to a message queue. A program that is run when the application is started can retrieve the messages from the queue and start the programs again.

There are several techniques for starting your applications again depending on your application needs. In choosing the technique, consider the following:

- When there are multiple users of a program at the same time, a single data area cannot be used as the notify object because after an abnormal system end, the commit ID for each user would overlay each other in the data area.
- Your design for deleting information in the notify object should handle the situation when a failure occurs immediately following use of the information:
  - If information is deleted immediately, it would not exist if another failure occurs before processing the interrupted transaction.
  - The information in the notify object should not be deleted until the successful processing of the interrupted transaction. In this case, more than one entry will exist in the notify object if it is a database file or message queue.
  - The program should access the last record if there is more than one entry.
- A notify object cannot be used to provide the work station user with the last transaction committed because the notify object is updated only if a system or job failure occurs or if uncommitted changes exist at the normal end of a job.
- If information is displayed to the work station user, it must be meaningful. To accomplish this may require that the program translate codes kept in the notify object into information that will help the user start again.
- Information for starting again should be displayed if the work station user needs it. Additional logic in the program is required to prevent information from being displayed again when it is no longer meaningful.
- A single notify object and a standard processing program can provide a starting again function if the notify object is a database file. This standard processing program is called by the programs that require the ability to start again to minimize the changes to each individual program.

Using a Unique Notify Object for Each Program

Using a single, unique notify object for each program allows use of an externally described commit ID even though there may be multiple users of the same program. In the following example (Figure 13-9 on page 13-37 through Figure 13-12 on page 13-39), a database file is used as a notify object and it is used only by this program.

The program has two database files (PRDMSTP and PRDLOCP) that must be updated for receipts to inventory. The display file used by the program is named PRDRCTD. A database file, PRDRCTP, is used as the notify object. This notify object is defined to the program as a file and is also used as the definition of a data structure for the notify function.

The DDS for the physical file PRDMSTP is shown in Figure 13-2 on page 13-30.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7

     A          R PRDLOCR                  TEXT('Location record')
     A            PRODCT         3         COLHDG('Product' 'Number')
     A            LOCATN         6         COLHDG('Location')
     A            LOCAMT         5  0      COLHDG('Location' 'Amount')
     A                                     EDTCDE(Z)
     A          K PRODCT
     A          K LOCATN
```

RV2W361-0

Figure 13-9. DDS for Physical File PRDLOCP

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

     A                                     REF(PRDMSTP)
     A          R PROMPT
     A                                     CA03(98 'End of program')
     A                                     SETOFF(71 'RESTART')
     A                                 1 20'PRODUCT RECEIPTS'
     A                                 3  2'Quantity'
     A            QTY            3  0I    +1
     A                                    +6'Product'
     A            PRODCT    R       I     +1
     A  61                                 ERRMSG('No record +
     A                                     found in the +
     A                                     master file' 62)
     A                                    +6'Location'
     A            LOCATN    R       I     +1REFFLD(LOCATN PRDLOCP)
     A  62                                 ERRMSG('No record +
     A                                     found in the +
     A                                     location file' 62)
     A                                 9  2'Last Transaction'
     A  71                                +6'This is restart +
     A                                       information'
     A                                     DSPATR(HI BL)
     A                                12  2'Quantity'
     A                                12 12'Product'
     A                                12 23'Location'
     A                                12 35'Description'
     A            LSTPRD    R         14 15REFFLD(PRODCT)
     A            LSTLOC    R         14 26REFFLD(LOCATN *SRC)
     A            LSTQTY    R         14  5REFFLD(QTY *SRC)
     A                                     EDTCDE(Z)
     A            LSTDSC    R         14 35REFFLD(DESCRP)
```

RSLS849-0

Figure 13-10. DDS for Display File PRDRCTD

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

  1.00      A                                          LIFO
  2.00      A                                          REF(PRDMSTP)
  3.00      A          R PRDRCTR
  4.00      A            USER          10
  5.00      A            PRODCT    R
  6.00      A            DESCRP    R
  7.00      A            QTY            3  0
  8.00      A            LOCATN    R                   REFFLD(LOCATN PRDLOCP)
  9.00      A          K USER
                                                                      RSLS850-0
```

Figure 13-11. DDS for Notify Object and Externally Described Data Structure (PRDRCTP)

The program processes the notify object as follows:

- At the beginning, the program randomly processes the notify object and displays a record if it exists for the specific key:
  - If multiple records exist, the last record for this key is used because the PRDRCTP file is in LIFO sequence.
  - If no record exists, a transaction was not interrupted so it is not necessary to start again.
  - If the program fails before the first successful commit operation, it does not consider that starting again is required.
- The routine to clear the notify object occurs at the end of the program:
  - If there were multiple failures, the routine can handle deletion of multiple records in the notify object.
  - Although the system places the commit ID in a database file, the commit ID must be specified as a variable in the RPG program.
  - Because RPG allows a data structure to be externally described, a data structure is a convenient way of specifying the commit ID. In this example, the data structure uses the same external description that the database file used as the notify object.

The processing for this program prompts the user for a product number, a location, and a quantity:

- Two files must be updated:
  - Product master file (PRDMSTP)
  - Product location file (PRDLOCP)
- A record in each file must exist before either is updated.
- The program moves the input fields to corresponding last fields after each transaction is successfully entered. These last fields are displayed to the operator on each prompt as feedback for what was last entered.
- If information for starting again exists, it is moved to these last fields and a special message appears on the display.

This process is outlined in Figure 13-12 on page 13-39. The user name is passed to the program to provide a unique record in the notify object.

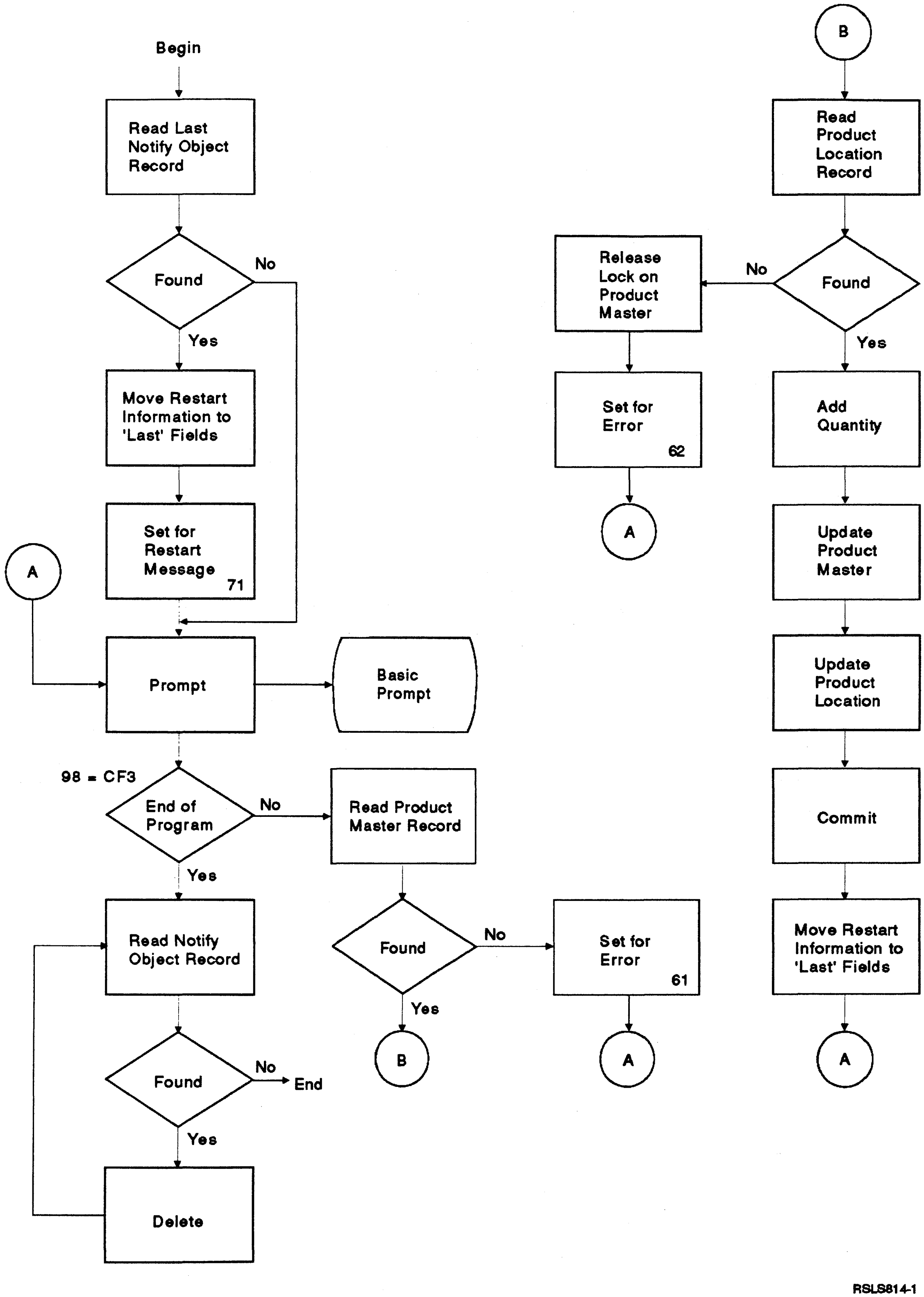


Figure 13-12. Program Flow

Figure 13-13 on page 13-40 shows the RPG source code for this example. The notify object (file PRDRCTP) is used as a normal file at the beginning and end of the program, and is also specified as the notify object in the CL (STRCMTCTL command) before calling the program.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

       FPRDMSTP UF  E           K        DISK         KCOMIT
       FPRDLOCP UF  E           K        DISK         KCOMIT
       FPRDRCTD CF  E                    WORKSTN
       F*
       F* The following file is the specific notify object for this pgm.
       F*      It is accessed only in a restart situation and at the
       F*        end of the program to delete any records.  The records
       F*        are written to the notify object by Commitment Control.
       F*
       FPRDRCTP UF  E           K        DISK
       ICMTID       E DSPRDRCTP
       C           *ENTRY    PLIST
       C                     PARM           USER10 10
       C                     MOVE USER10    USER
       C*
       C*  Check for restart information - get last rcd per user
       C*    PRDRCTP file access path is in LIFO sequence
       C*
       C           USER      CHAINPRDRCTR             20     Not found
       C   N20               DO                              Restart
       C                     EXSR MOVLST                     Move to last
       C                     SETON                    71     Restart
       C                     END
       C*
       C*  Basic processing loop
       C*
       C           LOOP      TAG
       C                     EXFMTPROMPT
       C    98               GOTO END                        End of pgm
       C           PRODCT    CHAINPRDMSTR             61     Not found
       C    61               GOTO LOOP
       C           KEY       KLIST
       C                     KFLD           PRODCT
       C                     KFLD           LOCATN
       C           KEY       CHAINPRDLOCR             62     Not found
       C    62               DO
       C                     EXCPTRLSMST                     Release lck
       C                     GOTO LOOP
       C                     END
       C                     ADD  QTY       ONHAND           Add
       C                     ADD  QTY       LOCAMT
       C                     UPDATPRDMSTR                    Update
       C                     UPDATPRDLOCR                    Update
       C*
       C*  Commit and move to previous fields
       C*
       C           CMTID     COMIT
       C                     EXSR MOVLST                     Move to last
       C                     GOTO LOOP
       C*
       C*  End of program processing
       C*
       C           END       TAG
       C                     SETON                    LR
       C*
       C*  Delete any records in the notify object
       C*
       C           DLTLP     TAG
       C           USER      CHAINPRDRCTR             20     Not found
       C   N20               DO
       C                     DELETPRDRCTR                    Delete
       C                     GOTO DLTLP
       C                     END
       C*
       C*  Move to -Last Used- fields for operator feedback
       C*
       C           MOVLST    BEGSR
       C                     MOVE PRODCT    LSTPRD
       C                     MOVE LOCATN    LSTLOC
       C                     MOVE QTY       LSTQTY
       C                     MOVE DESCRP    LSTDSC
       C                     ENDSR
       OPRDMSTR E                RLSMST
```

RSLW153-0

Figure 13-13. RPG Source

Using a Single Notify Object for All Programs

Using a single notify object for all programs is advantageous since all information required to start again is in the same object and a standard approach to the notify object can be used in all programs. In this situation, use a unique combination of user and program identifications to make sure that the program accesses the correct information when it starts again.

Because the information required to start again may vary from program to program, an externally described data structure for the commit ID should not be used. If a single notify object is used, the preceding program could describe the data structure within the program rather than externally. For example:

```
 1   10       USER
11   20       PGMNAM
21   23       PRODCT
24   29       LOCATN
30   49       DESC
50   51   0   QTY
52  220       DUMMY
```

In each program that uses this notify object, the information specified for the commit ID would be unique to the program (the user and program names are not unique). The notify object must be large enough to contain the maximum information that any program would place in the commit ID.

Using a Standard Processing Program

A standard processing program is one way to start your application again using one database file as the notify object for all applications. This approach assumes that user profile names are unique by user for all applications using the standard program.

For this approach, the physical file NFYOBJP is used as the notify object and defined as:

```
Unique user profile name   10 characters
Program identification     10 characters
Information for
     starting again        Character field
                           (This should be large
                           enough to contain the maximum
                           amount of information for starting
                           programs again that require
                           information for starting again.
                           This field is required by
                           the application programs.
                           In the example, it is
                           assumed to be a length of 200.)
```

The file is created with SHARE(*YES). The first two fields in the file are the key to the file. (This file can also be defined as a data structure in RPG programs.)

Processing Flow: The standard program is called from applications that must start again. The application programs pass this parameter list to the standard program:

- Request code
- Return code
- Data structure name (the contents of the notify object)

Request codes do the following:

- R (Read)

  Retrieves the last record added to the notify object with the same key. The return code is set as:

  0 No record is available (no start again required).

  1 Record returned in the information field for starting again (start again required).

- W (Write)

  Writes a record to the file. This code could be used if you use a notify object for your own purposes. For example, if the program determines that the transaction needs to be started again, the program could write a record to the notify object to simulate what the system will do if a job or the system fails.

- D (Delete)

  Deletes all records in the notify object with the same key. The return code is set as:

  0 No records exist to be deleted.

  1 One or more records were deleted.

- O (Open)

  The O request code is optional and is used to avoid having to start the processing program each time it is called.

- C (Close)

  After the open request code is used, using the close request code ensures the file is closed.

- S (Search)

  Returns the last record for this user. The program name is not used. This code can be used in an initial program to determine whether starting again is required.

Application Program Example: Following is an example of an application program that uses the standard program. The application shown in Figure 13-14 on page 13-43 performs as follows:

1. The application program receives the user name in a parameter and uses it with the program name as a unique identifier in the notify object.
2. The application program passes a request code of R to the standard commit processing program, which determines if a record exists in the notify object.
3. If the standard commit processing program returns a code of 1, a record was found and the application program presents the information needed to start again to the user.
4. The application program proceeds with normal processing.
5. When a transaction is completed, values are saved for reference so the work station user can see what was done for the previous transaction.

   The information saved is not provided by the notify object because the notify object is updated only if a job or system failure occurs.

```
*... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

     FPRDMSTP UF  E           K        DISK          KCOMIT
     FPRDLOCP UF  E           K        DISK          KCOMIT
     FPRDRCTD CF  E                    WORKSTN
     F*
     F* The following is a compile time array which contains the
     F*    restart information used in the next example
     F*
     E                    RTXT   50  50  1                Restart text
     I*
     I* Data structure used for info passed to notify object
     I*
     ICMTID       DS
     I                                        1  10 USER
     I                                       11  20 PGMNAM
     I                                       21  23 PRODCT
     I                                       24  29 LOCATN
     I                                       30  49 DESCRP
     I                                    P  50  510QTY
     I                                       52 170 DUMMY
     I                                      171 220 RSTART
     C           *ENTRY    PLIST
     C                     PARM           USER10 10
     C*
     C*  Initialize fields used to communicate with std program
     C*
     C                     MOVE USER10    USER
     C                     MOVEL'PRDRC2'  PGMNAM
     C                     MOVE 'R'       RQSCOD           Read Rqs
     C                CALL 'STDCMT'
     C                     PARM           RQSCOD  1
     C                     PARM           RTNCOD  1
     C                     PARM           CMTID 220        Data struct
     C           RTNCOD    IFEQ '1'                        Restart
     C                     EXSR MOVLST                     Move to last
     C SETON                                         71    Restart
     C                     END
     C*
     C*  Initialize fields used in notify object
     C*
     C                     MOVEARTXT,1    RSTART           Move text
     C*
     C* Basic processing loop
     C*
     C           LOOP      TAG
     C                     EXFMTPROMPT
     C   98                GOTO END
     C           PRODCT    CHAINPRDMSTR              61    Not found
     C   61                GOTO LOOP
     C           KEY       KLIST
     C                     KFLD           PRODCT
     C                     KFLD           LOCATN
```

RSLW154-0

Figure 13-14 (Part 1 of 2). Application Program Example

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

 52.00      C           KEY       CHAINPRDLOCR              62    Not found
 53.00      C   62                DO
 54.00      C                     EXCPTRLSMST                     Release lck
 55.00      C                     GOTO LOOP
 56.00      C                     END
 57.00      C                     ADD  QTY       ONHAND           Add
 58.00      C                     ADD  QTY       LOCAMT
 59.00      C                     UPDATPRDMSTR                    Update
 60.00      C                     UPDATPRDLOCR                    Update
 61.00      C*
 62.00      C*  Commit and move to previous fields
 63.00      C*
 64.00      C           CMTID     COMIT
 65.00      C                     EXSR MOVLST                     Move to last
 66.00      C                     GOTO LOOP
 67.00      C*  End of program processing
 68.00      C*
 69.00      C           END       TAG
 70.00      C                     MOVE 'D'       RQSCOD           Dlt Rqs
 71.00      C                     CALL 'STDCMT'
 72.00      C                     PARM           RQSCOD
 73.00      C                     PARM           RTNCOD
 74.00      C                     PARM           CMTID
 75.00      C                     SETON                     LR
 76.00      C*
 77.00      C*  Move to -Last Used- fields for operator feedback
 78.00      C*
 79.00      C           MOVLST    BEGSR
 80.00      C                     MOVE PRODCT    LSTPRD
 81.00      C                     MOVE LOCATN    LSTLOC
 82.00      C                     MOVE DESCRP    LSTDSC
 83.00      C                     MOVE QTY       LSTQTY
 84.00      C                     ENDSR
 85.00      OPRDMSTR E                RLSMST
 86.00 ** RTXT        Restart Text
 87.00 Inventory Menu - Receipts Option
```

RSLW155-0

Figure 13-14 (Part 2 of 2). Application Program Example

Standard Commit Processing Program: The standard commit (STDCMT) processing program performs the functions required to communicate with a single notify object used by all applications. While the commit control function automatically writes an entry to the notify object, a user-written standard program must process the notify object. The standard program simplifies and standardizes the approach.

The program is written to verify the parameters that were passed and perform the appropriate action as follows:

O=Open — The calling program requests the notify object file be kept open on return. Because the notify object is opened implicitly by the RPG program, the program should not close it. Indicator 98 is set so the program returns with LR off to keep the program's work areas and leaves the notify object open so it can be called again without excess overhead.

C=Close
: The calling program has determined it no longer needs the notify object and requests a close. Indicator 98 is set off to allow a full close of the notify object.

R=Read
: The calling program requests that a record with matching key fields be read and passed back. The program uses the passed key fields to attempt to retrieve a record from NFYOBJP. If duplicate records exist for the same key, the last record is returned. The return code is set accordingly and, if the record existed, it is passed back in the data structure CMTID.

W=Write
: The calling program requests a record to be written to the notify object to allow the calling program to start again the next time it is called. The program writes the contents of the passed data as a record in NFYOBJP.

D=Delete
: The calling program requests that records for this matching key be deleted. This function is usually performed at the successful completion of the calling program to remove any information on starting again. The program attempts to delete any records for passed key fields. If no records exist, a different return code is passed back.

S=Search
: The calling program requests a search for a record for a particular user regardless of which program wrote it. This function is used in the program for sign-on to indicate that starting again is required. The program uses only the user name as the key to see if records exist. The return code is set appropriately, and the contents of the last record for this key (if it exists) are read and passed back.

Figure 13-15 on page 13-46 shows the standard commit processing program, STDCMT.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

     FNFYOBJP UF  E           K        DISK                      A
     ICMTID       DS
     I                                        1   10 UNQUSR
     I                                       11   20 UNQPGM
     I                                       21 220 BIGFLD
     C           *ENTRY    PLIST
     C                     PARM           RQSCOD  1
     C                     PARM           RTNCOD  1
     C                     PARM           CMTID 220
     C           UNQUSR    CABEQ*BLANKS   BADEND         H1 Invalid
     C           UNQPGM    CABEQ*BLANKS   BADEND         H2 Invalid
     C*
     C*  '0' for Open
     C*
     C           RQSCOD    IFEQ '0'                         Open
     C                     SETON                     98    End LR
     C                     GOTO END
     C                     END
     C*
     C*  'C' for Close
     C*
     C           RQSCOD    IFEQ 'C'                         Close
     C                     SETOF                     98
     C                     GOTO END
     C                     END
     C*
     C*  'R' for Read - Get last record for the key
     C*
     C           RQSCOD    IFEQ 'R'                        Read
     C           KEY       KLIST
     C                     KFLD           UNQUSR
     C                     KFLD           UNQPGM
     C           KEY       CHAINNFYOBJR              51    Not found
     C   51                MOVE '0'       RTNCOD
     C   51                GOTO END
     C                     MOVE '1'       RTNCOD            Found
     C           LOOPI     TAG
     C           KEY       READENFYOBJR                  20 EOF
     C   20                GOTO END
     C                     GOTO LOOP1
     C                     END
     C*
     C*  'W' FOR Write
     C*
     C           RQSCOD    IFEQ 'W'                         Write
     C                     WRITENFYOBJR
     C                     GOTO END
     C                     END
     C*
     C*  'D' for Delete - Delete all records for the key
     C*
```

RSLW156-0

Figure 13-15 (Part 1 of 2). Standard Commit Processing Program

```
*... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..
C            RQSCOD    IFEQ 'D'                          Delete
C            KEY       CHAINNFYOBJR                51    Not found
C  51                  MOVE '0'      RTNCOD
C  51                  GOTO END
C                      MOVE '1'      RTNCOD              Found
C            LOOP2     TAG
C                      DELETNFYOBJR
C            KEY       READENFYOBJR                  20 EOF
C N20                  GOTO LOOP2
C                      GOTO END
C                      END
C*
C*  'S' for Search for the last record for this user
C*            (Ignore the -Program- portion of the key)
C*
C            RQSCOD    IFEQ 'S'                          Search
C            UNQUSR    SETLLNFYOBJR                  20 If equal
C N20                  MOVE '0'      RTNCOD
C N20                  GOTO END
C                      MOVE '1'      RTNCOD              Found
C            LOOP3     TAG
C            UNQUSR    READENFYOBJR                  20 EOF
C N20                  GOTO LOOP3
C                      GOTO END
C                      END
C*
C*  Invalid request code processing
C*
C                      SETON                       H2    Bad RQS code
C                      GOTO BADEND
C*
C*  End of program processing
C*
C            END       TAG
C N98                  SETON                       LR
C                      RETRN
C* BADEND tag is used then fall thru to RPG cycle error return
C            BADEND    TAG
```

RSLW157-0

Figure 13-15 (Part 2 of 2). Standard Commit Processing Program

Deciding If It Is Necessary to Start Again: The initial program can call the standard commit processing program to determine if it is necessary to start again. The work station user can than decide whether or not to start again.

The initial program passes a request code of S (search) to the standard program, which searches for any record for the user. If a record exists, the information for starting again is passed to the initial program and the information is displayed to the work station user.

The commit ID in the notify object should contain information that the initial program can display identifying what program needs to be started again. For example, the last 50 characters of the commit ID can be reserved to contain this information. In the application program, this information could be in a compile-time array and moved to the data structure in an initialization step. Figure 13-14 on page 13-43 shows how to include this in the application program.

Figure 13-16 is an example of an initial program that determines if a record exists in the notify object.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7

            PGM
            DCLF      CMTINLD
            DCL       &RQSCOD *CHAR LEN(1) VALUE(S)  /* Search */
            DCL       &RTNCOD *CHAR LEN(1)
            DCL       &CMTID *CHAR LEN(220)
            DCL       &USER *CHAR LEN(10)
            DCL       &INFO *CHAR LEN(50)
            RTVJOBA   USER(&USER)
            CHGVAR    &CMTID (&USER *CAT XX)
                      /* The XX is reqrd to prevent a blank Pgm nam */
            CALL      STDCMT PARM(&RQSCOD &RTNCOD &CMTID)
            IF        (&RTNCOD *EQ '1') DO /* RESTART REQD */
            CHGVAR    &INFO %SST(&CMTID 171 50)
            SNDRCVF   RCDFMT(RESTART)
            ENDDO
                      /*                                          */
                      /* Enter normal initial program statements */
                      /*    or -TFRCTL- to first menu program     */
                      /*                                          */
            ENDPGM
```

RV2W362-0

Figure 13-16. Initial Program Example

Commitment Control Practice Problem

This practice problem will assist you in understanding commitment control and its requirements. The following steps assume you are familiar with the OS/400 program and the data file utility (DFU), and have read this appendix. Before beginning this problem, do the following:

- Create a special library for this practice problem.
- Create source files and a job description.

Perform the following steps:

1. Create a physical file named ITMP (item master file). The data description specification (DDS) for this file is:

   ```
   A     R ITMR
   A       ITEM     2
   A       ONHAND   5   0
   A     K ITEM
   ```

2. Create a physical file named TRNP (transaction file). This file is used as a transaction log file. The DDS for this file is:

```
10    A    R TRNR
20    A      QTY     5  0
30    A      ITEM    2
40    A      USER    10
```

3. Create a logical file named TRNL (transaction logical). This file is used to assist in starting the application again. The *USER* field is the type LIFO sequence. The DDS for this file is:

```
10                           LIFO
20    A      R TRNR          PFILE (TRNP)
30    A      K USER
```

4. Enter the STRDFU command, and create a DFU application named ITMU for the ITMP file. Accept the defaults offered by DFU during the application definition.

5. Type the command CHGDTA ITMU and enter the following records for the ITMP file:

Item	**On Hand**
AA	450
BB	375
CC	4000

6. End the program using F3. This entry provides some data against which the program will operate.

7. Create the CL program Item Process (ITMPCSC) as follows:

```
PGM
DCL &USER *CHAR LEN(10)
RTVJOBA USER(&USER)
CALL ITMPCS PARM(&USER)
ENDPGM
```

   This is the control program that calls the ITMPCS program. It retrieves the user name and passes it to the processing program. This application assumes that unique user names are used.

8. Create a display file named ITMPCSD from the DDS shown in Figure 13-18 on page 13-55.

   There are two formats, the first for the basic prompt display and the second to allow the operator to review the last transaction entered. This display file is used by the ITMPCS program.

9. Study the logic flow provided in Figure 13-19 on page 13-56.

10. Enter the STRSEU command and type the source shown in Figure 13-17 on page 13-50.

```
 *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

 FITMP    UF  E           K        DISK
 F*                                               KCOMIT
 FTRNP    O   E                    DISK
 F*                                               KCOMIT
 FTRNL    IF  E           K        DISK
 F            TRNR                                KRENAMETRNR1
 FITMPCSD CF  E                    WORKSTN
 C* Enter parameter with User name for -TRNP- file
 C           *ENTRY    PLIST
 C                     PARM           USER   10
 C           LOOP      TAG
 C                     EXFMTPROMPT
 C* Check for CF3 for end of program
 C   93                DO                               End of Pgm
 C                     SETON                      LR
 C                     RETRN
 C                     END
 C* Check for CF4 for review last transaction
 C   94                DO                               Review last
 C* Check for existence of a record for this user in -TRNL- file
 C           USER      CHAINTRNR1                 64    Not found
 C   64                GOTO LOOP
 C                     EXFMTREVW
 C                     GOTO LOOP
 C                     END
 C* Access Item record
 C           ITEM      CHAINITMR                  62    Not found
 C* Handle -not found- Condition
 C   62                GOTO LOOP
 C* Does sufficient quantity exist
 C           ONHAND    SUB  QTY       TEST     50   61  Minus
 C* Handle insufficient quantity
 C   61                DO
 C* Release Item record which was locked by the CHAIN for update
 C                     EXCPTRLSITM
 C                     GOTO LOOP
 C                     END
 C* Change ONHAND and update the Item record
 C                     Z-ADDTEST      ONHAND
 C                     UPDATITMR
 C* Test for Special Simulation Conditions
 C           ITEM      IFEQ 'CC'
 C*     Simulate program need for rollback
 C           QTY       IFEQ 100
 C                     SETON                      63    Simult Rlbck
 C*                    ROLBK
 C                     GOTO LOOP
 C                     END
 C*     Simulate an abnormal program cancellation by Div by zero
 C*       Operator Should respond -C- to inquiry message
 C           QTY       IFEQ 101
 C                     Z-ADDO         ZERO   30
 C           TESTZ     DIV  ZERO      TESTZ  30         Msg occurs
 C                     END
 C*     Simulate an abnormal job cancellation by DSPLY.
 C*       Operator Should System Request to another job
 C*        and cancel this one with OPTION(*IMMED)
 C           QTY       IFEQ 102
 C           'CC=102'  DSPLY                            Msg occurs
 C                     END
 C                     END                              ITEM=CC
 C* Write the -TRNP- file
 C                     WRITETRNR
 C* Commit the update to -ITMP- and write to -TRNP-
 C*                    COMIT
 C                     GOTO LOOP
 OITMR    E                RLSITM
```

RSLW158-0

Figure 13-17. RPG Source Program for ITMPTCS

11. Enter the CRTRPGPGM command to create program ITMPCS from the source entered in the previous step.

12. Type the command CALL ITMPCSC, press Enter, and press CF4. A message should appear stating that there are no entries for this operator.
13. Enter the following data to see if the program operates correctly:

Quantity	Item
3	AA
4	BB

14. Press CF4. The review display should appear with the BB item last entered. Enter the following data:

Quantity	Item
5	FF (Invalid item number message should occur.)
9000	BB (Insufficient quantity error message should occur.)
100	CC (Rollback message should occur.)
102	CC (RPG DSPLY operation should occur. Press the Enter key.)
101	CC (The program should display an inquiry message stating that a divide-by-zero condition has occurred or end, depending on the setting of job attribute INQMSGRPY. If the inquiry message appears, enter C to cancel the RPG program and then C to cancel the CL program on the subsequent inquiry. This simulates an unexpected error condition.)

15. Type the Display Data command DSPDTA ITMP.

    See if the records AA and BB have been updated correctly. The values should be AA = 447, BB = 371, and CC = 3697. Note that the quantities subtracted from CC occurred, but the transaction records were not written.
16. Type the Create Journal Receiver (CRTJRNRCV) command and the parameter JRNRCV(library name/RCVR1) to create a journal receiver used for commit control. Files operating under the same commit control boundary must be journaled to the same journal.
17. Type the Create Journal (CRTJRN) command using the parameters JRN(library name/JRNTEST) JRNRCV(library name/RCVR1) to create a journal to be used for commit control.
18. Type the Start Journal Physical File (STRJRNPF) command using the parameters FILE(ITMP TRNP) JRN(JRNTEST) to journal the files to be used for commitment control.

    The IMAGES parameter uses a default of *AFTER, meaning only after-image changes of the records appear in the journal. The files ITMP and TRNP have now started journaling.

    This problem does not require the normal practice of saving files that are journaled.
19. Type the command CALL ITMPCSC and enter the following transactions:

Quantity	Item
5	AA
6	BB

    End the program by pressing CF3.
20. Type the Display Journal command DSPJRN JRNTEST

    Note the entries appearing in the journal. The same sequence of entries (UP = update of ITMP followed by PT = put of TRNP) occurs in the journal as was performed by the program. This is because a logical file is defined over the physical file TRNP and the system overrides the RPG default. If no logical file existed, the RPG assumption of SEQONLY(*YES) would be used, and a block of PT entries would appear because the records would be kept in the RPG buffer until the block is full.

21. Change the CL program ITMPCSC as follows (the new statements are shown with an asterisk):

```
      PGM
      DCL &USER *CHAR LEN(10)
      RTVJOBA USER(&USER)
*     STRCMTCTL LCKLVL(*CHG)
      CALL ITMPCS PARM(&USER)
*     MONMSG MSGID(RPG9001) EXEC(ROLLBACK)
*     ENDCMTCTL
      ENDPGM
```

    The STRCMTCTL command sets up the commit control environment. The LCKLVL word specifies that records read for update but not updated can be released during the transaction. The MONMSG command handles any RPG escape messages and performs a ROLLBACK in case the RPG program abnormally ends. The ENDCMTCTL command ends the commit control environment.

22. Delete the existing ITMPCSC program and create it again.
23. Change the RPG program to remove the comment symbols at statements 2.00, 4.00, 46.00, and 65.00. The source is now ready for use with commit control.
24. Delete the existing ITMPCS program and create it again. The program is now ready to operate under commit control.
25. Type the command CALL ITMPCSC and the following transactions:

Quantity	**Item**
7	AA
8	BB

26. Use System Request and request the option to display the current job. When the Display Job display appears, select option 16 to request the display of the commit control status.

    Note the values on the display. There should be two commits because two commit statements were run in the program.

27. Press CF9 to see a list of the files under commit control and the amount of activity for each file.
28. Return to the program and end it by pressing CF3.
29. Enter DSPJRN JRNTEST and note the entries for the files and the special journal entries for commit control:
    - BC: STRCMTCTL command occurred.
    - SC: Start commit cycle. This occurs whenever the first database operation in the transaction causes a record to be locked as part of commit control.
    - CM: Commit operation has occurred.
    - EC: ENDCMTCTL command occurred.

Note how the commit control before- and after-images (UB and UP types) automatically occur even though you had originally requested IMAGES(*AFTER) for the journal.

30. Type the command CALL ITMPCSC and the following transactions:

Quantity	Item
12	AA
100	CC (This is the condition to simulate the need for an application use of rollback. The CC record in the ITMP file, which was updated by RPG statement 40.00 is rolled back.)

31. Press CF4 to determine the last transaction entered.

    The last committed transaction is the entry for item AA.

32. Use System Request and request the Display Current Job option. When the Display Job display appears, request the display of the commit control status.

    Note the values on the display and how they have been changed by the rollback.

33. Return to the program.
34. Return to the basic prompt display and end the program by pressing CF3.
35. Type the command DSPJRN JRNTEST.

    Note the additional entries that appear in the journal for the use of the rollback entry (RB entry). When the ITMP record is rolled back, three entries are placed in the journal. This is because any change to the database file under commit control produces a before (BR) and after (UR) entry.

36. Display the UB, UP, BR, and UR records and use option 5 to display the full entries. Since the *Quantity* field is in packed decimal, use CF11 to request a hex display. Note the following:

    - The on-hand value of the ITMP record in the UB record
    - How the on-hand value is reduced by the UP record
    - How the BR record is the same as the UP record
    - How the UR record returns the value as originally displayed for the UB record

    The last entry is the RB entry for the end of the rollback.

37. Type the command CALL ITMPCSC, press Enter, and press CF4. Note the last transaction entered.
38. Type the following transactions:

Quantity	Item
13	AA
101	CC (This is the condition to simulate an unexpected error condition, which causes the program to end. The simulation occurs by dividing a field by 0. The program will display an inquiry message or end, depending on the setting of the job attribute INQMSGRPY. If the inquiry message appears, enter C to end the program. Because the CL program was changed to monitor for RPG program errors, the second inquiry which occurred does not occur.)

39. Type the command DSPJRN JRNTEST.

    The same type of rollback handling has occurred, but this time the rollback was caused by the EXEC parameter of the MONMSG command in the CL program instead of the RPG program. Display the two RB entries to see which program caused them.

40. Type the command WRKJOB and write down the fully qualified job name to be used later.

41. Type the command CALL ITMPCSC and enter the following transaction:

Quantity	**Item**
14	AA
102	CC (The RPG DSPLY operation should occur to the external message queue. Use the System Request key and select option 1 on the system request menu to transfer to a secondary job.)

42. Sign on to the second job and reestablish your environment.

43. Type the command ENDJOB and specify the fully qualified job name identified earlier and OPTION(*IMMED). This simulates an abnormal job or system end.

44. Wait about 30 seconds, type the command CALL ITMPCSC and press CF4.

    Note the last committed transaction. It should be the AA item entered earlier.

45. Return to the basic prompt display and end the program by pressing CF3.

46. Type the command DSPJRN JRNTEST.

    The same type of rollback handling has occurred, but this time the rollback was caused by the system instead of one of the programs. The RB entry was written by the program QWTPITPP, which is the work management abnormal end program.

You have now used the basic functions of commit control. You can proceed with commit control on your applications or try some of the other functions such as:

- Using a notify object
- Locking records that are only read with LCKLVL(*ALL)
- Locking multiple records in the same file with LCKLVL(*ALL)

Figure 13-18 on page 13-55 shows the DDS for the display file.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7 ..

        A            R PROMPT
        A                                          CA03(93 'End of program')
        A                                          CA04(94 'Review last')
        A                                          SETOFF(64 'No rcd to rvw')
        A                                       1  2'INVENTORY TRANSACTIONS'
        A                                       3  2'Quantity'
        A              QTY            5  0I       +1
        A  61                                      ERRMSG('Invalid +
        A                                          quantity' 61)
        A                                         +5'ITEM'
        A              ITEM           2   I       +1
        A  62                                      ERRMSG('Invalid +
        A                                          Item number' 62)
        A  63                                      ERRMSG('Rollback +
        A                                          occurred' 63)
        A  64                                  24  2'CF4 was pressed and +
        A                                          there are no +
        A                                          transactions for +
        A                                          this user'
        A                                          DSPATR(HI)
        A                                      23  2'CF4 Review last +
        A                                          transaction'
        A            R REVW
        A                                       1  2'INVENTORY TRANSACTIONS'
        A                                         +5'REVIEW LAST TRANSACTION'
        A                                       3  2'Quantity'
        A              QTY            5  0        +1EDTCDE(Z)
        A                                         +5'Item'
        A              ITEM           2           +1
```

RSLS826-1

Figure 13-18. DDS for the Display File

Figure 13-19 on page 13-56 illustrates the logic flow for the practice problem.

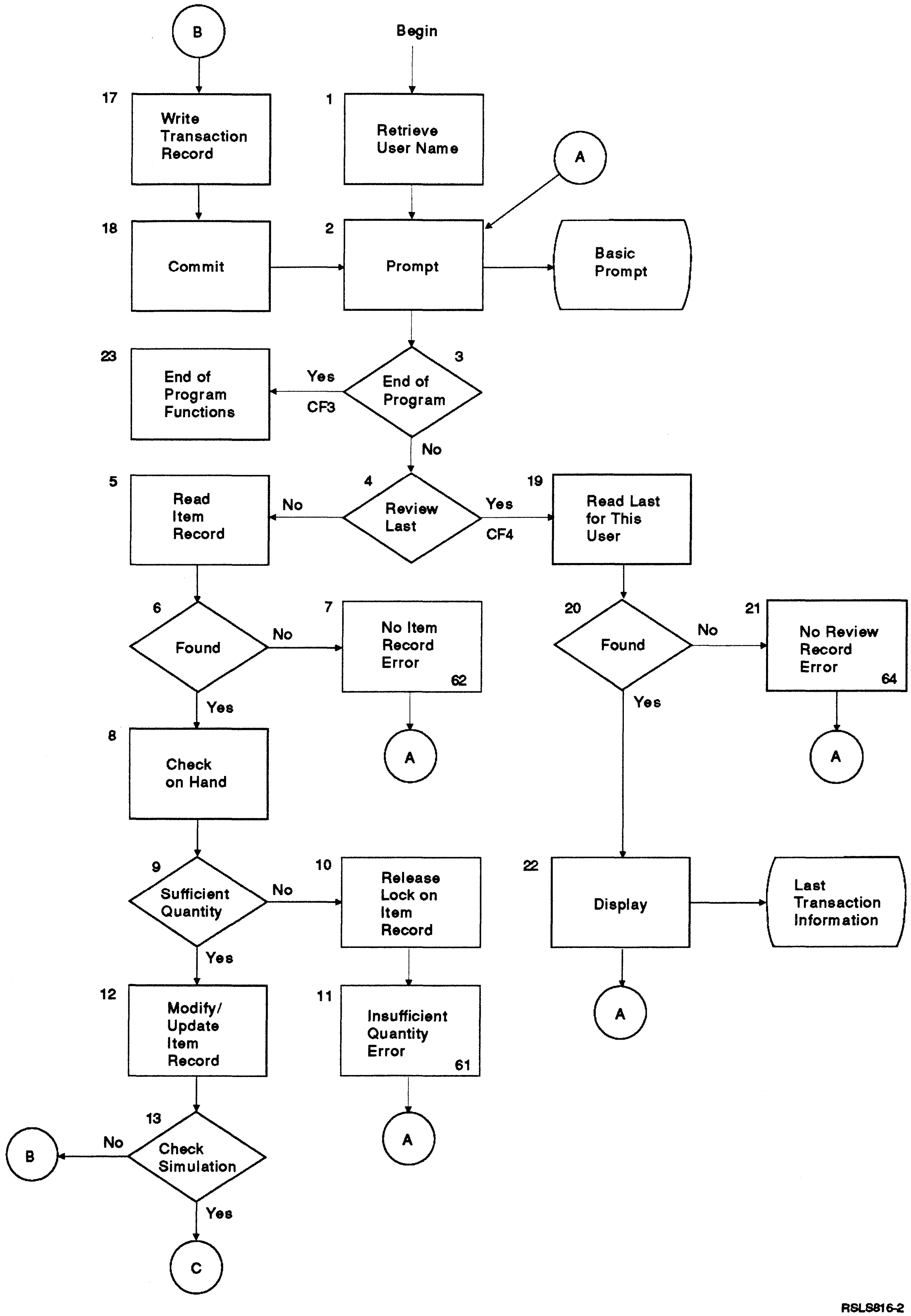


Figure 13-19 (Part 1 of 2). Logic Flow of the Practice Problem

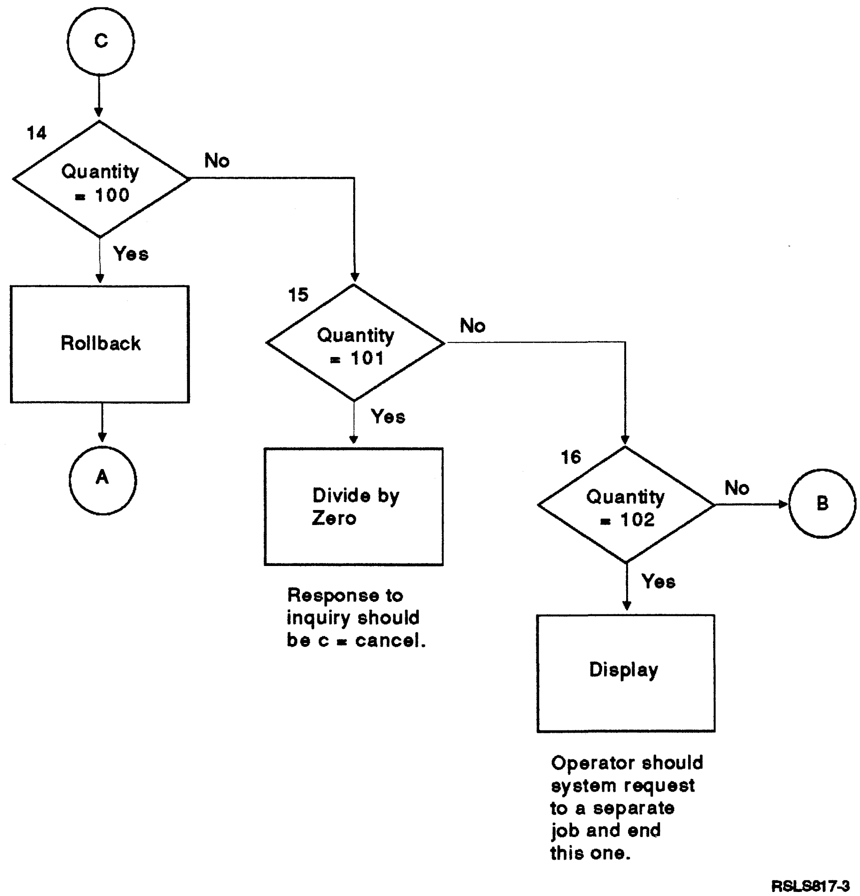


Figure 13-19 (Part 2 of 2). Logic Flow of the Practice Problem

Steps Associated with Logic Flow

The following steps are associated with the logic flow of the practice problem.

1. Retrieve the user name that is passed in as a parameter. This is used to write to the TRNP file and also used to retrieve the last transaction entered by each operator. This application assumes unique user names for operators.
2. Prompt for the basic display using the format name PROMPT.
3. If CF3 is pressed, start an end of program function.
4. If CF4 is pressed, start a routine to access the last transaction entered by the operator.
5. Read the item record using the field *ITEM*. Since the file is an update file, this request locks the record.
6. Check for a not found condition in the file ITMP.
7. If no ITMP record exists, set on indicator 62 to cause the error message and return to step 2.
8. Subtract the quantity requested (QTY) from the on hand balance (ONHAND) into a work area.
9. Check to see if sufficient quantity exists to meet the request.

10. If insufficient quantity exists, release the lock on the record in the ITMP file. This step is needed because of insufficient quantity.
11. Set on indicator 61 to signal an insufficient quantity display error message and return to step 2 on page 13-57.
12. Change the ONHAND field for the new balance and update the ITMR record.
13. Check for special entry in the ITEM field that can be used to simulate conditions where ROLLBACK is required.
14. Check for QTY=100. Issue a ROLLBACK operation. This simulates a condition where the program senses a need for rollback.
15. Check for QTY=101. Cause an exception in the program that will produce an inquiry message. Use divide by zero for this function. The operator should enter C to cancel the program unless the job description INQMSGRPH option provides an automatic reply. This simulates a condition where an unexpected error has occurred and the operator cancels the program.
16. Check for QTY=102. Issue a display with inquiry operation. This stops the program at this step and allows the use of the System Request key to get to a different job. Cancel the updating job. This simulates a condition where an abnormal job or system end has occurred in the middle of a commit boundary.
17. Write the transaction record to TRNP.
18. Commit the records for the transaction and return to step 2 on page 13-57.
19. Read the first record on the access path for file TRNL, using USER as the key. Since this file is in LIFO sequence, this will be the last transaction record entered by this user.
20. Check for a record not found condition in the TRNL file that would be caused if the file does not contain entries for this user.
21. If there is no record for this user, set on indicator 64 to cause an error message and return to step 2 on page 13-57.
22. Display the last transaction entered for this user. This information can be used if the operator forgets what was previously entered or when the transaction is restarted. When the operator responds, return to step 2 on page 13-57.
23. Perform any end of program functions.

Part 4. Auxiliary Storage Pools and Disk Recovery

Chapter 14. Auxiliary Storage Pools

The following topics provide general information about single-level storage, allocation of disk space, disk failure, and auxiliary storage pools (ASP).

Understanding Single-Level Storage

The concept of **single-level storage** is that, at a low level within the machine, a single virtual address space (virtual storage) exists. This storage is large enough to contain all data to be stored on the system. Functions operating above this low level see data as being stored in contiguously addressable locations in this space, no matter how the data may in fact be stored on auxiliary storage and in main storage.

The AS/400 machine interface (MI) instruction set provides an object-related interface. Space for an object is allocated as connected virtual storage. In reality, the data resides on auxiliary storage in disk extents that are not connected. Although programs see this data as being addressed directly in virtual storage, the data is brought into main storage, when needed, for use by programs run by the processing unit.

The internal machine functions that support the virtual address space involve three primary parts:

- **Auxiliary storage management** allocates and deallocates disk space for data placed in virtual storage.
- **Main storage management** copies data into main storage when it is needed, and then back to its permanent home on auxiliary storage after it has been updated.
- **Processing unit addressing** automatically addresses the appropriate location in main storage when a virtual address is used.

Refer to Figure 14-1 on page 14-2 for an illustration of single-level storage.

Recovery

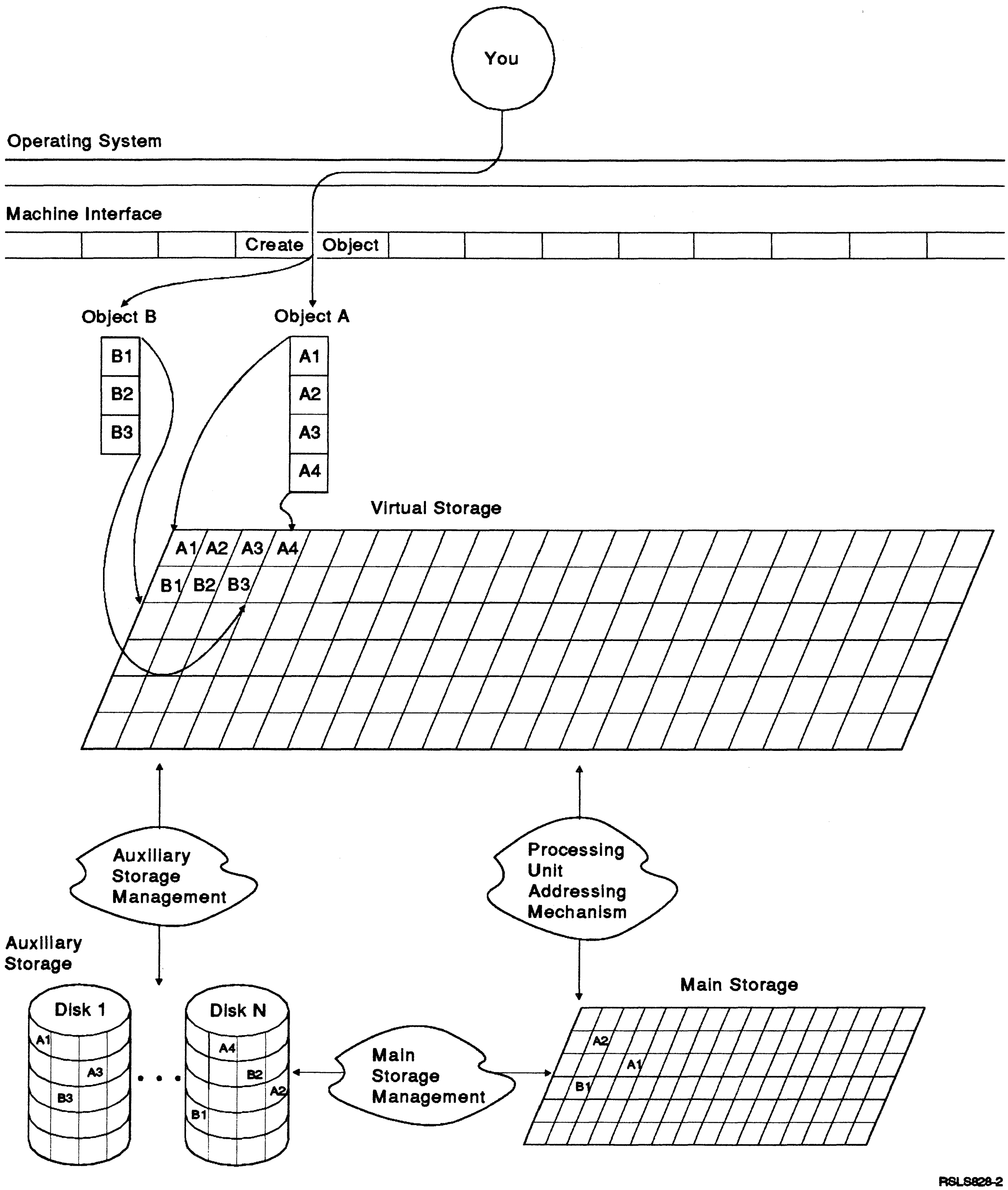


Figure 14-1. Single-Level Storage

Allocation of Space to Store Objects on Disk

When an object is created, you do not have to allocate space for the object. Allocation of space on auxiliary storage is done automatically and is determined by the space required, space available, and by space utilization of each disk attached to the system. To balance space utilization, auxiliary storage management typically allocates space for an object across two or more disks. If you had to perform the same management function, it would require an investment of time on a daily basis (several hours per week), and in all probability you could not do it as well.

As an example of allocation, Figure 14-1 on page 14-2 shows that object A, when created, was allocated into virtual storage units at connected locations A1, A2, A3, and A4. This unit of storage, referred to as a page, is determined by internal machine algorithms. These pages of virtual storage are then physically allocated to auxiliary storage to provide balanced use. The locations in auxiliary storage normally will not be connected. Similarly, locations selected in main storage for these pages of virtual storage (units A1 and A2 in the figure) are not connected.

Disk Failure with Data Loss

Having the system manage disk storage means you do not have to perform this time-consuming function, but it also means you do not know where data for objects resides on disk. Because you do not know this, there is no way for you to tell what was stored on a disk when it fails. Furthermore, because the system spreads the data for an object across many disks, it is likely that pieces of many objects were lost when a disk failed. No functions exist to tell you what piece of an object was lost or how to restore a piece of an object. Because you cannot always determine what was lost, your only recovery option is to reload the entire system from backup media.

If you have not saved all objects on offline media immediately prior to a failure, you will not be able to recover recently entered data. Therefore, when your previously saved objects are restored, the system is operational but the database is not current.

Even if journaling of database files is used, the receiver that contains the recently entered transactions will not be available if it too was lost when the disk failed. You will have to use alternatives to recover recently entered data, and in some cases that may not be possible. The recovery of recently entered data can complicate and lengthen the overall recovery process.

Considering these complications to recovering lost data, it is important to prepare for the possible need for recovery. You can simplify the recovery process by using disk recovery tools.

How Disk Units Are Attached to the System

In addition to understanding single-level storage, it is important that you understand how disk units are attached to the system before proceeding to the discussions of the disk recovery tools.

Different models of disk units are attached to the AS/400 system. The storage areas within the disk units are referred to as **storage units**. The number of storage units and the storage capacity per storage unit varies by disk unit type and model. Table 14-1 summarizes the characteristics of each disk device type and model.

Disk Unit Type	Model	Storage Units per Disk Unit	MB per Storage Unit	Total Storage Capacity (MB)
2800	001	2	320.18	640.36
6100	015	1	315.59	315.59
6102	010	1	320.18	320.18
6103	010/030	1	400.69	400.69
6105	010	1	320.18	320.18
6107	10/30	1	400.69	400.69
	20/40	2	400.69	801.38
9332	2xx	1	200.28	200.28
9332	4xx	2	200.28	400.57
9332	6xx	2	300.42	600.02
9335	B01	2	427.93	855.87
9336	010	2	471.2	942.4
9336	010	3	471.2	1413.6
9336	010	4	471.2	1884.8
9336	020	2	857.2	1714.4
9336	020	3	857.2	2571.6
9336	020	4	857.2	3428.8

Table 14-1. Storage Capacity by Disk Unit Type

Disks are assigned to an auxiliary storage pool (ASP) on a storage unit basis. The system treats each storage unit within a disk unit as a separate unit of auxiliary storage. When a new disk unit is attached to the system, the system initially treats each storage unit within it as nonconfigured storage units. Through dedicated service tool (DST) options you can allocate these storage units to either the system ASP or a user ASP of your choosing. When allocating nonconfigured storage units, use the serial number information assigned by the manufacturer to ensure you are selecting the correct physical device. Additionally, the individual storage units within the disk unit can be identified through the *Address* field on the DST Display Disk Configuration display. See Chapter 15, "Working with Auxiliary Storage Pools" for more information about displaying disk configuration.

When you allocate a nonconfigured storage unit to an ASP, the system assigns a number to the storage unit. The storage unit number can be used instead of the serial number and address.

When a storage unit has mirrored protection, two storage units (mirrored pair) are assigned the same unit number. The serial number and the address distinguish between the two units in a mirrored pair.

You may want to know which physical disk is being identified with each unit number. Make note of the unit number assignment to ensure that correct identification is made. If you need to verify the unit number assignment, use the DST Display Configuration Status display to show the serial numbers and addresses of each unit.

The storage unit addressed by the system as unit 1 is always used by the system to store licensed internal code. The amount of storage used on unit 1 is quite large and varies depending on the configuration of your system. Because unit 1 contains the initial programs and data used during an IPL of the system, it is also known as the **load source unit**.

The system reserves 1.08 megabytes of storage per storage unit (other than unit 1), reducing the amount of space available by that amount.

How the System Addresses Disk Units

The system processor requires information located in main storage. Main storage transfers information to disk (auxiliary storage pools (ASPs)).

Because ASPs are made of many disk units, other hardware is required to manage the transfer of data. Figure 14-2 illustrates the hardware.

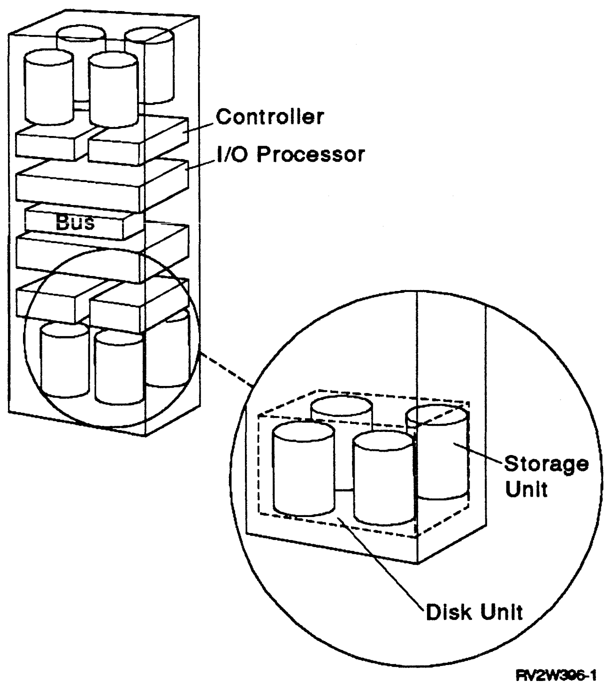


Figure 14-2. Hardware Used for Data Transfer

This hardware includes:

Bus: The bus is the main communications channel for input and output data transfer.

- All system models have at least one bus.
- Larger models (B50 and above) have two or three.
- The bus is addressed by the system as 0, 1, or 2.
- For 9406 Model D System Units:

Model	**Buses**
D30 and 40	Up to 2
D50	3
D60 and D70	5
D80	7

I/O processor: The I/O processor is attached to the bus and controls information between the bus and specific groups of I/O controllers.

- There are different types of I/O processors, depending on the input and output types to control.
- Disk storage uses the 6110, 6111, 6112, or 2615 I/O processor for the 9406 system unit.
- One 6110 or 6112 I/O processor can control up to:
  - Eight 9332 disk units
  - Eight 9335 B01 disk units
- One 6111 or 6112 I/O processor can control up to two 9336 disk units (four to eight storage units).
- 9332, 9335, and 9336 disk units cannot be mixed on the same I/O processor.
- I/O processor performance is directly related to the amount of input and output activity it controls.
- The I/O processor is addressed by the system as 0 through F.

Controller: The controller attaches to the I/O processor and handles the information transfer between the I/O processor and the disk units.

- Disk units have different types of controllers based on disk type.
- Disk unit 9332 has its own controller.
- The 9335, model A01, controller is used to control up to four 9335, model B01, disk units. Because of controller contention, performance requirements may dictate that the A01 controllers attach to no more than two B01 disk units.
- The controllers are addressed by the system as 00 through 07 on each bus.

Disk units: Disk units are the actual devices that contain the storage units.

- Disk units are further divided into independently addressed sections known as storage units.
- The number of storage units per disk unit vary by disk unit type and model. For example, the 9332, model 400, has two storage units.
- Storage units are addressable by the system. For example, the 9335 disk unit has 0 through 7, and the 9332 disk unit has 0 through 1.

How the System Addresses Individual Storage Units

For data transfer, the system needs a way to identify a single storage unit. Every hardware component (bus, I/O processor, controller, and storage unit) has a unique address.

The address has eight digits and is identified as follows:

```
X X X X - X X X X F F F F
└┘ |      └┘ └┘
 |  |      |   |
 |  |      |   +---- Storage Unit Number (00 through 07)
 |  |      +-------- Controller Number (00 through 07)
 |  +--------------- I/O Processor (0 through F)
 +------------------ Bus Number (00 through 07)
```

General Information about Auxiliary Storage Pools

An **auxiliary storage pool (ASP)** is a group of units defined from all the disk units that make up auxiliary storage. ASPs provide the means of isolating objects on a specific disk unit, or disk units, to prevent the loss of data due to a disk media failure on other disk units not included in the ASP.

If the system experiences a disk unit failure with data loss, recovery is required only for the libraries or objects in the ASP that contained the failed disk. System and user objects in other ASPs are protected from the disk failure.

In addition to the recovery advantage, placing libraries or objects in an ASP can improve performance because the system dedicates the disk units associated with that ASP to the objects in that ASP. In a heavy journaling environment, this can reduce contention between the journal receivers and the files if they are in different ASPs, and can improve journaling performance.

System ASP

The system ASP (ASP 1) is automatically created by the system and includes unit 1, as well as all other configured disks that are not assigned to a user ASP. Unlike a disk that is configured, a disk in nonconfigured status is attached to the system, but is not part of an ASP and is not being used. The system ASP contains all system objects for the OS/400 licensed program and all user objects not assigned to a user ASP.

If the system ASP fills to capacity, the system ends abnormally. You must IPL the system again and take the correct steps, such as deleting objects, to reduce the storage utilization within the system ASP.

You can specify a threshold that, when reached, warns the system operator of potential shortage of space.

For example, if you set the threshold value at 80 for the system ASP, the system operator (QSYSOPR) and system message queue (QSYSMSG) are notified when the system ASP is 80% full. This message is sent every hour until the value is changed, or until objects are deleted or transferred out of the system ASP. If this message is ignored and the system ASP fills to capacity, the system stops and you must perform an IPL of the system. The system may have trouble performing another IPL because objects were left in an unusable condition.

For a description of the details associated with the system ASP filling to capacity while checksum protection is in effect, refer to "Handling Protected Storage That Has Reached Maximum Storage Capacity" on page 18-48.

Notes:

1. For stage 1 hardware, main storage dump space must be contained on unit 1 in the system ASP.
2. For stage 2 hardware on 9406 Model D, the main store dump space must be contained on 2800 Model 001 storage units in the system ASP. Enough 2800 Model 001 storage units must be configured to contain the main storage dump space in the system ASP.
3. If the system contains (or is expected to contain) 224MB or more of main storage, then at least two 2800-001 storage units must be configured to the system ASP (ASP 1). If the system ASP has mirrored protection, then all four

2800-001 storage units must be configured to the system ASP. failure to provide enough disk units to contain the main storage dump space may result in significantly longer IPLs and will greatly limit problem support. Moving 2800-001 storage units from a user ASP to the system ASP requires you to save and then restore the user ASP.

4. If there is not enough free space to allocate to main store dump space, a message (CPI0987) will be sent to the QSYSOPR message queue and the QHST log.

User ASPs

A user auxiliary storage pool (ASP) is created by grouping together a physical set of disk units and assigning them to an ASP.

You can configure user ASPs 2 through 16. They can contain libraries and associated objects, or journals, journal receivers, and save files whose libraries are in the system ASP.

Isolating libraries and associated objects in a user ASP protects them from disk failures in other ASPs and reduces recovery time. See "Object Types Not Allowed in a User ASP" on page 14-11 for restrictions.

Journals and the files being journaled must be in the same ASP. However, the journal receivers should be placed in a different ASP. This protects against the loss of both the files and the receivers if a disk media failure occurs.

The advantages are:

- Additional data protection. By separating libraries or objects in a user ASP, you protect them from data loss when a disk unit in the system ASP fails. For example, if you have a disk unit failure, and data contained on the system ASP is lost, objects contained in user ASPs are not affected and can be used to recover objects in the system ASP. (For information on recovering objects, see "Recovering from Disk Unit Media Failures" on page 16-3.) Conversely, if a failure causes data contained in a user ASP to be lost, data in the system ASP is not affected.
- Improved system performance. You can place libraries or objects in a user ASP, allowing you to dedicate the disk units in the ASP exclusively for the use of those objects. If you do extensive journaling, a dedicated disk unit can also improve journaling performance.

  However, placing many active journal receivers in the same user ASP is not productive because the resulting contention between writing to more than one receiver in the ASP can slow system performance. For maximum performance, place each active journal receiver in a separate user ASP.
- All disk types can be allocated to a user ASP, but unit 1 is always allocated to the system ASP (ASP 1).
- User ASPs are configured using the DST display, Work with ASP Configuration. For more information, see the topic "Creating a User ASP and Adding Disk Units to the New User ASP" on page 15-1.
- As with the system ASP, you can specify individual threshold values for each user ASP. For information on how to do this using dedicated services tool or system service tools, see the topics "Accessing DST Options" on page 14-19 and "Accessing SST Options" on page 14-18. If you do not specify a value, the system uses the default value of 90%.

- You can use mirrored or checksum protection on one or more ASPs.

  Notes:

  1. All ASPs, including the system ASP, must have mirrored protection to ensure that the system will continue to run after a disk failure in an ASP.
  2. If a disk failure occurs in an ASP that does not have mirrored protection, the system may not continue to run, depending on the disk error.
  3. If a disk failure occurs in an ASP that has mirrored protection, the system will continue running (unless the failed unit has a mirrored unit that has failed).

Considerations for Using User ASPs

You can create two types of user ASPs. The first type is recommended.

1. Create your libraries in the user ASP by using the ASP parameter on the CRTLIB command. All objects that are created in those libraries are allocated to that user ASP.
2. Create your libraries in the system ASP (ASP 1), and then create the journals, journal receivers, and save files in user ASPs by using the ASP parameter on the corresponding create command (CRTJRN, CRTJRNRCV, or CRTSAVF).

   Note: This type of ASP is not recommended because the recovery steps are more complex.

The two types of user ASP are mutually exclusive: a user ASP can contain either libraries (and objects created in these libraries) or isolated journals, journal receivers, and save files, but not both.

Consider the following when creating user ASPs:

- An IPL cannot be performed with a failed unit in any ASP (unless the failing unit has a mirrored unit that has not failed). The failing unit must be repaired or replaced, or the ASP definition for the failing unit must be removed by changing the disk configuration.
- The system ASP should have checksum or mirrored protection. Using checksum or mirrored protection reduces the chance of the system ASP being cleared. When the system ASP is lost, addressability to objects in every user ASP is lost. The addressability can only be recovered by restoring the entire system or by running the Reclaim Storage (RCLSTG) command.
- There are performance considerations when using checksum protection or mirrored protection. For more information about performance considerations, see Chapter 23, "Performance Considerations for Mirrored Protection" on page 23-1, or see "System Performance When Using Checksum Protection" on page 17-3.
- When a user ASP becomes full, objects can overflow into the system ASP. If the user ASP overflows, the overflow status for the ASP should be reset as soon as possible for the following reasons:
  - The contents of the user ASP are cleared if a user ASP overflows and a data-loss failure occurs that causes either the user ASP or the system ASP to be cleared.

ASP/Disk Recovery

- If a user ASP is cleared while in overflow status, the user must run the Reclaim Storage (RCLSTG) command to recover. A reclaim storage operation finds and cleans up any objects or parts of objects that overflowed into the system ASP.

  Note: You can only use DST to reset the overflowed ASP.

- No database network can cross ASP boundaries:
  - Users cannot create a file in one ASP that depends on a file in a different ASP. All based-on physical files for a logical file must be in the same ASP as the logical file. The system builds access paths only for database files in the same ASP as the based-on physical file (temporary queries are not limited). Access paths are never shared by files in different ASPs. Record formats are not shared between different ASPs; a format request is ignored and a new record format is created.
  - SQL collections are not allowed in user ASPs. SQL collections require logical files with based-on physical files in library QSYS. Because the based-on physical files are in the system ASP and the logical files in a user ASP, an SQL collection is not allowed.
  - Journaling cannot be started on an object (STRJRNPF or STRJRNAP command) if the journal (object type *JRN) and the object to be journaled are in different libraries in different ASPs.
  - Journaling cannot be started again for a file that is saved and then restored to a library in a different ASP that does not contain the library for the journal. The libraries for the journal and the file must be in the same ASP in order for journaling to be automatically started again for the file.
  - The Reorganize Physical File Member (RGZPFM) command requires sufficient work space to contain a copy of the member being reorganized. The work space must exist in the same ASP.
- Normally, you cannot move objects from a library in one ASP to a library in another ASP using the Move Object (MOVOBJ) command. There are two exceptions to this rule:
  - Save files that are in a user ASP can be moved to libraries in the system ASP.
  - Files in library QRCL can be moved to their original libraries.

  Journals and journal receivers cannot be moved out of the libraries that they were created in. The only way to move an object from one ASP to another is to create a new object. An object can be placed in a different ASP by doing the following:

  1. Save the object.
  2. Delete the existing object.
  3. Create the library using the Create Library (CRTLIB) command.
  4. Restore the object to the library.

Object Types Not Allowed in a User ASP

The Create Library (CRTLIB) command allows you to place a library in a specific user ASP.

Commands that restore or create objects allowed in user ASPs, automatically place the object in the same ASP as the library. The system does not allow a user to create an object in a user ASP if the object type is not allowed in a user ASP.

The following list shows the object types that are not allowed in a user ASP.

Object Type	Description
***AUTL**	Authorization list
***CFGL**	Configuration list
***CNNL**	Connection list
***COSD**	Class-of-service description
***CTLD**	Controller description
***DEVD**	Device description
***DOC**	Documents
***DTADCT**	Data dictionary
***EDTD**	Edit description
***FLR**	Folders
***IGCSRT**	DBCS sort
***IGCTBL**	DBCS table
***JOBQ**	Job queue
***LIND**	Line description
***MODD**	Mode description
***NWID**	Network interface description
***OUTQ**	Output queue
***PRDAVL**	Product availability
***PRDDFN**	Product definition
***PRDFUN**	Product function
***PRDLOD**	Product load
***RCT**	Reference code translation table
***S36**	System/36 machine description
***USRPRF**	User profiles

Limiting the Types of Objects in a User ASP

You can limit the types of objects in a user ASP (ASPs 2 through 16) if a library does not exist in the user ASP. However this type of ASP is not recommended because it requires a complex set of recovery steps.

You can limit the object types to:

- Journals (*JRN)
- Journal receivers (*JRNRCV)
- Save files (*SAVF)

You create any of the object types listed above in a user ASP and have the library in the system ASP. You can isolate the object in a user ASP by specifying a user ASP number in the ASP parameter on the Create command for the specific object type (CRTJRN, CRTJRNRCV, or CRTSAVF) will create the object in the specified ASP. If the user ASP contains any of these isolated objects, no other object types (other than journals, journal receivers, and save files) can be created in that user ASP. After you have created the objects in the user ASP, delete the objects in the system ASP.

If you want to create a library in a user ASP, but the user ASP contains journals, journal receivers, and save files whose library is in the system ASP, you must first delete all the journals, journal receivers, and save files in the user ASP.

Checksum protection is not recommended for user ASPs containing journals, journal receivers, and save files whose library is in the system ASP for the following reasons:

- Inefficient because the use of journaling and save files is a different form of backup. Checksum protection would be backing up a backup copy.

- Poor performance because of redundancy data.

Use the Display Object description (DSPOBJD) command and specify DETAIL(*FULL) to identify the ASP that the object is in.

Planning the Configuration of User ASPs

Figure 14-3 on page 14-13 illustrates a configuration of system and user auxiliary storage pools.

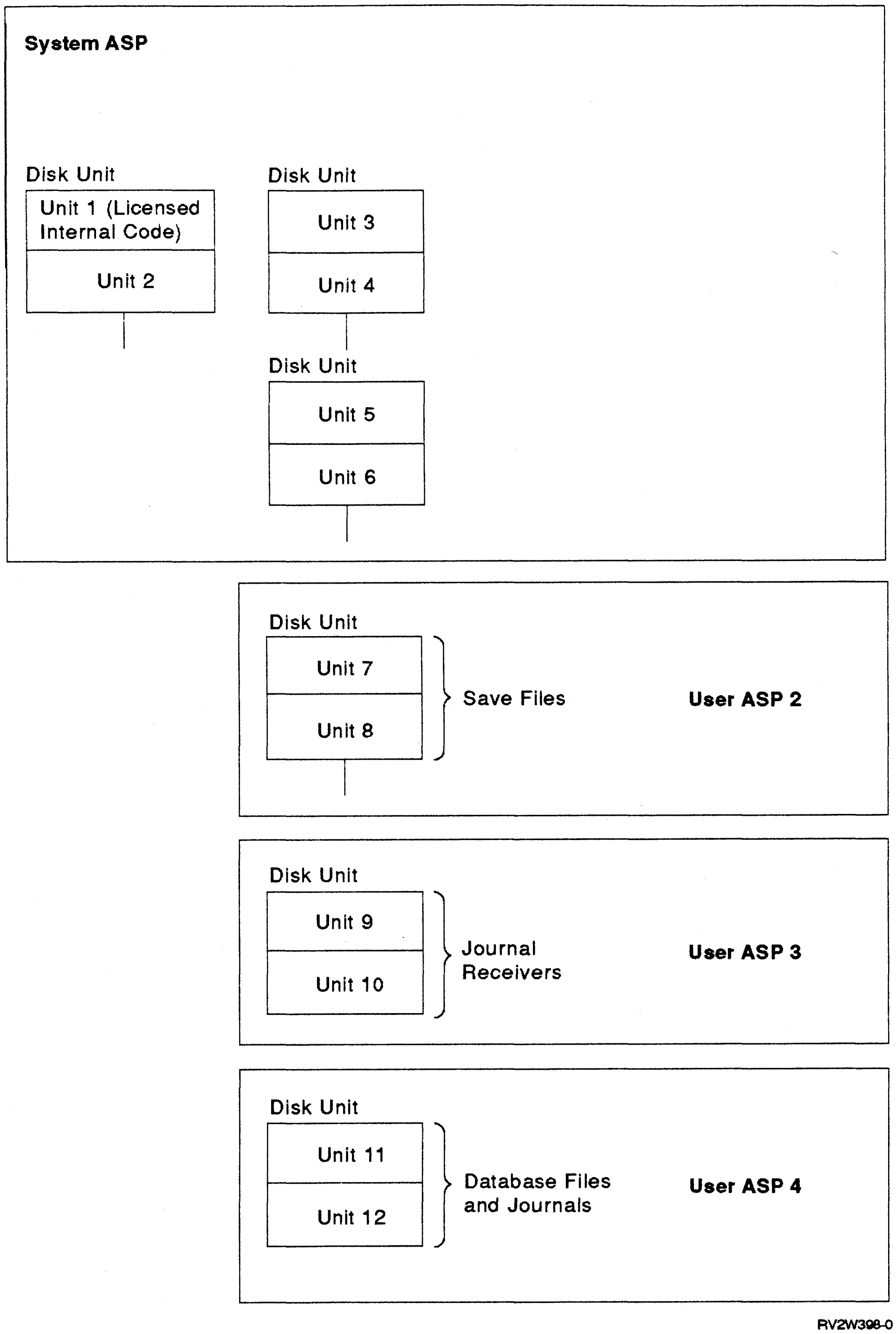


Figure 14-3. Example of System and User ASP Configuration

When planning your user ASP configuration, you must first consider what objects to place in each user ASP. The following list can help you determine the appropriate number of user ASPs for your system:

- Recovery

  If you are using user ASPs for recovery, configure at least one complete storage device for a single user ASP. Some 9332 or 9335 disk unit failures can damage both storage units in the disk unit. If this type of failure occurs, and the two units are in different unprotected ASPs, you would lose the data on both ASPs. If you configure one unit in the system ASP containing data files, and the second unit as a user ASP containing the journal receiver, a single device failure could destroy both the data files and journal entries for the file. The exceptions to this rule are the 2800, 9336, and 6107 disk units. The storage units within the 2800, 9336, or 6107 models 20 and 40 disk units are independent of each other.

- Improved system performance only

  If you are using user ASPs for better system performance, consider dedicating the ASP to one object that is very active. In this case, you can configure the ASP with only one storage unit, if that provides sufficient capacity, leaving the other storage unit in the device free for assignment to another ASP. However, this is not recommended.

  For example, journaling performance can be improved by allocating one user ASP exclusively for journal receivers attached to the same journal. By having the journal and database files in a separate ASP from the attached receivers, there is no contention for journal receiver write operations because the units associated with the ASP do not have to be repositioned before each read or write operation.

  Another way to improve performance is to make sure there are enough storage units in the user ASP to support the number of physical input and ouput operations that are done against the objects in the user ASP. You may have to experiment by moving objects to a different user ASP and then monitoring performance in the user ASP to see if the storage units are used excessively. For more information on working with disk status (WRKDSKSTS command) to determine if the storage units have excessive use, see the *Work Management Guide*. If the units have excessive use, you should consider adding more disk units to the user ASP.

- Extensive journaling

  Placing journal receivers in separate ASPs allows easier recovery if the user ASP overflows. If journals and files being journaled are in the same ASP as the receivers and the ASP overflows, you must end journaling of all files and delete all objects on the ASP. If the journal receiver is in a different ASP than the journal, and the user ASP that the receiver is in overflows, you can create a new receiver in a different user ASP, change the journal (CHGJRN command), and then reset the overflowed ASP without ending journaling.

- Access path journaling

  If you plan to use access path journaling, it is recommended that you first journal access paths to the system ASP (ASP 1) for a few days and look at storage requirements for the receiver before you allocate the specific size for a user ASP.

User ASPs can be configured with more than one storage unit. The maximum number of storage units allowed is determined by the maximum allowed for the entire system. User ASPs are configured and identified by a user-assigned number, from 2 through 16, and are assigned using the DST Work with ASP Configuration display. For more information on how to do this, see the topic "Creating a User ASP and Adding Disk Units to the New User ASP" on page 15-1.

There are three ways to obtain the storage units. The following list describes the three ways. The first is the most preferable only if the save and restore is avoided. Otherwise, the first and second are the same.

- You can reconfigure some of the storage units in the system ASP. If checksum protection is in effect, you may have units in the checksum protected system ASP that are not eligible to participate in checksum sets. (For information on checksum protection, see the topic Chapter 18, "Working with Checksum Protection" on page 18-1.)

  Because these units cannot contain data from protected storage, they are probably of limited value in the system ASP that has checksum protection. You can make better use of these units by moving them into user ASPs. The advantages of this reconfiguration include:

  - No additional disks need to be purchased to form the user ASP.
  - A save and restore operation of the ASP that has checksum protection is not required to accomplish the reconfiguration.

- You can move a unit from the system or user ASP. This method requires that you save and restore the ASP from which the unit is taken. No additional equipment is required.
- You can purchase new disks and add storage units to your existing user ASPs, or configure them into new user ASPs. Although this is an additional expense, this method removes the necessity of saving and restoring the ASP.

If you choose to reassign a unit from one ASP to another, you must carefully consider the effect that it does have on the capacity of both ASPs. After reviewing the system disk configuration, you must determine which disk units are candidates for reassignment.

Determine the effect of removing disk units from the source ASP by considering the space requirements for the objects to be stored in the source ASP and target user ASPs:

- For 9406 model D70 and D80 system units, sufficient space must be reserved in the system ASP for main storage dump space. Enough 2800-001 storage units must be provided for this reserved space. See note 3 on page 14-7 for the restrictions.
- The current storage requirements for an ASP can be determined from the SST Display Disk Configuration Capacity display.
- If you are planning to start checksum protection for an ASP, consider checksum disk requirements. Checksum protection requires that like types of disk units be grouped in minimum sets of two and can decrease the amount of disk storage available in the ASP. For information on checksum disk requirements refer to Chapter 18, "Working with Checksum Protection" on page 18-1.
- If you are planning to start mirrored protection for an ASP, consider the disk requirements for mirrored protection. Mirrored protection increases the number of disk units required to provide a given amount of disk storage. For more information on disk unit requirements for mirrored protection, see "Calculating Mirrored Capacity" on page 20-4.
- Determine storage requirements for objects you intend to create in user ASPs by running test applications, programs, or commands and observing the disk space used. From those observations, you can determine overall storage requirements for the target user ASPs.

- If you are going to use checksum or mirrored protection on the system ASP, you must have a sufficient number of storage units to allow for good system performance.

Overview of SST and DST Options

Disk unit storage management procedures are available through system service tools (SST) and dedicated service tools (DST). The SST options are a subset of the DST options.

Full function for Work with Disk Unit options is available only through DST before the IPL step for Storage Management Recovery is started. Once Storage Management Recovery is started, the Work with Disk Units options available through DST are identical to the SST subset of options.

System service tools (SST) and dedicated service tools (DST) provide menus for you to use to display and to change your disk configuration. This topic describes the menus and the procedures to use when working with auxiliary storage pools, checksum protection, and mirrored protection. Some of the disk functions can be done by service personnel. Other functions can be performed by you:

- Display Disk Configuration display:
  - Display disk configuration
  - Display disk configuration status
  - Display disk configuration capacity
  - Display disk configuration protection
  - Display nonconfigured units
- Work with ASP Configuration display:
  - Display disk configuration capacity
  - Create user ASP
  - Delete user ASP
  - Add disk unit to ASP
  - Reset overflowed user ASP
  - Change ASP storage threshold
  - Move unit from one ASP to another
  - Remove unit from configuration
- Work with Checksum Protection display:
  - Display checksum configuration
  - Start checksum protection
  - Stop checksum protection
  - Change unprotected storage
  - Calculate checksum configuration
- Work with Mirrored Protection display:
  - Display disk configuration
  - Start mirrored protection
  - Stop mirrored protection
  - Calculate mirrored capacity

The following functions are performed by or under the direction of the service representative to analyze and correct hardware problems.

- Analyze disk problem

- Work with Disk Unit Recovery display:
  - Save disk unit data
  - Restore disk unit data
  - Replace configured unit
  - Assign missing unit
  - Recover configuration
  - Disk unit recovery procedures
  - Suspend/resume mirrored protection
  - Copy disk unit data
  - Delete disk unit data
  - Upgrade load source utility
- Work with Disk Unit Information display:
  - Work with vital product data
  - Work with field replacement unit data

Figure 14-4 on page 14-18 provides an overview of disk unit storage management functions you can perform.

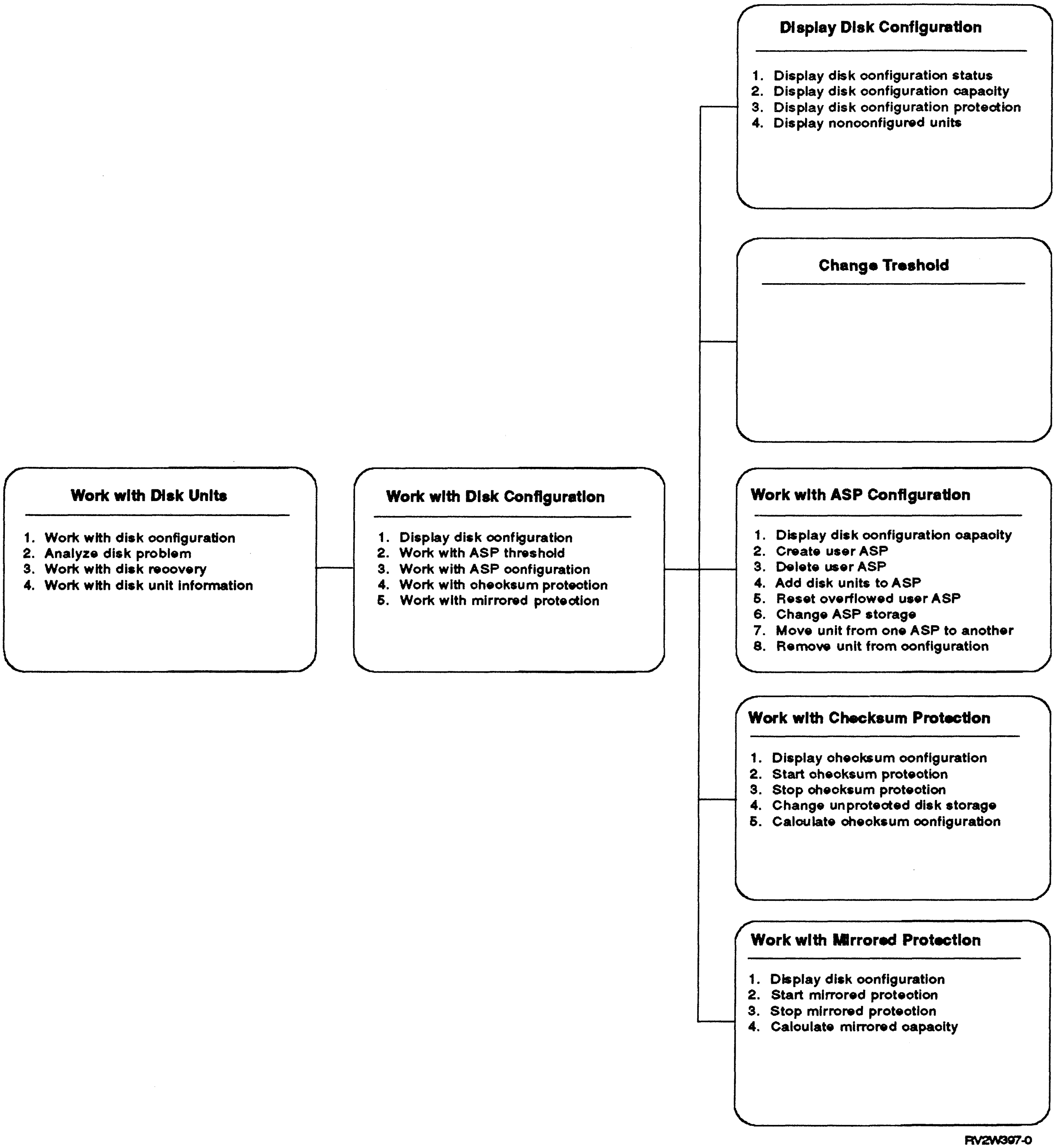


Figure 14-4. Disk Storage Management Functions

Accessing SST Options

SST options are started using the Start System Service Tools (STRSST) command or through the following displays:

- AS/400 Main Menu
- Problem Handling
- SST
- Work with Disk Units from SST menu

The Work with Disk Units display from SST is a different display than the DST full function display because it shows only those options allowed under SST. Figure 14-5 shows the Work with Disk Units display.

```
                    Work with Disk Units

Select one of the following:

     1. Display disk configuration
     2. Display checksum configuration
     3. Calculate checksum configuration
     4. Work with ASP storage threshold
     5. Work with disk unit recovery
     6. Work with disk information
     7. Calculate mirrored capacity








Selection
     __

F3=Exit          F12=Cancel
```

Figure 14-5. Work with Disk Units Display for SST Options

Use SST when your system has already been installed and you want to display your disk configuration, display your checksum configuration (if you have checksum protection), calculate a checksum configuration, plan a system with mirrored protection, or display, change, or calculate storage thresholds. If your system has mirrored protection, you can also suspend or resume mirrored protection and perform other recovery procedures.

Accessing DST Options

The DST options are started manually during an IPL of the system. Because full DST functions are only available during IPL processing, you must first power down your system, if it is currently running, before starting the IPL procedure.

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   `CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)`

3. End all subsystems:

   `ENDSBS SBS(*ALL) OPTION(*IMMED)`

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.

4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   `PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)`

7. When the system has powered down, the IPL or Install the System display appears.

```
                         IPL or Install the System

Select one of the following:

         1. Perform an IPL
         2. Install the operating system
         3. Use Dedicated Service Tools (DST)
         4. Perform automatic installation of the operating system









Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                              © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                   Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                     Use Dedicated Service Tools (DST)

Select one of the following:

           1. Perform an IPL
           2. Install the operating system
           3. Work with licensed internal code
           4. Work with disk units
           5. Work with DST environment
           6. Select DST console mode
           7. Start a service tool
           8. Perform automatic installation of the operating system
           9. Work with save storage and restore storage






Selection
     4

F3=Exit           F12=Cancel
```

Select option 4 (Work with disk units) on the Use Dedicated Service Tools (DST) menu.

Chapter 15. Working with Auxiliary Storage Pools

The following ASP operations are described in this topic:

- Creating a user ASP and adding units to the new ASP
- Deleting a user ASP
- Adding units to an existing ASP
- Calculating storage requirements using SST
- Moving a disk unit from the system ASP to another ASP
- Moving a disk unit from one user ASP to another
- Resetting an overflowed user ASP
- Removing a disk unit from the system ASP
- Removing a disk unit from a user ASP

Creating a User ASP and Adding Disk Units to the New User ASP

This example attaches a new disk unit to the system and creates user ASP 2 to contain it. The threshold limit in this example is set to 88%.

Your service representative attaches the new disk unit. The new unit is in non-configured status.

Task Overview

1. Access DST
2. Display Disk Configuration
3. Create the user ASP
4. Add units to the ASP
5. Change Storage Threshold

Warning: A sufficient number of 2800-001 storage units must be configured to the system ASP (ASP 1) to allow for enough main storage dump space. (See note 3 on page 14-7.)

Task 1. Access DST Options

Do the following steps to access the DST options.

1. Sign on to DST using the following steps:
   a. Notify the users to sign off the system by sending a break message.
   b. Change the QSYSOPR message queue to break mode:
   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```
   c. End all subsystems:
   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```
   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
   d. Insert the key into the keylock switch on the control panel.
   e. Turn the key to the Manual position.
   f. Power down the system:

ASP/Disk Recovery

```
PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
```

g. When the system has powered down, the IPL or Install the System display appears.

```
                         IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system









Selection
     3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                              © COPYRIGHT IBM CORP. 1980, 1991.
```

h. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                     Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . . .   _____
```

i. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

The Use Dedicated Service Tools (DST) menu is shown.

```
                     Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage







Selection
    4

F3=Exit          F12=Cancel
```

Task 2. Display the Disk Configuration

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                          Work with Disk Units

Select one of the following:

    1. Work with disk configuration
    2. Analyze disk problem
    3. Work with disk recovery
    4. Work with disk unit information









Selection
    1

F3=Exit        F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display after the disk is attached.

ASP/Disk Recovery

```
                          Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshhold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection









Selection
     1

F3=Exit          F12=Cancel
```

3. Select option 1 (Display disk configuration status) on the Work with Disk configuration display.

```
                              Display Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Display disk configuration capacity
     3. Display disk configuration protection
     4. Display non-configured units









Selection
     1

F3=Exit          F12=Cancel
```

Note: Disk units on the system are either configured or nonconfigured. Options 1, 2, and 3 on the Display Disk Configuration Status display show information about the configured units. Option 4 shows the nonconfigured units attached to the system. When you physically attach a unit to the system, it becomes part of the nonconfigured set until you place it in an ASP.

4. Select option 1 (Display disk configuration status) and press the Enter key.

```
                     Display Disk Configuration Status

               Serial
ASP  Unit  Number       Type  Model  Address        Status
 1                                                   Mirrored
        1  10-00A7529   9332   400   0010-0000FFFF  Active
        1  10-00A4936   9332   400   0010-0100FFFF  Active
        2  10-00A7498   9332   400   0010-0300FFFF  Resuming
        2  10-00A7529   9332   400   0010-0001FFFF  Active
        3  10-00A7498   9332   400   0010-0600FFFF  Active
        3  10-00A4936   9332   400   0010-0101FFFF  Active
 2                                                   Unprotected
        8  10-00A7530   9332   400   0020-0701FFFF  Configured
 3                                                   Unprotected
        6  10-00A7530   9332   400   0010-0700FFFF  Configured






Press Enter to continue.

F3=Exit   F5=Refresh   F11=Display non-configured units   F12=Cancel
```

Note: Non-configured units are not shown on this display. To display non-configured units, press F11 (Display non-configured units).

The following fields appear on the Display Disk Configuration Status display:

- *ASP*. The auxiliary storage pool number.
- *Unit*. The number assigned by the system to identify a specific disk unit.
- *Serial Number*. The number assigned by the manufacturer to identify a specific disk unit.
- *Type*. The number assigned by the manufacturer to identify a type of disk unit.
- *Model*. The numbers or letters used to identify the feature level of a specific product type.
- *Address*. Identifies the following:
  - Location of the storage device controller card (columns 1 through 4)
  - Functional controller for the disk unit (columns 5 and 6)
  - Disk unit itself (columns 7 and 8)
  - FFFF (columns 9 through 12)
- *Status*. The valid values for this field are:
  - For ASPs:
    - *Unprotected*. No protection exists for this ASP.
    - *Checksummed*. The units in the ASP are protected by checksum protection if the units are a part of a checksum set.
    - *Mirrored*. All the units in the ASP are protected by mirrored protection.
  - For units in an unprotected ASP.
    - *Configured*.
  - For units in an ASP that has checksum protection.

ASP/Disk Recovery

- *Checksummed*. Indicates that the unit is part of a checksum set.
- *Configured*. Indicates that the unit is not part of a checksum set.

— For units in a mirrored ASP.

- *Active*. This unit is capable of having data written to it, or read from it.
- *Suspended*. This unit is not capable of having data written to it, or read from it. The data on this unit is not current. For example, if the disk needs repair action or has been manually suspended, it would be in a *Suspended* state.
- *Resuming*. The current data is being copied (or will be copied) to this unit from the other active unit of the mirrored pair.
- *Unknown*. The system configuration mechanism cannot determine what the valid configuration should be.

5. Press F11 (Display non-configured unit) on the Display Disk Configuration Status display.

```
                     Display Non-Configured Units

Serial
Number        Type   Model   Address          Status
10-00A7503    9332    400    0010-0100FFFF    Non-configured
10-00A7503    9332    400    0010-0101FFFF    Non-configured
10-00A3651    9332    400    0010-0400FFFF    Non-configured
10-00A3651    9332    400    0010-0401FFFF    Non-configured








 Press Enter to continue.

 F3=Exit   F5=Refresh  F11=Display disk configuration Status  12=Cancel
```

6. Write down the serial number and address of the unit that you are going to use.
7. Return to the Work with Disk Configuration display by pressing F12 (Cancel) three times.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum configuration
     5. Work with mirrored protection

Selection
  3

F3=Exit          F12=Cancel
```

Task 3. Create the User ASP

1. Select option 3 (Work with ASP configuration) on the Work with Disk configuration display.

```
                    Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration

Selection
     2

F3=Exit          F12=Cancel
```

2. Select option 2 (Create user ASP) on the Work with Disk Configuration to select user ASP 2 as the ASP to be created.

```
                         Create User ASP

Type choice, press Enter.

  ASP number . . . . . .    _         2-16
```

3. Enter an ASP number in the *ASP number* prompt on the Create User ASP display and press the Enter key.

Task 4. Adding Disk Units to the New User ASP

1. After you enter an ASP number, the Select Units to Add to ASP display appears.

```
                          Select Units to Add to ASP

Type options, press Enter.
  1=Add unit to ASP

        Serial
Option  Number        Type   Model   Address
  1     10-00A7503    9332    400    0010-0100FFFF
  1     10-00A7503    9332    400    0010-0201FFFF
        10-00A3651    9332    400    0010-0001FFFF
        10-00A3651    9332    400    0010-0401FFFF









F3=Exit                      F12=Cancel
```

2. Use the serial number of the units you wrote down when you displayed the nonconfigured units to select the disk units to place within the specified ASP. Type a **1** in the *Option* column for each unit and press the Enter key.

   Note: Each storage unit within a replaceable disk unit will have the same serial number.

   The system assigns a threshold value of 90% for newly created user ASPs.

   The Confirm Add Units display is shown. The Confirm Add Units display shows what the the entire system configuration will be when you add the units. Use the serial number of the disk unit you wrote down when you displayed the disk configuration to verify that you are selecting the correct disk units to add to the ASP.

   Note: It is important that you verify that you want the disk units in the specified ASP. If you add a unit to the wrong ASP and want to move it, the source ASP is cleared after the move operation and will require a restore operation to recover the source ASP.

```
                              Confirm Add Units

Units will be added to ASP number  . . . :     2

Add will take several minutes for each unit.  The system will
have the displayed protection after the unit(s) are added.

Press Enter to confirm your choice for 1=Add units.
Press F12=Cancel to return and change your choice.


              Serial
ASP  Unit  Number       Type  Model  Address          Protection
 1                                                    Unprotected
        1  10-0002772   9332   200   0010-0100FFFF    Unprotected
        2  57-0058803   9335   B01   0110-0100FFFF    Unprotected
 2                                                    Unprotected
        3  10-00A7503   9332   400   0010-0100FFFF    Unprotected
        4  10-00A7503   9332   400   0010-0101FFFF    Unprotected



* - Indicates that a unit has reduced protection

F12=Cancel
```

3. If you are satisfied with the configuration, press the Enter key to add the disk units to the ASP.

   Adding units takes several minutes, during which the system may appear inactive. The time it takes depends on the size of each unit being added and the ability of the system to do multiple adds at the same time.

Task 5. Changing the Storage Threshold of the New User ASP

1. After the add operation is complete, return to the Work with ASP Configuration display.

```
                         Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration






Selection
    6

F3=Exit          F12=Cancel
```

2. Select option 6 (Change ASP storage threshold) on the Work with ASP Configuration menu.

```
                                 Select ASP

Type option, press Enter.
  1=Select

                                          --Protected--    --Unprotected--
Option  ASP  Threshold  Overflow           Size  %Used       Size  %Used
  _      1      90%        No                0  0.00%        627  69.15%
  1      2      90%        No                0  0.00%        855   0.25%
```

3. Select the ASP you want to change the threshold for on the Select ASP display and press the Enter key. The following display is shown.

```
                        Change Storage Threshold

                              --Protected--  --Unprotected--
ASP   Threshold  Overflow      Size   %Used      Size  %Used
 2          90%    No             0   0.00%       855  0.25%

This is an unprotected ASP.  The threshold represents the amount
of unprotected storage used before a warning message is sent to
the system operator.

Type choice, press Enter.

  New threshold . . . . . . . .      88% 1-100








F3=Exit    F11=Display disk configuration status    F12=Cancel
```

Figure 15-1. Change Storage Threshold Display

4. Change the threshold value from the default value of 88% to the value you want.

   The display used to change the storage threshold has the following fields:

   - *ASP*. The auxiliary storage pool number.
   - *Threshold*. The current threshold. The system notifies you when this percentage of storage is full.
   - *Overflow*. Indicates if this ASP has overflowed.
   - *Protected Size*. Amount of storage in the ASP that is protected by checksum or mirrored protection.
   - *Protected % Used*. Percent of protected size that is used.
   - *Unprotected Size*. Amount of storage in the ASP that is unprotected.
   - *Unprotected % Used*. Percent of unprotected size that is used.
   - *New threshold*. The field used to enter a new threshold value. The new value must be higher than the current amount used or a warning message is shown or sent to QSYSOPR.

5. Return to the Work with ASP Configuration display.
6. Select option 1 (Display disk configuration capacity).
7. Verify the configuration on the Display Disk Configuration Capacity display.

```
                      Display Disk Configuration Capacity

                                                 --Protected--  --Unprotected--
ASP  Unit  Type  Model  Threshold  Overflow  Size  %Used         Size  %Used
 1                          90%     No        600  77.84%           0  0.00%
       1   9332  400                          200  99.87%           0  0.00%
       1   9332  400                          200  99.87%           0  0.00%
       2   9332  400                          200  66.50%           0  0.00%
       2   9332  400                          200  66.50%           0  0.00%
       3   9332  400                          200  67.16%           0  0.00%
       3   9332  400                          200  67.16%           0  0.00%
 2                          88%     No                            400  0.53%
       8   9332  400                            0  0.00%          200  0.53%
       6   9332  400                            0  0.00%          200  0.53%





Press Enter to continue.

F3=Exit   F5=Refresh   F11=Display non-configured units    F12=Cancel
```

Deleting a User ASP

Deleting an ASP causes all units within the ASP to be removed from the configuration and all data in the ASP to be destroyed. You cannot delete the system ASP. A warning message appears, providing the opportunity to cancel the deletion. You may save data before deleting a user ASP.

Before deleting a user ASP, be sure to delete all the objects in the user ASP. Deleting the objects allows the operating system to correctly update the control information and avoids references to destroyed or damaged objects. The Display Object Description (DSPOBJD) command can be used to determine the ASP storage for an object that has been allocated.

To delete a user ASP, do the following:

Task 1. Delete the Objects in the User ASP

Delete the objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:

1. Display the library in the user ASP:

   ```
   WRKLIB LIB(library-name)
   ```

2. Type a **12** (Work with objects) in the *Opt* column and press the Enter key.
3. Find the objects to be deleted in the *Object* column.
4. Type a **4** (Delete) in the *Opt* column for each object you want to delete.
5. Press the Enter key.

Task 2. Access DST Options

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   `CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)`
3. End all subsystems:

   `ENDSBS SBS(*ALL) OPTION(*IMMED)`

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   `PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)`
7. When the system has powered down, the IPL or Install the System display appears.

```
                         IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system









Selection
     3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                    © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                  Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                     Use Dedicated Service Tools (DST)

Select one of the following:

           1. Perform an IPL
           2. Install the operating system
           3. Work with licensed internal code
           4. Work with disk units
           5. Work with DST environment
           6. Select DST console mode
           7. Start a service tool
           8. Perform automatic installation of the operating system
           9. Work with save storage and restore storage

Selection
    4

F3=Exit            F12=Cancel
```

Task 3. Delete the User ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                          Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information

Selection
    1

F3=Exit         F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection

Selection
     3

F3=Exit          F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Configuration display.

```
                    Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration

Selection
    3

F3=Exit          F12=Cancel
```

4. Select option 3 (Delete user ASP) on the Work with ASP Configuration display and press the Enter key.

```
                    Delete User ASP

Type option, press Enter
  4=Delete

                                    --Protected--  --Unprotected--
Option  ASP  Threshold  Overflow    Size  %Used     Size  %Used
         1      90%     No           600  77.84%       0  0.00%
  4      2      90%     No             0   0.00%     200  0.53%
         3      90%     No             0   0.00%     200  0.53%




Press Enter to continue.

F3=Exit   F5=Refresh   F11=Display non-configured units   F12=Cancel
```

5. Type 4 in the *Option* field of the ASP you want to delete and press the Enter key. The Confirm Delete of User ASP display is shown.

```
                 Confirm Delete Of User ASP

Warning: Deleting a user ASP will remove all units of that
ASP from the configuration.  The units will become nonconfigured.

Press F10 to confirm your choice for 4=delete
Press F12=Cancel to return to change your choice

                                    --Protected--  --Unprotected--
Option  ASP  Threshold  Overflow    Size  %Used     Size  %Used
  4      2      90%     No             0   0.00%     200  0.53%




F10=Confirm  F12=Cancel
```

6. Press F10 (Confirm) to confirm that the user ASP should be deleted.

Adding Units to an Existing ASP

This example adds a new disk unit to the system ASP or a user ASP.

Your service representative attaches the new disk unit. The new unit is in non-configured status.

Task Overview

You will perform the following steps during this task

1. Access DST
2. Display Disk Configuration
3. Add Unit to existing ASP

Warning: A sufficient number of 2800-001 storage units must be configured to the system ASP (ASP 1) to allow for enough main storage dump space. (See note 3 on page 14-7.)

Task 1. Access DST Options

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

3. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
   ```

7. When the system has powered down, the IPL or Install the System display appears.

```
                         IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system










Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                             © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                    Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage

Selection
    4

F3=Exit          F12=Cancel
```

Task 2. Display the Disk Configuration

You will use the information in step 6 when adding the units to the existing ASP.

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                         Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information

Selection
    1

F3=Exit        F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display after the disk is attached.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshhold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection









Selection
     1

F3=Exit          F12=Cancel
```

3. Select option 1 (Display disk configuration status) on the Work with Disk configuration display.

```
                    Display Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Display disk configuration capacity
     3. Display disk configuration protection
     4. Display non-configured units









Selection
     1

F3=Exit          F12=Cancel
```

Note: Disk units on the system are either configured or nonconfigured. Options 1, 2, and 3 on the Display Disk Configuration Status display show information about the configured units. Option 4 shows the non-configured units attached to the system. When you physically attach a unit to the system, it becomes part of the nonconfigured set until you place it in an ASP.

4. Select option 1 (Display disk configuration status) and press the Enter key.

```
                    Display Disk Configuration Status

              Serial
ASP  Unit  Number       Type  Model  Address        Status
 1                                                   Mirrored
        1  10-00A7529  9332   400   0010-0000FFFF  Active
        1  10-00A4936  9332   400   0010-0100FFFF  Active
        2  10-00A7498  9332   400   0010-0300FFFF  Resuming
        2  10-00A7529  9332   400   0010-0001FFFF  Active
        3  10-00A7498  9332   400   0010-0600FFFF  Active
        3  10-00A4936  9332   400   0010-0101FFFF  Active
 2                                                   Unprotected
        8  10-00A7530  9332   400   0020-0701FFFF  Configured
 3                                                   Unprotected
        6  10-00A7530  9332   400   0010-0700FFFF  Configured

Press Enter to continue.

F3=Exit   F5=Refresh    F11=Display non-configured units    F12=Cancel
```

Note: Non-configured units are not shown on this display. To display non-configured units, press F11 (Display non-configured units).

The following fields appear on the Display Disk Configuration Status display:

- *ASP*. The auxiliary storage pool number.
- *Unit*. The number assigned by the system to identify a specific disk unit.
- *Serial Number*. The number assigned by the manufacturer to identify a specific disk unit.
- *Type*. The number assigned by the manufacturer to identify a type of disk unit.
- *Model*. The numbers or letters used to identify the feature level of a specific product type.
- *Address*. Identifies the following:
  - Location of the storage device controller card (columns 1 through 4)
  - Functional controller for the disk unit (columns 5 and 6)
  - Disk unit itself (columns 7 and 8)
  - FFFF (columns 9 through 12)
- *Status*. The valid values for this field are:
  - For ASPs:
    - *Unprotected*. No protection exists for this ASP.
    - *Checksummed*. The units in the ASP are protected by checksum protection if the units are a part of a checksum set.
    - *Mirrored*. All the units in the ASP are protected by mirrored protection.
  - For units in an unprotected ASP.
    - *Configured*.
  - For units in an ASP that has checksum protection.

ASP/Disk Recovery

- *Checksummed.* Indicates that the unit is part of a checksum set.
- *Configured.* Indicates that the unit is not part of a checksum set.

— For units in a mirrored ASP.

- *Active.* This unit is capable of having data written to it, or read from it.
- *Suspended.* This unit is not capable of having data written to it, or read from it. The data on this unit is not current. For example, if the disk needs repair action or has been manually suspended, it would be in a *Suspended* state.
- *Resuming.* The current data is being copied (or will be copied) to this unit from the other active unit of the mirrored pair.
- *Unknown.* The system configuration mechanism cannot determine what the valid configuration should be.

5. Press F11 (Display non-configured unit) on the Display Disk Configuration Status display.

```
                    Display Non-Configured Units

Serial
Number       Type   Model   Address          Status
10-00A7503   9332    400    0010-0100FFFF    Non-configured
10-00A7503   9332    400    0010-0101FFFF    Non-configured
10-00A3651   9332    400    0010-0400FFFF    Non-configured
10-00A3651   9332    400    0010-0401FFFF    Non-configured









 Press Enter to continue.

 F3=Exit   F5=Refresh  F11=Display disk configuration Status  12=Cancel
```

6. Write down the serial number and address of the unit that you are going to use.
7. Return to the Work with Disk Configuration display by pressing F12 (Cancel) three times.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum configuration
     5. Work with mirrored protection

Selection
  3

F3=Exit        F12=Cancel
```

Task 3. Add Units to an Existing ASP

1. Select option 3 (Work with ASP configuration) on the Work with Disk Configuration display.

```
                    Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration

Selection
     2

F3=Exit        F12=Cancel
```

2. Select option 4 (Add units to existing ASP) on the Work with Disk Configuration display.

```
                         Select ASP to Add Units to

Type option, press Enter.
  1=Select

                                      --Protected--  --Unprotected--
Option  ASP  Threshold  Overflow      Size  %Used    Size  %Used
         1      90%      No            0  00.00%      628  77.84%
  1      2      90%      No            0   0.00       200   0.53%
```

3. Type a **1** in the *Option* column to select the ASP.

   After you select an ASP, the Select Units to Add to ASP display appears.

```
                         Select Units to Add to ASP

Type options, press Enter.
  1=Add unit to ASP

        Serial
Option  Number        Type  Model  Address
        10-00A7503    9332   400   0010-0100FFFF
        10-00A7503    9332   400   0010-0101FFFF
  1     10-00A3651    9332   400   0010-0400FFFF
  1     10-00A3651    9332   400   0010-0401FFFF









F3=Exit                     F12=Cancel
```

4. Use the serial number of the units you wrote down when you displayed the nonconfigured units to select the units to place within the specified ASP. Type a **1** in the *Option* column for each unit and press the Enter key.

   Note: Storage units within the same replaceable disk unit will have the same serial number.

```
                          Confirm Add Units

Units will be added to ASP number . . . :    2

Add will take several minutes for each unit.  The system will
have the displayed protection after the unit(s) are added.

Press Enter to confirm your choice for 1=Add units.
Press F12=Cancel to return and change your choice.


          Serial
ASP  Unit  Number       Type  Model  Address          Protection       CSS
 1                                                    Unprotected
       1  10-0002772  9332   200   0010-0100FFFF    Unprotected
       2  57-0058803  9335   B01   0110-0100FFFF    Unprotected
 2                                                    Unprotected
       4  10-00A3651  9332   400   0010-0400FFFF    Unprotected
       5  10-00A3651  9332   400   0010-0401FFFF    Unprotected



F12=Cancel
```

The Confirm Add Units display shows what the the entire system configuration will be when you add the units. Use the serial number of the unit you wrote down when you displayed the disk configuration to verify that you are selecting the correct units to add to the ASP.

Note: It is important that you verify that you want the disk units in the specified ASP. If you add a unit to the wrong ASP and want to move it, the source ASP is cleared after the move operation and will require a restore operation to recover the source ASP.

5. If you are satisfied with the configuration, press the Enter key to add the disk units to the ASP.

   Adding units will take several minutes during which the system may appear inactive. The time it takes depends on the size of each unit being added and the ability of the system to do multiple adds at the same time.

ASP/Disk Recovery

Calculating Disk Storage Requirements Using SST

When moving or removing unit a from the system ASP to a user ASP, you can calculate and verify disk storage requirements to determine if the disk units that would remain in the system ASP will provide sufficient storage. This is done using the option Work with ASP Threshold on the System Service Tools (SST) display. The following is an example of how to perform a possible calculation:

1. Start SST by typing the following:

   `STRSST`

   Press the Enter key. The following display is shown.

```
                         System Service Tools (SST)

 Select one of the following:

      1. Start a service tool
      2. Work with active service tools
      3. Work with disk units
      4. Work with disk data recovery









 Selection
     3

 F3=Exit          F10=Command entry         F12=Cancel
```

2. Select option 3 (Work with Disk Units) and press the Enter key. The following display is shown.

```
                            Work with Disk Units

 Select one of the following:

      1. Display disk configuration
      2. Display checksum configuration
      3. Calculate checksum configuration
      4. Work with ASP threshold
      5. Work with disk unit recovery
      6. Work with disk unit information
      7. Calculate mirrored capacity







 Selection
     1

 F3=Exit          F12=Cancel
```

3. Select option 1 (Display disk configuration) and press the Enter key. The following display is shown.

```
                         Display Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Display disk configuration capacity
     3. Display disk configuration protection
     4. Display non-configured units

Selection
     2

F3=Exit         F12=Cancel
```

4. Select option 2 (Display disk configuration capacity) and press the Enter key.

```
                   Display Disk Configuration Capacity

                                                --Protected--  --Unprotected--
ASP  Unit  Type  Model  Threshold  Overflow     Size  %Used      Size  %Used
 1                            90%  No              0  0.00%      4746 43.12%
        1  9332  400                               0  0.00%       200 97.84%
        2  9332  400                               0  0.00%       200 56.62%
        3  9332  400                               0  0.00%       200 57.70%
        4  9332  400                               0  0.00%       200 57.15%
        5  9332  400                               0  0.00%       200 58.34%
        6  9332  400                               0  0.00%       200 56.12%
        7  9332  200                               0  0.00%       200 58.95%
        8  9332  200                               0  0.00%       200 47.36%
        9  9332  200                               0  0.00%       200 47.59%

Press Enter to continue.

F3=Exit   F5=Refresh   F11=Display non-configured units    F12=Cancel
```

Figure 15-2. Display Disk Configuration Capacity Display

The following information is used to calculate storage requirements:

- *ASP*. The auxiliary storage pool number.
- *Unit*. The number assigned by the system to identify the disk unit.
- *Type*. The disk unit type assigned by the manufacturer.
- *Model*. The disk model that identifies the feature level for a specific type of disk unit.
- *Threshold*. The current threshold. The system notifies you when this percentage of storage is full.

- *Overflow.* Indicates if this ASP is overflowed.
- *Protected Size.* Amount of storage in the ASP that is protected by checksum or mirrored protection.
- *Protected % Used.* Percent of protected size that is used.
- *Unprotected Size.* Amount of storage in the ASP that is unprotected.
- *Unprotected % Used.* Percent of unprotected size that is used.

5. From the Display Disk Configuration Capacity display, calculate the disk storage requirements by doing the following:
6. For an ASP that does not have checksum or mirrored protection, the protected size will be zero. Multiply the unprotected size in megabytes by the unprotected percentage used.

```
(Unprotected Size) x (Unprotected % Used / 100) = Total Unprotected Space

                  4746MB x 43.12% / 100 = 2046.5MB
```

7. Determine the disk space you will be moving to the target user ASP by adding their unprotected storage size in megabytes per unit. This the size of the space being moved.
8. Determine the new size of the source ASP by subtracting the size of the space being moved from the old system ASP size.

```
(Old Size) - (Space Being Moved) = New Size

                 4746M - 800M = 3946MB
```

9. Calculate the new amount used percentage for the source ASP by dividing the total space used value that you calculated above by the new size in megabytes. (This assumes that none of the objects now in the system ASP will be moved to the target user ASP.)

```
(Total Space Used) / (New Size) x 100 = New Amount Used %

          2046.5MB / 3946MB = 51.86%
```

10. Review the new amount used percentage to see if it is within the guidelines you have established for the source ASP.

A similar procedure applies to units that are removed from an ASP that has mirrored protection. The system does not support moving units from a mirrored ASP. The units must be removed and then added. Use the *protected amount used* value and subtract only the size of one unit of each mirrored pair being removed.

Moving a Disk Unit from the System ASP to Another ASP

On the Work with ASP Configuration display, option 7 (Move units from one ASP to another) allows you to move a disk unit from one ASP to another.

In the following example, a unit currently in the system ASP is being moved to a user ASP.

Task Overview

You will perform the following steps during this task:

1. Save the system
2. Access DST
3. Move the unit
4. Restore the licensed internal code using function code 23
5. Restore the operating system
6. Restore the objects to the system

Warning: Use caution when changing the allocation of a unit from one ASP to another ASP. When the system changes the allocation of the unit, the data in the source ASP is destroyed, except for a special case under checksum protection (see the topic "Moving a Storage Unit Not in a Checksum Set from the System ASP to a User ASP" on page 18-46).

Warning: A sufficient number of 2800-001 storage units must be configured to the system ASP (ASP 1) to allow for main storage dump space. (See note 3 on page 14-7.)

Before You Begin

1. Moving a unit from the system ASP requires that you restore the licensed internal code again using function code 23 from the control panel during this procedure.
2. You cannot move unit 1 from the system ASP. (Unit 1 is reserved for the licensed internal code.)
3. Only one unit is moved at a time.
4. No units can be moved from the system ASP if any user ASP has overflowed and has not been reset, or selected to be reset.
5. All data in the ASP from which you are moving the unit is destroyed except when you move a unit that is not in a checksum set from an ASP that is checksum protected.
6. If the ASP to which you are moving the unit does not exist, it is created and the system assigns a default threshold of 90%.
7. No units can be moved into or out of an ASP that has mirrored protection. Units can be added or removed.

Task 1. Save the Entire System

Save the entire system with the appropriate save commands.

Notes:

1. When saving the entire system, it is important for you to remember to display the system log (QHST) and move or delete all but the current copies of the history log. These steps improve the performance of the save system operation.
2. To significantly reduce the time it takes to restore the system, consider specifying SAVLIB LIB(*NONSYS) ACCPTH(*YES).
3. If your application programs are dependent on spooled output files, you can save them. The system does not directly support saving spooled output files. However, you can save the files by copying them (CPYSPLF command) before you perform the save system operation. For an example of how to save spooled output files, see the topic "Example of Saving Spooled Output Files Using the Copy Spooled File (CPYSPLF) Command" on page 9-22

There are two methods you can use to save the entire system:

1. Using option 21 (Entire system) on the Save menu allows you to save the entire system without entering the commands.
2. Using the Save commands allows you to save the entire system by entering the commands from the command line.

Method 1. Using Option 21 (Entire system) on the Save Menu

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:
   ```
   WRKACTJOB
   ```
3. Display the system log QHST to verify it is up to date:
   ```
   DSPLOG  LOG(QHST)
   ```
   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:
   ```
   WRKF FILE(QSYS/QHST*)
   ```
   Look at the list to verify that you saved all copies of the log that will be needed later.
5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.
6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.
   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```
7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:
   ```
   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)
   ```
   Keep this list with your backup log or your save system tapes for future reference.
8. Go to the Save menu:

GO SAVE

The Save menu is shown.

```
SAVE                                 Save
                                                       System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Changed objects only
     7. Licensed programs
     8. Security data
     9. Storage

    20. All libraries other than system library
    21. Entire system
    22. All IBM libraries other than system library
                                                                 More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
     (C) COPYRIGHT IBM CORP. 1980, 1991.
```

9. Select option 21 (Entire system) from the Save menu and press the Enter key.

```
                           Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .   TAP01        Names
                                   ________
                                   ________
                                   ________

  Prompt for commands  . . . . .   Y            Y=Yes, N=No

  Check for active files . . . .   Y            Y=Yes, N=No
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device when the first tape is full.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting (for unattended save operations) and uses the default values.

Check for active files

Allows you to specify whether or not you want to check for active files. If you specify Y=Yes, the system sends a message when active files are encountered. You can end the checking process or clear the existing files and continue. If N=No is specified, all active files encountered during the save are cleared.

Option 21 will guide you through the following if you selected Y on the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS SBS(*ALL) OPTION(*IMMED)

b. SAVSYS

c. SAVLIB LIB(*NONSYS) ACCPTH(*YES)

d. SAVDLO DLO(*ALL) FLR(*ANY)

e. STRSBS SBSD(controlling-subsystem)

If you want to be notified when the subsystems are ended, change the QSYSOPR message queue by typing the following and pressing the Enter key.

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
```

Messages are sent to the QSYSOPR message queue indicating when the subsystems have ended and the system is in a restricted state.

10. Continue loading tapes when the system sends a message asking you to load the next volume.

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).

2. Determine the next library to be restored beyond the failure by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
          OUTPUT(*PRINT)
   ```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.

3. Load a new tape and type the following:

   ```
   SAVLIB LIB(*NONSYS) STRLIB(library-name)
   ```

   Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

Method 2. Using the Save commands

If you do not want to use option 21, you can do the following steps from the command line of a menu:

1. Sign on the system as QSECOFR.

2. Verify that no users are on the system and that no batch jobs are running:

   ```
   WRKACTJOB
   ```

3. Display the system log QHST to verify it is up to date:

   ```
   DSPLOG  LOG(QHST)
   ```

   Displaying the QHST log automatically brings it up to date.

4. Display all copies of the system log:

   ```
   WRKF FILE(QSYS/QHST*)
   ```

   Look at the list to verify that you saved all copies of the log that will be needed later.

5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.

6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```

7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)
   ```

   Keep this list with your backup log or your save system tapes for future reference.

8. Change the QSYSOPR message queue:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK)  SEV(60)
   ```

9. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Messages are sent to the QSYSOPR message queue indicating that the subsystems have ended and the system is in a restricted state.

10. Load the first tape and make the tape device ready.

11. Save the system:

    ```
    SAVSYS DEV(TAP01) ENDOPT(*LEAVE)
    ```

12. When a message similar to the following appears, load the next tape or make the device ready, and then enter R.

    ```
    Device was not ready or next volume was not loaded (C R)
    ```

13. Save all user and IBM libraries:

    ```
    SAVLIB LIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
           ACCPTH(*YES)
    ```

ASP/Disk Recovery

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be restored beyond the failure by by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
          OUTPUT(*PRINT)
   ```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.
3. Load a new tape and type the following:

   ```
   SAVLIB LIB(*NONSYS) STRLIB(library-name)
   ```

   Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

14. Save the documents, folders, and distribution documents:

    ```
    SAVDLO DLO(*ALL) FLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
    ```

15. Start the subsystems:

    ```
    STRSBS SBSD(controlling-subsystem)
    ```

Task 2. Access DST Options

Sign on the DST using the following steps.

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

3. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
   ```

7. When the system has powered down, the IPL or Install the System display appears.

```
                    IPL or Install the System

Select one of the following:

         1. Perform an IPL
         2. Install the operating system
         3. Use Dedicated Service Tools (DST)
         4. Perform automatic installation of the operating system

Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                   © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                    Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage

Selection
    4

F3=Exit          F12=Cancel
```

Task 3. Move the Disk Unit From the System ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                          Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information

Selection
     1

F3=Exit        F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units Display and press the Enter key.

```
                         Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection







Selection
     3

F3=Exit          F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display. The following display is shown.

```
                              Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration






Selection
    7

F3=Exit          F12=Cancel
```

4. Select option 7 (Move Units from one ASP to another) to move the desired unit from the system ASP to user ASP 2. The following display is shown.

   Note: If any user ASP has overflowed, reset the ASP before moving the unit from the system ASP. For more information on how to reset the ASP, see "Resetting an Overflowed User ASP" on page 15-67.

```
                         Move Unit from one ASP to another

Type choices, press Enter.

  Unit to move . . . .  ____
  ASP to move to . . .  ____

                                                  --Protected--  --Unprotected--
 ASP  Unit  Type  Model  Threshold  Overflow       Size  %Used      Size  %Used
  1                            90%    No               0  0.00%      1802 59.74%
         1  9332   400                                 0  0.00%       200 97.84%
         2  9332   400                                 0  0.00%       200 56.62%
         3  9332   400                                 0  0.00%       200 57.70%
         4  9332   400                                 0  0.00%       200 57.15%
         5  9332   400                                 0  0.00%       200 58.34%
         6  9332   400                                 0  0.00%       200 56.12%
         7  9332   200                                 0  0.00%       200 58.95%
         8  9332   200                                 0  0.00%       200 47.36%
         9  9332   200                                 0  0.00%       200 47.59%

 Press Enter to continue.

 F3=Exit   F5=Refresh   F11=Display non-configured units   F12=Cancel
```

5. Select the unit you want moved from the source ASP to the target ASP by typing the unit number in the *Unit to move* prompt and the target ASP number in the *ASP to move to* prompt.
6. Press the Enter key.

   Note: All data in the system ASP is destroyed, unless you are moving a unit that does not have checksum protection from the system ASP that has checksum protection.

   The Confirm Move of Units display is shown.

```
                              Confirm Move of Units

Warning: Moving this unit will destroy all the data in the ASP
from which the unit is being moved.

Press F10 to confirm your choice to move the unit.
Press F12=Cancel to return to change your choice.

ASP being moved from . . . . . . :    ___
Unit to move . . . . . . . . . . :   ____
ASP to move to . . . . . . . . . :    ___



F10=Confirm     F12=Cancel
```

7. To confirm the move of the unit, press F10 (Confirm). To return and change your choice, press F12 (Cancel)
8. Press F3 (Exit) until you return to the Use Dedicated Service Tools menu.

9. Select option 7 (Start a service tool) on the Use Dedicated Service Tools menu and press the Enter key.
10. Select option 8 (Power off the system) on the Start a Service Tool menu. The Confirm Power Off the System display is shown.
11. Press F10 to confirm your choice to power off the system.

Task 4. Restore the Licensed Internal Code

After the system has powered off, continue with the following to recover the PTF index.

1. Ensure the key is in the keylock switch on the control panel.
2. Turn the key to the Manual position.
3. Press the Function Select switch to display **02** in the Function display on the control panel.
4. Press the Enter button on the control panel.
5. Select IPL type D (this specifies that the IPL source comes from tape) by pressing the Function Select switch on the control panel until **D** is shown on the Data display.
6. Press the Enter button on the control panel.
7. For the 9406 system unit, ensure that the power switches for the IPL tape unit and all disk units are in the On position.
8. Find the licensed internal code tape, which is the first volume of the most recent set of save tapes.
9. Place the tape in the tape unit used for the IPL. See the setup manual for the device for more information on loading the tape.

   Note: If your tape unit cannot be loaded when the power is off, continue with the next step. You will be prompted later by an SRC code for the tape.
10. Turn on the power to the system by pushing the Power switch up. The switch returns to center after you push it. The 9402 system unit has a green button labeled Power On.
11. Wait as explained below for the tape unit to power on. See the following explanations:

    Notes:

    a. SRC A100 1938 (Tape not found) may be displayed along with the System Attention light until the tape unit is turned on. Continue with the next step.

    b. SRC A100 1933 or 1934 (Tape not ready) may also be displayed along with the System Attention light. When the tape is ready, continue with the next step. If this SRC is displayed for more than 1 or 2 minutes, call your service representative.

    c. SRC 2507 XXXX appears if manual select IPL intervention is required. If intervention is required, press the Function Select switch until 03 appears. Press the Enter button on the control panel.
12. Ensure that the tape is online or ready. No action is required for tape units that perform this step automatically (such as the tape cartridge unit).
13. Ensure that the console display is turned on.

14. Wait for the yellow System Attention light on the control panel to light up.

    There is a delay while the system loads information from the tape. SRCs showing status are continuously updated on the control panel while processing occurs. This can take from 5 to 20 minutes; the time varies depending on the speed of the tape unit and the processor speed for the specific system model.

    When SRC A6xx 6001 is displayed, the system is prepared to start installing or restoring the licensed internal code on the disk unit containing unit 1. Continue with the next step.

15. Select the correct function code by pressing the Function Select switch on the control panel until the correct function code is displayed. (See the "Before You Begin" section in this topic to determine the correct function code.)

    Warning: Option 24 (Install) is used only to recover from the loss of unit 1 in the system ASP, or to recovery from a disaster. Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

    If you selected function code 24, the System Attention light light may appear in one or two minutes and SRC A6xx 6002 is shown. If you are sure you want to install the licensed internal code, select function code 24 again.

    Notes:

    a. When SRC A6xx 6002 is displayed, option 23 can also be selected. If option 23 is selected, a restore of the licensed internal code is performed (not an install). Data on the disk units will not be lost.

    b. If another SRC appears after A6xx 6001 that is not in the A6xx xxxx format, then the system needs additional attention. Call your service representative.

       If the following SRCs appear after SRC A6xx 6001 is displayed, see Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" for an explanation of these SRCs and the steps to follow.

       - A6xx 6002 Disk unit may contain a valid system
       - A6xx 6003 Disk unit not currently a load source
       - A6xx 6004 Disk unit not currently a load source
       - A6xx 6005 Disk unit not found

16. Press the Enter button on the control panel.

    Note: The system will start displaying status SRCs again, which will be continuously updated to show the status of installing the licensed internal code. An example of a status SRC is D6xx 6201 (stand-alone install operations is running).

17. If the yellow System Attention light is on again, and SRC A6xx 6048 (New tape volume needs to be loaded) appears, the system needs the next tape. The xx tells which volume needs to be loaded. Load the correct tape and make the device ready. The install operation will automatically continue.

    If SRC A6xx 6051 appears, the stand-alone function is requesting the model-unique licensed internal code tape found in the service kit. Unload the current tape from the tape device and load the model-unique licensed internal code tape.

- A6xx 6051 Model-unique licensed internal code tape needs to be loaded
- A6xx 6052 Tape loaded was not the model-unique licensed internal code,

If another SRC A6xx xxxx is displayed, look up the displayed SRC in Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" on page A-1 and follow the instructions. For all other SRCs call your service representative.

18. After the install or restore operation of licensed internal code or the model-unique licensed internal code is complete, the system will automatically perform an IPL.
19. Remove the model-unique licensed internal code tape from the tape unit (if it is loaded) when the IPL or Install the System menu is shown.
20. If you have other restore steps to perform and the model-unique licensed internal code tape was loaded, load the previous save tape.

Task 5. Restore the Operating System

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                        IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

   Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

Recovery

```
                    Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.

















 F12=Cancel
```

3. Press the Enter key.
4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                         Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .  2924















F3=Exit          F12=Cancel
```

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary

language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                    Confirm Language Feature Selection

Language feature . . . . . . . . . . . . . . . . . :   2924

Press Enter to confirm your choice for language feature.
  Installing the system will continue.
Press F12 to return to change your choice for
  language feature.









F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. The following display is shown only if disk units have been attached to the system and are in nonconfigured status.

```
                    Add All Disk Units to the System

Select one of the following:

        1. Add all disk units to the system auxiliary storage pool
        2. Keep the current disk configuration
        3. Perform disk configuration using DST










Selection
    __
```

Option 1 (Add all disk units to the system auxiliary storage pool)

Select this option if you want to add all of the nonconfigured units to the system auxiliary storage pool. Before adding the units to the system, all data stored on the non-configured units is deleted.

Note: Adding units can change the checksum set configuration of the system ASP. You can use option 3 (Perform disk configuration using DST) to calculate the effect of adding units to the system ASP.

Option 2 (Keep the current disk configuration)

Select this option if you plan to add the nonconfigured units to user ASPs or use them as spare units. This option continues the IPL without adding units to the system configuration. The disk units that are in nonconfigured status will remain so.

Option 3 (Perform disk configuration using DST)

This option starts the Dedicated Service Tools (DST). On the DST main menu, select option 4 (Work with Disk Units).

8. Press the Enter key.

   As the disk units are configured, the following display is shown:

```
Formatting additional disk units in progress.
          Seconds running . . . . :  ____
```

   Adding disk units takes several minutes. The time it takes depends on the size of each unit and the ability of the system to do multiple adds at the same time.

9. Status messages are displayed.

   The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                          Running IPL Step

Current IPL step . . . . : Storage Management Recovery
```

After the IPL steps complete, the Install the Operating System menu appears.

```
                     Install the Operating System

Type choices, press Enter.

   Default
     option . . . .  _             1=Take defaults, show no
                                     other installing displays
                                   2=Change installing options

   Date:
     Year . . . . .  _             00-99
     Month  . . . .  _             01-12
     Day  . . . . .  _             01-31

   Time  . . . . . .  _    00-23   HH is hours
                      _    00-59   MM is minutes
                      _    00-59   SS is seconds
```

10. When the Install the Operating System display is shown, use the following information to respond to the prompts.

Default Option

Value	Description
1	Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation. If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.
2	If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system. If you select 2 for *Default option*, the Installing Options display appears.

Date

The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

Time

The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

11. Press the Enter key.

If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages

ASP/Disk Recovery

are for your information only. Continue loading tapes in sequence when messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

If you selected option 2 (Change installing options), the following display is shown.

```
                              Specify Install Options

Type choices, press Enter.

    Restore option . . . . __          1=Restore programs and language
                                         objects from current tape
                                       2=Do not restore programs or
                                         language objects
                                       3=Restore only language objects
                                         from current tape
                                       4=Restore only language objects
                                         from a different tape




    Job and output
      queue options. . . .  1          1=Clear,  2=Keep
```

12. When the Installing Options display appears, use the following information to respond to the prompts.

    Restore Option

Value	Description
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

    Notes:

    a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.
    b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.
    c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.
    d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.
    e. The system entry-point table is created again.
    f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user

decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	**Description**
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                         Specify Restore Options

Type choices, press Enter.

    Restore from tape:

      System values . . . . . 2       1=Restore,  2=Do not restore
      Edit descriptions . . . 2       1=Restore,  2=Do not restore
      Message reply list. . . 2       1=Restore,  2=Do not restore
```

13. Using the following information, respond to the prompts on the Restore Options display.

System Values Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID	Default system code page and character set
QCURSYM	Currency symbol
QDATFMT	Date editing format
QDATSEP	Date separator character
QDECFMT	Decimal data editing format
QKBDTYPE	Default work station keyboard type
QLEAPADJ	Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Value	Description
1=Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

14. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

15. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    `Load the next volume on the tape device (1 2).`

16. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    `Operating system has been installed.  IPL in progress.`

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

17. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).
18. Press the Enter key. Informational messages are displayed.
19. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.
20. When the IPL Options display is shown, respond to the prompts using the following information.

ASP/Disk Recovery

```
                              IPL Options

Type choices, press the Enter key.

  System date . . . . . . . . . . . . . .   07 / 26 / 88   MM / DD / YY
  System time . . . . . . . . . . . . . .   12 : 00 : 00   HH : MM : SS
  Clear job queues . . . . . . . . . . . .  N              Y=Yes,  N=No
  Clear output queues  . . . . . . . . . .  N              Y=Yes,  N=No
  Clear incomplete job logs  . . . . . . .  N              Y=Yes,  N=No
  Start print writers  . . . . . . . . . .  Y              Y=Yes,  N=No
  Start this device only . . . . . . . . .  N              Y=Yes,  N=No


  Set major system options . . . . . . . .  N              Y=Yes,  N=No
  Define or change system at IPL . . . . .  N              Y=Yes,  N=No







Last power-down operation was normal
                                     (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure 15-3. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a Y to start the console device only, or type an N to start all devices.

d. Enter the value for set major system options.

   Type an N for each option to leave them unchanged, or type Y to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to N. If you installed the licensed internal code using option 24 (Install), the default value is set to Y.

21. Press the Enter key.

22. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

```
                Edit Rebuild of Access Paths                RCHAS331
                                                    05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                 ------------Access Paths----------  Unique  Rebuild
Seq   Status     File        Library     Member      Keyed   Time
25__  IPL        FILE234512  LIBRARY111  MBR1234567  No      00:00:56
25__  IPL        FILE234513  LIBRARY111  MBR1234567  No      00:00:56
75__  IPL        FILE234514  LIBRARY111  MBR1234567  Yes     00:00:56
75__  IPL        FILE234515  LIBRARY111  MBR1234567  Yes     00:00:56
88__  IPL        FILE234516  LIBRARY111  MBR1234567  No      00:00:56
99__  AFTIPL     FILE234517  LIBRARY111  MBR1234567  Yes     00:00:56
*OPN  OPEN       FILE126789  L123456789  MBR4567890  Yes     12:34:56
*OPN  OPEN       FILE346789  L123456789  MBR4567890  No      12:34:56
*HLD  HELD       F123336789  L123456789  MBR4567890  No      10:30:06
*HLD  HELD       F123456789  L123456789  MBR4567890  Yes     99:56:01
                                                         More...


F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                    Display Access Path Status

IPL Threshold . . . . . . : 88

          ----------Access Paths----------   Rebuild     Current
Status    File         Library    Member     Build Time  Run Time
RUN       F123456789   L123456789 MBR4567890             00:00:01
JRN       F123456789   L123456789 MBR4567890
JRN       F123456789   L123456789 MBR4567890
JRN       F123456789   L123456789 MBR4567890
JRN       F123456789   L123456789 MBR4567890
JRN       F123456789   L123456789 MBR4567890
JRN       F123456789   L123456789 MBR4567890
SYS       F123456789   L123456789 MBR4567890 12:34:56
SYS       F123456789   L123456789 MBR4567890 12:34:56
IPL       F123456789   L123456789 MBR4567890 12:34:56
                                                          More...




F3=Exit and continue IPL   F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN—indicates that the access path is being rebuilt.
- SYS—indicates the access path is a system access path and is waiting to be rebuilt.
- JRN—indicates that the access path is being recovered from its associated journal.
- IPL—indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

23. Press the Enter key to continue.
24. Ensure the keylock switch is in the Normal position.
25. This completes the restore operation for the operating system if you have no other restore steps to perform.
26. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

After the install process is complete, the command entry display or the Main menu appears. You can proceed with the next step.

Task 6. Restore the Remaining Objects to the System

There are two methods to restore the remaining objects to the system, depending on your system configuration:

- Method 1 is used to restore objects if no user ASPs exist, or if you want to avoid a more complicated set of restore steps. This method is also used to restore the system ASP and recover journals, journal receivers, or save files in user ASPs. All user data that was previously saved is restored.
- Method 2 is used to restore the system ASP, and recover libraries and their associated objects in user ASPs.

Select the method to use and continue with the following steps.

1. Type the following to reclaim storage:

   ```
   RCLSTG
   ```

   If you had journals, journal receivers, or save files (whose library is in the system ASP) stored in user ASPs when you began this procedure, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile.

If you had libraries and associated objects stored in a user ASP, this command will restore their addressability and transfers ownership of the objects to QDFTOWN user profile.

If objects are damaged in the ASP, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile. This happens whether you have isolated objects or an entire library in an ASP.

2. Restore user profiles from the correct save tape file (label QFILEUPR).

   Note: Use the tapes from the most recent complete save operation (SAVSYS). If a SAVSECDTA command has been run since the last complete save operation, use the SAVSECDTA tapes.

   If the SAVSYS tape is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
   ```

   If the save security data tape (SAVSECDTA) is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

3. Restore the device configuration objects from your most recent SAVSYS tapes. Type the following:

   ```
   RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

4. It is recommended that you save library QRCL and its contents to create a backup copy of the user ASPs containing isolated objects for use in case another failure should occur. This copy may be needed in the following restore steps, depending how you chose to do the restore operation.

   Load a scratch tape, type the following and press the Enter Key.

   ```
   SAVLIB LIB(QRCL) DEV(TAP01) ENDOPT(*UNLOAD)
   ```

5. Restore the objects to the system in **one** of the following ways:

Method 1. Recovery When No Objects Exist in User ASPs or Old Type ASPs Exist

If you had no objects stored in user ASPs or had journals, journal receivers and save files in user ASPs (where the library for the objects are in the system ASP) at the start of this procedure, or if you want to simplify the restore process at the expense of going through restoring objects that are still intact in the user ASPs on your system, perform the following steps:

1. Delete library QRCL by typing the following and pressing the Enter key.

   ```
   DLTLIB LIB(QRCL)
   ```

2. Restore the IBM and user libraries:

   Type the following and press the Enter key,

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01)  ENDOPT(*UNLOAD)
   ```

3. If you have document library objects to restore, load the SAVDLO tape, and then type the following:

   ```
   RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
           SEQNBR(beginning-number ending-number)
   ```

4. Grant all private object authorities that existed when the system was saved by typing the following and pressing the Enter key. No media is required.

   ```
   RSTAUT
   ```

Method 2. Recovery of Objects and Libraries Existing in User ASPs

After the RCLSTG command is run, the addressability of libraries and objects in the user ASP is restored.

1. Restore the individual libraries to the system ASP from your save tapes.

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) OMITLIB(user-ASP-library-name)
   ```

   Do not restore libraries from the SAVLIB tapes that are in user ASPs. Specify the OMITLIB parameter to exclude up to 50 libraries that exist in the user ASPs. Journals should be restored before the journaled files, or journaling is not resumed for the restored files.

2. Display library QRCL to determine if damaged objects were found.

   ```
   DSPLIB LIB(QRCL)
   ```

   If objects are found in QRCL, do the following:

   a. Delete the damaged objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:

      1) Display the library in the user ASP:

         ```
         WRKLIB LIB(library-name)
         ```

      2) Type a **12** (Work with objects) in the *Opt* column and press the Enter key.

      3) Find the objects to be deleted in the *Object* column.

      4) Type a **4** (Delete) in the *Opt* column for each object you want to delete.

      5) Press the Enter key.

   b. After the objects in the ASP are deleted, move the objects in library QRCL back to their original library.

      ```
      MOVOBJ OBJ(QRCL/object-name) OBJTYPE(*XXXX) TOLIB(library-name)
      ```

3. The RCLSTG command changed ownership of objects existing in user ASPs to QDFTOWN user profile. Transfer ownership of the objects in the user ASP library from QDFTOWN user profile to the correct user profile.

   a. Type the following and press the Enter key:

      ```
      WRKOBJOWN USRPRF(QDFTOWN)
      ```

      The Work with Objects by Owner display is shown.

   b. On the Work with Objects by Owner display, type a **9** in the *Opt* column for each object in the ASP library that you want to change ownership for.

   c. If all the objects will have the same owner, type the following on the command line of the Work with Objects by Owner display. Otherwise, continue with step 3d.

      ```
      NEWOWN(owner-name)
      ```

      Note: If you enter NEWOWN(owner-name) on the command line of the Work with Object by Owner display, you will not have to enter an

ASP/Disk Recovery

owner name in the *New owner* prompt on the Change Object Owner display for each object.

d. Press the Enter key.

e. On the Change Object Owner (CHGOBJOWN) display, type the name of the new owner in the *New owner* prompt and press the Enter key. Repeat this step for all the objects that need the ownership changed.

4. Load the the SAVDLO tape, and then type the following to restore document library objects:

```
RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
        SEQNBR(beginning-number ending-number)
```

5. Type the following to grant all private object authorities that existed when the system was saved:

```
RSTAUT
```

Moving a Disk Unit from One User ASP to Another ASP

Task Overview

You will perform the following steps during this procedure:

1. Display the disk configuration using SST
2. Save the security data
3. Save all objects in the source ASP
4. Delete the objects in the source ASP
5. Access DST
6. Move the unit from the user ASP
7. Restore the objects to the source ASP

Warning: Use caution when changing the allocation of a unit from one ASP to another ASP. When the system changes the allocation of the unit, the data in the ASP is destroyed, except for a special case under checksum protection (see the topic "Moving a Storage Unit Not in a Checksum Set from the System ASP to a User ASP" on page 18-46).

If you move all units out of a particular user ASP, the ASP is automatically deleted. Follow these steps to move a unit to a different ASP:

Task 1. Display the Disk Configuration

Use the SST Display Disk Configuration display to determine the units you will move into the ASP. Make a note about which units will be reassigned.

1. Type the following and press the Enter key.

   STRSST

   The System Service Tools (SST) menu is shown.

```
                      System Service Tools (SST)

 Select one of the following:

      1. Start a service tool
      2. Work with active service tools
      3. Work with disk units
      4. Work with disk data recovery









 Selection
      3

 F3=Exit          F10=Command entry          F12=Cancel
```

ASP/Disk Recovery

2. Select option 3 (Work with disk units) on the System Service Tools menu and press the Enter key.

   The following display is shown.

```
                              Work with Disk Units

 Select one of the following:

     1. Display disk configuration
     2. Display checksum configuration
     3. Calculate checksum configuration
     4. Work with ASP storage threshold
     5. Work with disk unit recovery
     6. Work with disk information
     7. Calculate mirrored capacity









 Selection
       1

 F3=Exit             F12=Cancel
```

3. Select option 1 (Display disk configuration) on the Work with Disk Units display.

```
                              Display Disk Configuration

 Select one of the following:

      1. Display disk configuration status
      2. Display disk configuration capacity
      3. Display disk configuration protection
      4. Display non-configured units











 Selection
      1

 F3=Exit           F12=Cancel
```

4. Select option 1 (Display disk configuration status) on the Display Disk configuration display.

   Note: Disk units on the system are either configured or nonconfigured. Options 1, 2, and 3 on the Display Disk Configuration Status display show information about the configured units. Option 4 shows the non-configured units attached to the system. When you physically attach a unit to the system, it becomes part of the nonconfigured set until you place it in an ASP.

5. Select option 1 (Display disk configuration status) and press the Enter key.

```
                     Display Disk Configuration Status

             Serial
ASP  Unit    Number      Type  Model  Address        Status
 1                                                    Mirrored
        1    10-00A7529  9332   400   0010-0000FFFF  Active
        1    10-00A4936  9332   400   0010-0100FFFF  Active
        2    10-00A7498  9332   400   0010-0300FFFF  Resuming
        2    10-00A7529  9332   400   0010-0001FFFF  Active
        3    10-00A7498  9332   400   0010-0600FFFF  Active
        3    10-00A4936  9332   400   0010-0101FFFF  Active
 2                                                    Unprotected
        8    10-00A7530  9332   400   0020-0701FFFF  Configured
 3                                                    Unprotected
        6    10-00A7530  9332   400   0010-0700FFFF  Configured






Press Enter to continue.

F3=Exit   F5=Refresh    F11=Display non-configured units    F12=Cancel
```

The following fields appear on the Display Disk Configuration Status display:

- *ASP*. The auxiliary storage pool number.
- *Unit*. The number assigned by the system to identify a specific disk unit.
- *Serial Number*. The number assigned by the manufacturer to identify a specific disk unit.
- *Type*. The number assigned by the manufacturer to identify a type of disk unit.
- *Model*. The numbers or letters used to identify the feature level of a specific product type.
- *Address*. Identifies the following:
  - Location of the storage device controller card (columns 1 through 4)
  - Functional controller for the disk unit (columns 5 and 6)
  - Disk unit itself (columns 7 and 8)
  - FFFF (columns 9 through 12)
- *Status*. The valid values for this field are:
  - For ASPs:
    - *Unprotected*. No protection exists for this ASP.
    - *Checksummed*. The units in the ASP are protected by checksum protection if the units are a part of a checksum set.
    - *Mirrored*. All the units in the ASP are protected by mirrored protection.
  - For units in an unprotected ASP.
    - *Configured*.

- For units in an ASP that has checksum protection.
  - *Checksummed*. Indicates that the unit is part of a checksum set.
  - *Configured*. Indicates that the unit is not part of a checksum set.
- For units in a mirrored ASP.
  - *Active*. This unit is capable of having data written to it, or read from it.
  - *Suspended*. This unit is not capable of having data written to it, or read from it. The data on this unit is not current. For example, if the disk needs repair action or has been manually suspended, it would be in a *Suspended* state.
  - *Resuming*. The current data is being copied (or will be copied) to this unit from the other active unit of the mirrored pair.
  - *Unknown*. The system configuration mechanism cannot determine what the valid configuration should be.

6. Write down the serial numbers of the units you are going to use.
7. Return to the Main menu by pressing F3 (Exit) three time.

Task 2. Save the Security Data

Use the Save Security Data (SAVSECDTA) command to save the security data. By entering this command now, you can save all private object authorities. If you do not do this now, you must later manually restore private authorities to every object (with the GRTOBJAUT command) that resides on the user ASP containing the units to be reassigned.

To save the security data, do the following:

1. To change the system operator message queue so all messages will appear on the display, type the following and press the Enter key.

   ```
   CHGMSGQ QSYSOPR *BREAK SEV(60)
   ```

2. Load the first tape, and make the tape device ready.
3. Save the security data:

   If you are saving distribution (mail) objects, type the following and press the Enter key.

   ```
   SAVSECDTA DEV(TAP01) MAIL(*YES)
   ```

   If one or more tapes have not been initialized, a message similar to the following may appear:

   ```
   Volume on device TAP01 wrong type (C INZ R)
   ```

   Enter INZ and press the Enter key to initialize the tape.

4. When the following message appears, load the next tape, make the device ready, and then enter R.

   ```
   Device was not ready or next volume not loaded (C R)
   ```

Task 3. Save All Objects in the Source User ASP

1. Save all objects contained in the source user ASP with the appropriate command:

```
SAVLIB LIB(library-name) DEV(TAP01) LABEL(label-name)
```

```
SAVOBJ OBJ(object-name) LIB(library-name) DEV(TAP01) OBJTYPE(*JRN)
       VOL(*MOUNTED) ENDOPT(*LEAVE)
```

```
SAVOBJ OBJ(object-name) LIB(library-name) DEV(TAP01) OBJTYPE(*FILE)
       VOL(*MOUNTED) ENDOPT(*UNLOAD)
```

   If the objects you save include one or more journals, you may want to save all database files associated with those journals. This step makes it possible to reestablish journaling by deleting the old files and restoring the saved files.

When the save operation is complete, remove the tape from the tape unit.

Task 4. Delete the Objects in the Source User ASP

Delete all objects in the user ASP using the appropriate delete commands before you move the unit. This avoids having any pointers in the system ASP to objects that are destroyed in the user ASP, and having partial objects in the system ASP.

1. Delete the objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:
   a. Display the library in the user ASP:

      ```
      WRKLIB LIB(library-name)
      ```

   b. Type a **12** (Work with objects) in the *Opt* column and press the Enter key.
   c. Find the objects to be deleted in the *Object* column.
   d. Type a **4** (Delete) in the *Opt* column for each object you want to delete.
   e. Press the Enter key.

Task 5. Access DST Options

Sign on to DST using the following steps:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

3. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
   ```

7. When the system has powered down, the IPL or Install the System display appears.

```
                              IPL or Install the System

Select one of the following:

           1. Perform an IPL
           2. Install the operating system
           3. Use Dedicated Service Tools (DST)
           4. Perform automatic installation of the operating system









Selection
     3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                   © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                         Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                    Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage

Selection
     4

F3=Exit            F12=Cancel
```

Task 6. Move the Unit from the User ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tool menu and press the Enter key.

```
                         Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information

Selection
     1

F3=Exit          F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

ASP/Disk Recovery

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection








Selection
     3

F3=Exit         F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Configuration display and press the Enter key.

```
                         Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration







Selection
    7

F3=Exit         F12=Cancel
```

4. Select option 7 (Move unit from one ASP to another) on the Work with ASP Configuration menu. The following display is shown.

```
                 Move Unit from one ASP to another

Type choices, press Enter.

  Unit to move . . . .  ____
  ASP to move to . . .  ____

                                             --Protected--  --Unprotected--
 ASP  Unit  Type  Model  Threshold  Overflow   Size  %Used     Size  %Used
  1                            90%  No            0  0.00%     1802 59.74%
         1  9332   400                            0  0.00%      200 97.84%
         2  9332   400                            0  0.00%      200 56.62%
         3  9332   400                            0  0.00%      200 57.70%
         4  9332   400                            0  0.00%      200 57.15%
         5  9332   400                            0  0.00%      200 58.34%
         6  9332   400                            0  0.00%      200 56.12%
  2                            88%  No            0  0.00%      480 80.00%
         7  9332   200                            0  0.00%      200 58.95%
         8  9332   200                            0  0.00%      200 47.36%
         9  9332   200                            0  0.00%      200 47.59%

 Press Enter to continue.

 F3=Exit   F11=Display configuration units    F12=Cancel
```

5. Select the desired unit to move from the source ASP to the target ASP by entering a unit in the *Unit to move* prompt and the ASP number in the *ASP to move to* prompt.
6. Press the Enter key.

   Note: All data in the source ASP is destroyed unless you are moving a unit that does not have checksum protection from an ASP that has checksum protection.

   The Confirm Move of Units display is shown.

```
                         Confirm Move of Units

Warning: Moving this unit will destroy all the data in the ASP
from which the unit is being moved.

Press F10 to confirm your choice to move the units.
Press F12=Cancel to return to change your choice.

ASP being moved from . . . . . . :    ___
Unit to move . . . . . . . . . . :   ____
ASP to move to . . . . . . . . . :    ___



F10=Confirm      F12=Cancel
```

7. To confirm the move of the unit, press F10 (Confirm). Press F12 (Cancel) to return and change your choice.
8. You can display the new configuration by using the Display Disk Configuration Status display.
9. Press F12 until you return to the the IPL or Install the System menu.
10. Select option 1 (IPL the system) on the IPL or Install the System menu. After IPL processing is complete, the command entry display appears and you can proceed to the next step.

Task 7. Restore the Objects to the Source User ASP

1. End the subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

2. When the subsystems have ended, load the correct volume of the SAVSECDTA tapes.
3. Type the following to restore the user profiles:

   ```
   RSTUSRPRF USRPRF(*ALL) DEV(TAP01)
   ```

4. Load the correct volume of the RSTLIB or RSTOBJ tapes.
5. Use the appropriate restore command to restore all objects that were in the source ASP:

   ```
   RSTLIB SAVLIB(library-name) DEV(TAP01)
   ```

   or:

   ```
   RSTOBJ OBJ(*ALL) SAVLIB(library-name) OBJTYPE(*ALL)
         ENDOPT(*REWIND) MBROPT(*ALL)
   ```

   If an object resided in a user ASP that no longer exists after the object is saved, the object is not restored unless you specify an existing ASP or the default value of *SAVASP for the RSTASP parameter on the appropriate Restore command.

6. Restore private authorities:

   ```
   RSTAUT
   ```

   If any objects were restored into a different library than the one from which they were saved, you must manually grant private authority for the restored objects with the GRTOBJAUT command.

   Notes:

   a. If the journals are in the system ASP, create a new receiver in the user ASP, and run the CHGJRN command to attach the new journal.

   b. If the user ASP contained journals, and you did not delete the old files and then restore them from the media, create a new journal receiver and the journal in the user ASP. Then use the Start Journal Physical Files (STRJRNPF) and the Start Journal Access Paths (STRJRNAP) commands to start journaling again.

7. This completes the steps to move a unit to a different ASP.

Resetting an Overflowed User ASP

If a user ASP exceeds its storage capacity, data for objects in the user ASP overflows into the system ASP. Consequently, the recovery benefits of separating auxiliary storage into the system and user ASPs are no longer guaranteed.

You can monitor user ASP storage use to avoid overflowing storage capacity by setting the storage threshold value for the user ASP. When the ASP storage usage goes past the threshold, the system operator receives messages warning of an impending user ASP overflow. If the user ASP overflows, it will be necessary to delete the objects in the user ASP, reset the overflow status, and then restore objects back to it.

Avoid restoring all saved objects to the same ASP or data may overflow again. Consider restoring some of the objects to another user ASP by using the RSTASP parameter on the restore commands. You may also want to consider adding an additional unit to the user ASP that overflowed.

If you are using journaling and have journals or journal receivers in the overflowed user ASP, see "Moving a Journal From an Overflowed User ASP to a Different ASP" on page 15-115 or "Moving Journal Receivers From an Overflowed User ASP to a Different ASP" on page 15-115 for the procedures to restore the journals and journal receivers and start journaling again.

To reset the overflow status of the user ASP and restore objects into it:

Task 1. Verify the Current Storage

1. Start SST by typing the following:

   STRSST

   Press the Enter key. The following display is shown.

```
                      System Service Tools (SST)

Select one of the following:

     1. Start a service tool
     2. Work with active service tools
     3. Work with disk units
     4. Work with disk data recovery









Selection
     3

F3=Exit          F10=Command entry          F12=Cancel
```

2. Select option 3 (Work with Disk Units) and press the Enter key. The following display is shown.

```
                              Work with Disk Units

 Select one of the following:

      1. Display disk configuration
      2. Display checksum configuration
      3. Calculate checksum configuration
      4. Work with ASP threshold
      5. Work with disk unit recovery
      6. Work with disk unit information
      7. Calculate mirrored capacity









 Selection
      1

 F3=Exit          F12=Cancel
```

3. Select option 1 (Display disk configuration) on the Work with Disk Units display and press the Enter key. The following display is shown.

```
                           Display Disk Configuration

 Select one of the following:

      1. Display disk configuration status
      2. Display disk configuration capacity
      3. Display disk configuration protection
      4. Display non-configured units










 Selection
      2

 F3=Exit          F12=Cancel
```

4. Select option 2 (Display disk configuration capacity) and press the Enter key.

```
                    Display Disk Configuration Capacity

                                                     --Protected--  --Unprotected--
ASP  Unit  Type  Model  Threshold  Overflow     Size  %Used        Size  %Used
 1                          90%       No           0  0.00%         801 67.33%
        1  9332   400                              0  0.00%         200 97.84%
        2  9332   400                              0  0.00%         200 56.62%
        3  9332   400                              0  0.00%         200 57.70%
        4  9332   400                              0  0.00%         200 57.15%
 2                          90%       Yes          0  0.00%         600 57.80
        5  9332   400                              0  0.00%         200 58.34%
        6  9332   400                              0  0.00%         200 56.12%
        7  9332   200                              0  0.00%         200 58.95%
 3                          90%       No           0  0.00%         400 47.48%
        8  9332   200                              0  0.00%         200 47.36%
        9  9332   200                              0  0.00%         200 47.59%

Press Enter to continue.

F3=Exit   F5=Refresh   F11=Display non-configured units    F12=Cancel
```

Figure 15-4. Display Disk Configuration Capacity Display

The following information is used to calculate storage requirements:

- *ASP.* The auxiliary storage pool number.
- *Unit.* The number assigned by the system to identify the disk unit.
- *Type.* The disk unit type assigned by the manufacturer.
- *Model.* The disk model that identifies the feature level for a specific type of disk unit.
- *Threshold.* The current threshold. The system notifies you when this percentage of storage is full.
- *Overflow.* Indicates if this ASP is overflowed.
- *Protected Size.* Amount of storage in the ASP that is protected by checksum or mirrored protection.
- *Protected % Used.* Percent of protected size that is used.
- *Unprotected Size.* Amount of storage in the ASP that is unprotected.
- *Unprotected % Used.* Percent of unprotected size that is used.

Task 2. Save the Security Data

Use the Save Security Data (SAVSECDTA) command to save the security data.

By entering this command now, you can save all private object authorities. If you do not do this now, you must later manually restore private authorities to every object (with the EDTOBJAUT command) that resides on the user ASP containing the units to be reassigned.

To save the security data, do the following:

1. To change the system operator message queue so all messages will appear on the display, type the following and press the Enter key.

```
CHGMSGQ QSYSOPR *BREAK SEV(60)
```

2. Load the first tape, and make the tape device ready.
3. Save the security data:

   If you are saving distribution (mail) objects, type the following and press the Enter key.

   ```
   SAVSECDTA DEV(TAP01) MAIL(*YES)
   ```

   If one or more tapes have not been initialized, a message similar to the following may appear:

   ```
   Volume on device TAP01 wrong type (C INZ R)
   ```

   Enter INZ and press the Enter key to initialize the tape.
4. When the following message appears, load the next tape, make the device ready, and then enter R.

   ```
   Device was not ready or next volume not loaded (C R)
   ```

Task 3. Save the Objects in the User ASP

1. Save all objects contained in the user ASP with the appropriate command:

   ```
   SAVLIB LIB(user-ASP-library-name) DEV(TAP01) LABEL(label-name)

   SAVOBJ OBJ(object-name) LIB(library-name) DEV(TAP01) OBJTYPE(*JRN)
          VOL(*MOUNTED) ENDOPT(*LEAVE)

   SAVOBJ OBJ(object-name) LIB(library-name) DEV(TAP01) OBJTYPE(*FILE)
          VOL(*MOUNTED) ENDOPT(*UNLOAD)
   ```

   If the objects you save include one or more journals, you may want to save all database files associated with those journals. This step makes it possible to reestablish journaling by deleting the old files and restoring the saved files.

Task 4. Delete the Objects in the ASP

Delete all objects from the user ASP that overflowed using the appropriate delete commands before resetting the ASP. This avoids having any pointers in the system ASP to objects that are destroyed in the user ASP, and having partial objects in the system ASP.

1. Display the library in the user ASP:

   ```
   WRKLIB LIB(library-name)
   ```

2. Type a **12** (Work with objects) in the *Opt* column and press the Enter key.
3. Find the objects to be deleted in the *Object* column.
4. Type a **4** (Delete) in the *Opt* column for each object you want to delete.
5. Press the Enter key.

Task 5. Access DST Options

Start the DST Work with Disk Units function:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

3. End all subsystems:

```
ENDSBS SBS(*ALL) OPTION(*IMMED)
```

Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.

4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

```
PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
```

7. When the system has powered down, the IPL or Install the System display appears.

```
                      IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system

Selection
     3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                   © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                 Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

Recovery

```
                         Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage





Selection
    4

F3=Exit          F12=Cancel
```

Task 6. Reset the Overflowed ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu and press the Enter key.

```
                              Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information






Selection
    1

F3=Exit        F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display and press the Enter key.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection

Selection
     3

F3=Exit          F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Configuration display and press the Enter key.

```
                         Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration

Selection
    5

F3=Exit          F12=Cancel
```

4. Select option 5 (Reset overflowed user ASPs) on the Work with ASP Configuration display and press the Enter key. The following display is shown.

```
                         Reset Overflowed User ASP

Type option, press Enter
   1=Reset

                                         --Protected--  --Unprotected--
Option  ASP  Threshold  Overflow   Size  %Used      Size  %Used
         1       90%    No          600  77.84%        0  0.00%
  1      2       90%    Yes         200  67.87%        0  0.00%
         3       90%    No          200  70.31%        0  0.00%


Press Enter to continue.

F3=Exit   F11=Display configuration status    F12=Cancel
```

5. Type a **1** in the *Option* column of the ASP you want to reset and press the Enter key. The Confirm Reset of User ASPs display is shown.

```
                         Confirm Reset of ASPs

Warning: Resetting a user ASP will cause all the data in that
ASP to be destroyed.

Press F10 to confirm your choice(s) for 1=Reset ASP.
Press F12=Cancel to return to change your choice(s).

                                         --Protected--  --Unprotected--
Option  ASP  Threshold  Overflow   Size  %Used      Size  %Used
  1      2       90%    Yes         200  67.87%      200  0.53%


 F10=Confirm          F12=Cancel
```

6. Press F10 (Confirm) or F12 (Cancel)
7. Verify the configuration using the Display Disk Configuration displays.
8. Exit from the Work with Disk Units function.

Task 7. Restore Objects to the User ASP

1. From the IPL or Install the System menu, select option 1 (Perform an IPL)

   After IPL processing is complete, the command entry screen will appear and you can proceed to the next step.

2. Load the correct volume of the SAVSECDTA tapes.

3. Restore the user profiles.

   ```
   RSTUSRPRF USRPRF(*ALL)
   ```

4. Load the correct volume of the RSTLIB or RSTOBJ tapes.

5. Use the appropriate restore command to restore the desired objects to the ASP:

   ```
   RSTLIB SAVLIB(library-name) DEV(TAP01)
   ```

   Or:

   ```
   RSTOBJ OBJ(*ALL) SAVLIB(library-name) OBJTYPE(*ALL)
         ENDOPT(*REWIND) MBROPT(*ALL)
   ```

6. Restore the private authorities:

   ```
   RSTAUT
   ```

   If any objects were restored into a different library than the one from which they were saved, manually grant private authority for the restored objects with the EDTOBJAUT command.

   Notes:

   a. If the journals are in the system ASP, create a new receiver in the user ASP, and run the CHGJRN command to attach the new journal.

   b. If the user ASP contained journals, and you did not delete the old files and then restore them from the media, create a new journal receiver and the journal in the user ASP. Then use the Start Journal Physical Files (STRJRNPF) and the Start Journal Access Paths (STRJRNAP) commands to start journaling again.

7. This completes the steps to reset the overflow status of the user ASP and restore objects into it.

Removing a Disk Unit from the System ASP

Task Overview

You will perform the following steps during this procedure:

1. Save the System
2. Access DST
3. Remove the unit from the system ASP
4. Restore the licensed internal code using function code 23.
5. Restore the operating system
6. Restore the remaining objects to the system

Warning: Use caution when removing a unit from an ASP. When the system removes a unit from an ASP, all objects are destroyed unless a unit that does not have checksum protection is being removed from an ASP that has checksum protection (see "Moving a Storage Unit Not in a Checksum Set from the System ASP to a User ASP" on page 18-46.)

Warning: A sufficient number of 2800-001 storage units must be configured to the system ASP (ASP 1) to allow for enough main storage dump space. (See note 3 on page 14-7.)

Before you begin...

The following conditions apply when you remove a unit:

- When removing a disk unit from the system ASP, the licensed internal code must be restored using function code 23 (restore).
- You cannot remove unit 1 from the system ASP. (Unit 1 is reserved for the system licensed internal code.)
- Any overflowed use ASP will be cleared if a unit is removed from the system ASP.
- All data in the ASP from which you are moving the unit is destroyed, except when you remove a unit that is not checksum protected from an ASP that is checksum protected.

In the following example, a unit currently in the system ASP is being removed.

Do the following tasks to remove a disk unit:

Task 1. Save the Entire System

If you are removing a unit from the system ASP, save the entire system with the appropriate save commands.

Notes:

1. When saving the entire system, it is important for you to remember to display the system log (QHST) and move or delete all but the current log. These steps will improve the performance of the save system operation.
2. To significantly reduce the time it takes to restore the system, consider specifying SAVLIB LIB(*NONSYS) ACCPTH(*YES).
3. If your application programs are dependent on spooled output files, you can save them. The system does not directly support saving spooled output files. However, you can save the files by copying them (CPYSPLF command) before you perform a save system operation. For an example of how to save the spooled ouput files, see the topic "Example of Saving Spooled Output Files Using the Copy Spooled File (CPYSPLF) Command" on page 9-22.

There are two methods you can use to save the entire system:

1. Using option 21 (Entire system) on the Save menu allows you to save the entire system without entering the commands.
2. Using the Save commands allows you to save the entire system by entering the commands from the command line.

Method 1. Using Option 21 (Entire system) on the Save Menu

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:

   WRKACTJOB
3. Display the system log QHST to verify it is up to date:

   DSPLOG LOG(QHST)

   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:

   WRKF FILE(QSYS/QHST*)

   Look at the list to verify that you saved all copies of the log that will be needed later.
5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.
6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)

   Keep this list with your backup log or your save system tapes for future reference.
8. Go to the Save menu:

   GO SAVE

   The Save menu is shown.

```
SAVE                                  Save
                                                     System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Changed objects only
     7. Licensed programs
     8. Security data
     9. Storage

    20. All libraries other than system library
    21. Entire system
    22. All IBM libraries other than system library
                                                               More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
     (C) COPYRIGHT IBM CORP. 1980, 1991.
```

Recovery

9. Select option 21 (Entire system) from the Save menu and press the Enter key.

```
                         Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .   TAP01       Names
                                   ______
                                   ______
                                   ______

  Prompt for commands  . . . . .   Y           Y=Yes, N=No

  Check for active files . . . .   Y           Y=Yes, N=No
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device when the first tape is full.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting (for unattended save operations) and uses the default values.

Check for active files

Allows you to specify whether or not you want to check for active files. If you specify Y=Yes, the system sends a message when active files are encountered. You can end the checking process or clear the existing files and continue. If N=No is specified, all active files encountered during the save are cleared.

Option 21 will guide you through the following if you selected Y on the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS SBS(*ALL) OPTION(*IMMED)

b. SAVSYS

c. SAVLIB LIB(*NONSYS) ACCPTH(*YES)

d. SAVDLO DLO(*ALL) FLR(*ANY)

e. STRSBS SBSD(controlling-subsystem)

If you want to be notified when the subsystems are ended, change the QSYSOPR message queue by typing the following and pressing the Enter key.

`CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)`

Messages are sent to the QSYSOPR message queue indicating when the subsystems have ended and the system is in a restricted state.

10. Continue loading tapes when the system sends a message asking you to load the next volume.

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be restored beyond the failure by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
           OUTPUT(*PRINT)
   ```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.
3. Load a new tape and type the following:

   ```
   SAVLIB LIB(*NONSYS) STRLIB(library-name)
   ```

   Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

Method 2. Using the Save commands

If you do not want to use option 21, you can do the following steps from the command line of a menu:

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:

   ```
   WRKACTJOB
   ```
3. Display the system log QHST to verify it is up to date:

   ```
   DSPLOG  LOG(QHST)
   ```

   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:

   ```
   WRKF FILE(QSYS/QHST*)
   ```

   Look at the list to verify that you saved all copies of the log that will be needed later.
5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.
6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```
7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)
   ```

Keep this list with your backup log or your save system tapes for future reference.

8. Change the QSYSOPR message queue:

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK)  SEV(60)
```

9. End all subsystems:

```
ENDSBS SBS(*ALL) OPTION(*IMMED)
```

Messages are sent to the QSYSOPR message queue indicating that the subsystems have ended and the system is in a restricted state.

10. Load the first tape and make the tape device ready.
11. Save the system:

```
SAVSYS DEV(TAP01) ENDOPT(*LEAVE)
```

12. When a message similar to the following appears, load the next tape or make the device ready, and then enter R.

```
Device was not ready or next volume was not loaded (C R)
```

13. Save all user and IBM libraries:

```
SAVLIB LIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
       ACCPTH(*YES)
```

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be restored beyond the failure by by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

```
DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
        OUTPUT(*PRINT)
```

The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.

3. Load a new tape and type the following:

```
SAVLIB LIB(*NONSYS) STRLIB(library-name)
```

Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

14. Save the documents, folders, and distribution documents:

```
SAVDLO DLO(*ALL) FLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
```

15. Start the subsystems:

```
STRSBS SBSD(controlling-subsystem)
```

Task 2. Access DST Options

Sign on to DST using the following steps:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)

3. End all subsystems:

   ENDSBS SBS(*ALL) OPTION(*IMMED)

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.

4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)

7. When the system has powered down, the IPL or Install the System display appears.

```
                          IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system










Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                   © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

ASP/Disk Recovery

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                    Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage




Selection
    4

F3=Exit            F12=Cancel
```

Task 3. Remove the Unit From the System ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                         Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information









Selection
     1

F3=Exit         F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units Display and press the Enter key.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection








Selection
     3

F3=Exit         F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display; the following display is shown.

```
                         Work with ASP Configuration

 Select one of the following:

      1. Display disk configuration capacity
      2. Create User ASP
      3. Delete user ASP
      4. Add units to existing ASP
      5. Reset overflowed user ASPs
      6. Change ASP storage threshold
      7. Move units from one ASP to another
      8. Remove unit from configuration









 Selection
     8

 F3=Exit         F12=Cancel
```

4. Select option 8 (Remove unit from configuration) to remove the desired unit. The following display is shown.

   Note: If any user ASP has overflowed, reset the ASP before removing the unit from the system ASP. For more information on how to reset the ASP, see "Resetting an Overflowed User ASP" on page 15-67.

```
                     Remove Units from Configuration

Type options, press Enter.
   4=Remove unit from the configuration

                   Serial
 OPT  Unit  ASP  Number  Type  Model  Address        Status
         1    1          9332   400   0010-0000FFFF  Configured
         2    1          9332   400   0010-0100FFFF  Configured
         3    1          9332   400   0010-0300FFFF  Configured
         4    1          9332   400   0010-0001FFFF  Configured
         5    1          9332   400   0010-0600FFFF  Configured
         6    1          9332   400   0010-0101FFFF  Configured
         7    2          9332   200   0010-0201FFFF  Configured
         8    3          9332   200   0020-0701FFFF  Configured
         9    4          9332   200   0010-0700FFFF  Configured

                                                               More...
 Press Enter to continue.

 F3=Exit   F5=Refresh   F11=Display non-configured units   F12=Cancel
```

5. Select the unit you want removed from the configuration by typing a **4** in the *OPT* column and pressing the Enter key.

   Note: More than one unit can be removed at the same time. Confirmation depends on:

   - Removing a unit not in a checksum set from an ASP with checksum protection does not clear the ASP.

- Removing a unit from a checksum set will clear the ASP and stop checksum protection.
- Clearing the system ASP while user ASPs are configured will clear all overflowed user ASPs.
- Removing a unit from a mirrored or unprotected ASP will clear the ASP.

The Confirm Remove of Units display is shown.

```
                        Confirm Remove of Units

Warning: Removing a unit from an ASP will destroy all the data
from that ASP.

Press F10 to confirm your choice to remove the unit.
Press F12=Cancel to return and change your choice.
                  Serial
 Opt  Unit  ASP  Number  Type  Model  Address        Status
        4    1           9332   400   0010-0001FFFF  Configured
        5    1           9332   400   0010-0600FFFF  Configured









                                                              Bottom


 F10=Confirm   F12=Cancel
```

6. To confirm the remove of the unit, press F10 (Confirm). Press F12 (Cancel) to return and change your choice.

   All data in the ASP is destroyed unless you are removing a unit that does not have checksum protection from an ASP that has checksum protection.
7. Press F3 (Exit) until you return to the Use Dedicated Service Tools menu.
8. Select option 7 (Start a service tool) on the Use Dedicated Service Tools menu and press the Enter key.
9. Select option 8 (Power off the system) on the Start a Service Tool menu.

Task 4. Restore the Licensed Internal Code

1. Restore the licensed internal code to recover the PTF index if the unit was removed from the system ASP by doing the following:
2. Ensure the key is in the keylock switch on the control panel.
3. Turn the key to the Manual position.
4. Press the Function Select switch to display **02** in the Function display on the control panel.
5. Press the Enter button on the control panel.
6. Select IPL type D (this specifies that the IPL source comes from tape) by pressing the Function Select switch on the control panel until **D** is shown on the Data display.
7. Press the Enter button on the control panel.

8. For the 9406 system unit, ensure that the power switches for the IPL tape unit and all disk units are in the On position.
9. Find the licensed internal code tape, which is the first volume of the most recent set of save tapes.
10. Place the tape in the tape unit used for the IPL. See the setup manual for the device for more information on loading the tape.

    Note: If your tape unit cannot be loaded when the power is off, continue with the next step. You will be prompted later by an SRC code for the tape.
11. Turn on the power to the system by pushing the Power switch up. The switch returns to center after you push it. The 9402 system unit has a green button labeled Power On.
12. Wait as explained below for the tape unit to power on. See the following explanations:

    Notes:

    a. SRC A100 1938 (Tape not found) may be displayed along with the System Attention light until the tape unit is turned on. Continue with the next step.

    b. SRC A100 1933 or 1934 (Tape not ready) may also be displayed along with the System Attention light. When the tape is ready, continue with the next step. If this SRC is displayed for more than 1 or 2 minutes, call your service representative.

    c. SRC 2507 XXXX appears if manual select IPL intervention is required. If intervention is required, press the Function Select switch until 03 appears. Press the Enter button on the control panel.
13. Ensure that the tape is online or ready. No action is required for tape units that perform this step automatically (such as the tape cartridge unit).
14. Ensure that the console display is turned on.
15. Wait for the yellow System Attention light on the control panel to light up.

    There is a delay while the system loads information from the tape. SRCs showing status are continuously updated on the control panel while processing occurs. This can take from 5 to 20 minutes; the time varies depending on the speed of the tape unit and the processor speed for the specific system model.

    When SRC A6xx 6001 is displayed, the system is prepared to start installing or restoring the licensed internal code on the disk unit containing unit 1. Continue with the next step.
16. Select the correct function code by pressing the Function Select switch on the control panel until the correct function code is displayed. (See the "Before You Begin" section in this topic to determine the correct function code.)

    Warning: Option 24 (Install) is used only to recover from the loss of unit 1 in the system ASP, or to recovery from a disaster. Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

If you selected function code 24, the System Attention light light may appear in one or two minutes and SRC A6xx 6002 is shown. If you are sure you want to install the licensed internal code, select function code 24 again.

Notes:

a. When SRC A6xx 6002 is displayed, option 23 can also be selected. If option 23 is selected, a restore of the licensed internal code is performed (not an install). Data on the disk units will not be lost.

b. If another SRC appears after A6xx 6001 that is not in the A6xx xxxx format, then the system needs additional attention. Call your service representative.

   If the following SRCs appear after SRC A6xx 6001 is displayed, see Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" for an explanation of these SRCs and the steps to follow.

   - A6xx 6002 Disk unit may contain a valid system
   - A6xx 6003 Disk unit not currently a load source
   - A6xx 6004 Disk unit not currently a load source
   - A6xx 6005 Disk unit not found

17. Press the Enter button on the control panel.

    Note: The system will start displaying status SRCs again, which will be continuously updated to show the status of installing the licensed internal code. An example of a status SRC is D6xx 6201 (stand-alone install operations is running).

18. If the yellow System Attention light is on again, and SRC A6xx 6048 (New tape volume needs to be loaded) appears, the system needs the next tape. The xx tells which volume needs to be loaded. Load the correct tape and make the device ready. The install operation will automatically continue.

    If SRC A6xx 6051 appears, the stand-alone function is requesting the model-unique licensed internal code tape found in the service kit. Unload the current tape from the tape device and load the model-unique licensed internal code tape.

    - A6xx 6051 Model-unique licensed internal code tape needs to be loaded
    - A6xx 6052 Tape loaded was not the model-unique licensed internal code,

    If another SRC A6xx xxxx is displayed, look up the displayed SRC in Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" on page A-1 and follow the instructions. For all other SRCs call your service representative.

19. After the install or restore operation of licensed internal code or the model-unique licensed internal code is complete, the system will automatically perform an IPL.

20. Remove the model-unique licensed internal code tape from the tape unit (if it is loaded) when the IPL or Install the System menu is shown.

21. If you have other restore steps to perform and the model-unique licensed internal code tape was loaded, load the previous save tape.

Task 5. Restore the Operating System

Load the SAVSYS media in the tape unit and continue with the following:

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                          IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

   Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

```
                      Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.






 F12=Cancel
```

3. Press the Enter key.
4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                    Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .  2924

F3=Exit         F12=Cancel
```

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                 Confirm Language Feature Selection

 Language feature  . . . . . . . . . . . . . . . . . :   2924

 Press Enter to confirm your choice for language feature.
   Installing the system will continue.
 Press F12 to return to change your choice for
   language feature.

 F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. The following display is shown only if disk units have been attached to the system and are in nonconfigured status.

```
                  Add All Disk Units to the System

Select one of the following:

         1. Add all disk units to the system auxiliary storage pool
         2. Keep the current disk configuration
         3. Perform disk configuration using DST

Selection
    __
```

Option 1 (Add all disk units to the system auxiliary storage pool)

Select this option if you want to add all of the nonconfigured units to the system auxiliary storage pool. Before adding the units to the system, all data stored on the non-configured units is deleted.

Note: Adding units can change the checksum set configuration of the system ASP. You can use option 3 (Perform disk configuration using DST) to calculate the effect of adding units to the system ASP.

Option 2 (Keep the current disk configuration)

Select this option if you plan to add the nonconfigured units to user ASPs or use them as spare units. This option continues the IPL without adding units to the system configuration. The disk units that are in nonconfigured status will remain so.

Option 3 (Perform disk configuration using DST)

This option starts the Dedicated Service Tools (DST). On the DST main menu, select option 4 (Work with Disk Units).

8. Press the Enter key.

   As the disk units are configured, the following display is shown:

```
Formatting additional disk units in progress.
        Seconds running . . . . : ____
```

Adding disk units takes several minutes. The time it takes depends on the size of each unit and the ability of the system to do multiple adds at the same time.

9. Status messages are displayed.

   The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                              Running IPL Step

Current IPL step . . . . : Storage Management Recovery
```

After the IPL steps complete, the Install the Operating System menu appears.

```
                    Install the Operating System

 Type choices, press Enter.

    Default
       option . . . .  _            1=Take defaults, show no
                                      other installing displays
                                    2=Change installing options

    Date:
       Year . . . . .  _            00-99
       Month  . . . .  _            01-12
       Day  . . . . .  _            01-31

    Time  . . . . . .  _    00-23   HH is hours
                       _    00-59   MM is minutes
                       _    00-59   SS is seconds
```

10. When the Install the Operating System display is shown, use the following information to respond to the prompts.

Default Option

Value	Description
1	Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation. If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.
2	If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system. If you select 2 for *Default option*, the Installing Options display appears.

Date

The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

Time

The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

11. Press the Enter key.

If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages are for your information only. Continue loading tapes in sequence when

messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

If you selected option 2 (Change installing options), the following display is shown.

```
                         Specify Install Options

Type choices, press Enter.

    Restore option . . . . _          1=Restore programs and language
                                        objects from current tape
                                      2=Do not restore programs or
                                        language objects
                                      3=Restore only language objects
                                        from current tape
                                      4=Restore only language objects
                                        from a different tape




    Job and output
      queue options. . . . 1          1=Clear,  2=Keep
```

12. When the Installing Options display appears, use the following information to respond to the prompts.

Restore Option

Value	Description
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

Notes:

a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.

b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.

c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.

d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.

e. The system entry-point table is created again.

f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user

ASP/Disk Recovery

decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	Description
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                         Specify Restore Options

Type choices, press Enter.

    Restore from tape:

       System values . . . . . 2       1=Restore,  2=Do not restore
       Edit descriptions . . . 2       1=Restore,  2=Do not restore
       Message reply list. . . 2       1=Restore,  2=Do not restore
```

13. Using the following information, respond to the prompts on the Restore Options display.

System Values Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID	Default system code page and character set
QCURSYM	Currency symbol
QDATFMT	Date editing format
QDATSEP	Date separator character
QDECFMT	Decimal data editing format
QKBDTYPE	Default work station keyboard type
QLEAPADJ	Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Value	Description
1=Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

14. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

15. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    ```
    Load the next volume on the tape device (1 2).
    ```

16. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    ```
    Operating system has been installed.  IPL in progress.
    ```

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

17. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).
18. Press the Enter key. Informational messages are displayed.
19. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.
20. When the IPL Options display is shown, respond to the prompts using the following information.

```
                              IPL Options

Type choices, press the Enter key.

  System date . . . . . . . . . . . . . .   07 / 26 / 88    MM / DD / YY
  System time . . . . . . . . . . . . . .   12 : 00 : 00    HH : MM : SS
  Clear job queues . . . . . . . . . . . .  N               Y=Yes,  N=No
  Clear output queues  . . . . . . . . . .  N               Y=Yes,  N=No
  Clear incomplete job logs  . . . . . . .  N               Y=Yes,  N=No
  Start print writers  . . . . . . . . . .  Y               Y=Yes,  N=No
  Start this device only . . . . . . . . .  N               Y=Yes,  N=No


  Set major system options . . . . . . . .  N               Y=Yes,  N=No
  Define or change system at IPL . . . . .  N               Y=Yes,  N=No







Last power-down operation was normal
                                     (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure 15-5. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a `Y` to start the console device only, or type an `N` to start all devices.

d. Enter the value for set major system options.

   Type an `N` for each option to leave them unchanged, or type `Y` to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to `N`. If you installed the licensed internal code using option 24 (Install), the default value is set to `Y`.

21. Press the Enter key.

22. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

```
                  Edit Rebuild of Access Paths                  RCHAS331
                                                       05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                  ------------Access Paths----------  Unique  Rebuild
Seq   Status      File        Library     Member      Keyed   Time
25__  IPL         FILE234512  LIBRARY111  MBR1234567  No      00:00:56
25__  IPL         FILE234513  LIBRARY111  MBR1234567  No      00:00:56
75__  IPL         FILE234514  LIBRARY111  MBR1234567  Yes     00:00:56
75__  IPL         FILE234515  LIBRARY111  MBR1234567  Yes     00:00:56
88__  IPL         FILE234516  LIBRARY111  MBR1234567  No      00:00:56
99__  AFTIPL      FILE234517  LIBRARY111  MBR1234567  Yes     00:00:56
*OPN  OPEN        FILE126789  L123456789  MBR4567890  Yes     12:34:56
*OPN  OPEN        FILE346789  L123456789  MBR4567890  No      12:34:56
*HLD  HELD        F123336789  L123456789  MBR4567890  No      10:30:06
*HLD  HELD        F123456789  L123456789  MBR4567890  Yes     99:56:01
                                                            More...

F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                    Display Access Path Status

IPL Threshold . . . . . . : 88

         ----------Access Paths----------   Rebuild     Current
Status   File        Library     Member      Build Time  Run Time
RUN      F123456789  L123456789  MBR4567890              00:00:01
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
SYS      F123456789  L123456789  MBR4567890  12:34:56
SYS      F123456789  L123456789  MBR4567890  12:34:56
IPL      F123456789  L123456789  MBR4567890  12:34:56
                                                          More...

F3=Exit and continue IPL    F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

ASP/Disk Recovery

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN – indicates that the access path is being rebuilt.
- SYS – indicates the access path is a system access path and is waiting to be rebuilt.
- JRN – indicates that the access path is being recovered from its associated journal.
- IPL – indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

23. Press the Enter key to continue.
24. Ensure the keylock switch is in the Normal position.
25. This completes the restore operation for the operating system if you have no other restore steps to perform.
26. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

After the restore process is complete, the continue with the following:

Task 6. Restore the Remaining Objects to the System

There are two methods to restore the remaining objects to the system, depending on your system configuration:

- Method 1 is used to restore objects if no user ASPs exist or if you want to avoid a more complicated set of restore steps. This method is also used to restore the system ASP and recover journals, journal receivers, or save files in user ASPs. All user data that was previously saved is restored.
- Method 2 is used to restore the system ASP and recover libraries and their associated objects in user ASPs.

Select the method to use and continue with the following steps.

1. Type the following to reclaim storage:

   ```
   RCLSTG
   ```

   If you had journals, journal receivers, or save files (whose library is in the system ASP) stored in user ASPs when you began this procedure, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile.

If you had libraries and associated objects stored in a user ASP, this command will restore their addressability and transfers ownership of the objects to QDFTOWN user profile.

If objects are damaged in the ASP, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile. This happens whether you have isolated objects or an entire library in an ASP.

2. Restore user profiles from the correct save tape file (label QFILEUPR).

   Note: Use the tapes from the most recent complete save operation (SAVSYS). If a SAVSECDTA command has been run since the last complete save operation, use the SAVSECDTA tapes.

   If the SAVSYS tape is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
   ```

   If the save security data tape (SAVSECDTA) is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

3. Restore the device configuration objects from your most recent SAVSYS tapes. Type the following:

   ```
   RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

4. It is recommended that you save library QRCL and its contents to create a backup copy of the user ASPs containing isolated objects for use in case another failure should occur. This copy may be needed in the following restore steps, depending how you chose to do the restore operation.

   Load a scratch tape, type the following and press the Enter Key.

   ```
   SAVLIB LIB(QRCL) DEV(TAP01) ENDOPT(*UNLOAD)
   ```

5. Restore the objects to the system in **one** of the following ways:

Method 1. Recovery When No Objects Exist in User ASPs or Old Type ASPs Exist

If you had no objects stored in user ASPs or had journals, journal receivers and save files in user ASPs (where the library for the objects are in the system ASP) at the start of this procedure, or if you want to simplify the restore process at the expense of going through restoring objects that are still intact in the user ASPs on your system, perform the following steps:

1. Delete library QRCL by typing the following and pressing the Enter key.

   ```
   DLTLIB LIB(QRCL)
   ```

2. Restore the IBM and user libraries:

   Type the following and press the Enter key,

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01)  ENDOPT(*UNLOAD)
   ```

3. If you have document library objects to restore, load the SAVDLO tape, and then type the following:

   ```
   RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
            SEQNBR(beginning-number ending-number)
   ```

4. Grant all private object authorities that existed when the system was saved by typing the following and pressing the Enter key. No media is required.

   ```
   RSTAUT
   ```

Method 2. Recovery of Objects and Libraries Existing in User ASPs

After the RCLSTG command is run, the addressability of libraries and objects in the user ASP is restored.

1. Restore the individual libraries to the system ASP from your save tapes.

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) OMITLIB(user-ASP-library-name)
   ```

   Do not restore libraries from the SAVLIB tapes that are in user ASPs. Specify the OMITLIB parameter to exclude up to 50 libraries that exist in the user ASPs. Journals should be restored before the journaled files, or journaling is not resumed for the restored files.

2. Display library QRCL to determine if damaged objects were found.

   ```
   DSPLIB LIB(QRCL)
   ```

   If objects are found in QRCL, do the following:

   a. Delete the damaged objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:

      1) Display the library in the user ASP:

         ```
         WRKLIB LIB(library-name)
         ```

      2) Type a **12** (Work with objects) in the *Opt* column and press the Enter key.

      3) Find the objects to be deleted in the *Object* column.

      4) Type a **4** (Delete) in the *Opt* column for each object you want to delete.

      5) Press the Enter key.

   b. After the objects in the ASP are deleted, move the objects in library QRCL back to their original library.

      ```
      MOVOBJ OBJ(QRCL/object-name) OBJTYPE(*XXXX) TOLIB(library-name)
      ```

3. The RCLSTG command changed ownership of objects existing in user ASPs to QDFTOWN user profile. Transfer ownership of the objects in the user ASP library from QDFTOWN user profile to the correct user profile.

   a. Type the following and press the Enter key:

      ```
      WRKOBJOWN USRPRF(QDFTOWN)
      ```

      The Work with Objects by Owner display is shown.

   b. On the Work with Objects by Owner display, type a **9** in the *Opt* column for each object in the ASP library that you want to change ownership for.

   c. If all the objects will have the same owner, type the following on the command line of the Work with Objects by Owner display. Otherwise, continue with step 3d.

      ```
      NEWOWN(owner-name)
      ```

      Note: If you enter NEWOWN(owner-name) on the command line of the Work with Object by Owner display, you will not have to enter an owner name in the *New owner* prompt on the Change Object Owner display for each object.

d. Press the Enter key.

e. On the Change Object Owner (CHGOBJOWN) display, type the name of the new owner in the *New owner* prompt and press the Enter key. Repeat this step for all the objects that need the ownership changed.

4. Load the the SAVDLO tape, and then type the following to restore document library objects:

```
RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
        SEQNBR(beginning-number ending-number)
```

5. Type the following to grant all private object authorities that existed when the system was saved:

```
RSTAUT
```

Removing a Disk Unit from a User ASP

Task Overview

You will perform the following steps during this task

1. Save the security data
2. Save the objects in the user ASP
3. Access DST
4. Remove the unit from the user ASP
5. Restore the objects to the user ASP

Warning: Use caution when removing a unit from an ASP. When the system removes a unit from an ASP, all objects are destroyed unless a unit that does not have checksum protection is being removed from an ASP that has checksum protection (see the topic "Moving a Storage Unit Not in a Checksum Set from the System ASP to a User ASP" on page 18-46).

In the following example, a unit currently in a user ASP is being removed.

Do the following steps to remove a disk unit:

Task 1. Save the Security Data

Use the Save Security Data (SAVSECDTA) command to save the security data.

By entering this command now, you can save all private object authorities. If you do not do this now, you must later manually restore private authorities to every object (with the EDTOBJAUT command) that resides on the user ASP containing the units to be reassigned.

To save the security data, do the following:

1. To change the system operator message queue so all messages will appear on the display, type the following and press the Enter key.

   ```
   CHGMSGQ QSYSOPR *BREAK SEV(60)
   ```

2. Load the first tape, and make the tape device ready.
3. Save the security data:

   If you are saving distribution (mail) objects, type the following and press the Enter key.

   ```
   SAVSECDTA DEV(TAP01) MAIL(*YES)
   ```

   If one or more tapes have not been initialized, a message similar to the following may appear:

   ```
   Volume on device TAP01 wrong type (C INZ R)
   ```

   Enter `INZ` and press the Enter key to initialize the tape.
4. When the following message appears, load the next tape, make the device ready, and then enter `R`.

   ```
   Device was not ready or next volume not loaded (C R)
   ```

Task 2. Save the Objects in the User ASP

1. Save all objects contained in the user ASP with the appropriate command:

   ```
   SAVLIB LIB(library-name) DEV(TAP01) LABEL(label-name)
   ```

   or

   ```
   SAVOBJ OBJ(*ALL) LIB(library-name) DEV(TAP01) OBJTYPE(*ALL)
          VOL(*MOUNTED) ENDOPT(*LEAVE)
   ```

   If the objects you save include one or more journals, you may want to save all database files associated with those journals. This step makes it possible to reestablish journaling by deleting the old files and restoring the saved files.

Task 3. Access DST Options

Sign on to DST using the following steps:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

3. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
   ```

7. When the system has powered down, the IPL or Install the System display appears.

```
                       IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system










Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                              © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                      Use Dedicated Service Tools (DST)

Select one of the following:

           1. Perform an IPL
           2. Install the operating system
           3. Work with licensed internal code
           4. Work with disk units
           5. Work with DST environment
           6. Select DST console mode
           7. Start a service tool
           8. Perform automatic installation of the operating system
           9. Work with save storage and restore storage

Selection
     4

F3=Exit            F12=Cancel
```

Task 4. Remove the Unit From the User ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu and press the Enter key.

```
                         Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information









Selection
     1

F3=Exit         F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display and press the Enter key.

```
                      Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection









Selection
     3

F3=Exit         F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display; the following display is shown.

ASP/Disk Recovery

```
                    Work with ASP Configuration

 Select one of the following:

      1. Display disk configuration capacity
      2. Create User ASP
      3. Delete user ASP
      4. Add units to existing ASP
      5. Reset overflowed user ASPs
      6. Change ASP storage threshold
      7. Move units from one ASP to another
      8. Remove unit from configuration









 Selection
     8

 F3=Exit         F12=Cancel
```

4. Select option 8 (Remove unit from configuration) to remove the desired unit.

   Note: If any user ASP has overflowed, reset the ASP before removing the unit from the ASP. For more information on how to reset the ASP, see "Resetting an Overflowed User ASP" on page 15-67.

```
                     Remove Units from Configuration

 Type options, press Enter.
   4=Remove unit from the configuration

                  Serial
 OPT  Unit  ASP  Number  Type  Model  Address        Status
        1    1           9332   400   0010-0000FFFF  Configured
        2    1           9332   400   0010-0100FFFF  Configured
        3    1           9332   400   0010-0300FFFF  Configured
        4    1           9332   400   0010-0001FFFF  Configured
        5    1           9332   400   0010-0600FFFF  Configured
        6    1           9332   400   0010-0101FFFF  Configured
  4     7    2           9332   200   0010-0201FFFF  Configured
        8    2           9332   200   0020-0701FFFF  Configured
        9    3           9332   200   0010-0700FFFF  Configured

                                                                More...
 Press Enter to continue.

 F3=Exit   F5=Refresh    F11=Display non-configured units    F12=Cancel
```

5. Select the unit you want removed from the configuration by typing a **4** in the *OPT* column and pressing the Enter key.

   Note: All data in the ASP is destroyed unless you are removing a unit that does not have checksum protection from an ASP that has checksum protection.

   The Confirm Remove of Units display is shown.

```
                        Confirm Remove of Units

Warning: Removing a unit from an ASP will destroy all the data
from that ASP.

Press F10 to confirm your choice to remove the unit.
Press F12=Cancel to return to change your choice.
                    Serial
 Opt  Unit  ASP  Number  Type  Model  Address        Status
  4     7    2           9332   200   0010-0201FFFF  Configured










                                                                   Bottom


 F10=Confirm   F12=Cancel
```

6. Press F10 (Confirm) to confirm you want the unit removed. Press F12 (Cancel) to return and change your choice.
7. Exit from the Work with Disk Units function.

Task 4. Restore the Objects to the Source User ASP

1. Load the correct volume of the SAVSECDTA tapes.
2. Type the following to restore the user profiles:

   ```
   RSTUSRPRF USRPRF(*ALL) DEV(TAP01)
   ```

3. Load the correct volume of the RSTLIB or RSTOBJ tapes.
4. Use the appropriate restore command to restore all objects that were in the source ASP:

   ```
   RSTLIB SAVLIB(library-name) DEV(TAP01)
   ```

   Or:

   ```
   RSTOBJ OBJ(*ALL) SAVLIB(library-name) OBJTYPE(*ALL)
          ENDOPT(*REWIND) MBROPT(*ALL)
   ```

   If an object resided in a user ASP that no longer exists after the object is saved, the object is not restored unless you specify an existing ASP or the default value of *SAVASP for the RSTASP parameter on the appropriate Restore command.

5. Restore private authorities:

   ```
   RSTAUT
   ```

   If any objects were restored into a different library than the one from which they were saved, you must manually grant private authority for the restored objects with the GRTOBJAUT command.

Notes:

a. If the journals are in the system ASP, create a new receiver in the user ASP, and then run the CHGJRN command to attach the new journal.

b. If the user ASP contained journals, and you did not delete the old files and then restore them from the media, create a new journal receiver and the journal in the user ASP. Use the Start Journal Physical Files (STRJRNPF) and the Start Journal Access Paths (STRJRNAP) commands to start journaling again.

6. This completes the steps to remove a unit.

Working with Objects in User ASPs

The following ASP operations are described in this topic:

- Creating objects in a user ASP
- Transferring objects between ASPs
- Deleting objects in a user ASP
- Displaying objects in a user ASP
- Transferring existing journals into a user ASP
- Transferring existing journal receivers into a user ASP
- Moving journal receivers from an overflowed user ASP
- Moving journals from an overflowed user ASP
- Moving journals in an overflowed user ASP to a different ASP
- Moving journal receivers in an overflowed user ASP to a different ASP

The following command language (CL) commands support these ASP operations by providing an ASP or RSTASP parameter:

- Create Journal (CRTJRN)
- Create Journal Receiver (CRTJRNRCV)
- Create Library (CRTLIB)
- Create Save File (CRTSAVF)
- Restore Object (RSTOBJ)
- Restore Library (RSTLIB)
- Display Object Description (DSPOBJD)
- Display File Description (DSPFD)
- Work with Journal Attributes (WRKJRNA)
- Display Journal Receiver Attributes (DSPJRNRCVA)

For additional information on these commands, refer to the *CL Reference*.

Creating Objects in a User ASP

You may find it easier to display the contents of a user ASP and to save and restore it by creating a separate library for each user ASP on your system (such as ASP2LIB in the following example).

Once your user ASPs are configured, you can place libraries or objects in them as follows:

1. Create the libraries directly in the ASP by specifying a value on the ASP parameter on the Create Library (CRTLIB) command and then create the objects in the library. All objects created in a library in a user ASP are automatically created in the same user ASP.

   For example, to create a journal receiver in ASP 2, do the following:

```
CRTLIB LIB(ASP2LIB) ASP(2)
```

```
CRTJRNRCV  JRNRCV(ASP2LIB/RCVINASP2)
```

This is the recommended way of creating objects in a user ASP.

2. Objects can be created directly in an ASP by specifying a value on the ASP parameter on the Create command for some objects. However, the library that the object is being created in must be in either the system ASP or in the same user ASP that the object is being created in. A user ASP cannot contain a library if it contains journals, journal receivers, or save files whose library is in the system ASP. If a library exists in the user ASP, you cannot create journals, journal receivers, and save files in the same ASP unless the library for the journals, journal receivers, or save files is in the same ASP. For example,

   CRTSAVF FILE(ASP2LIB/TEST) ASP(2)

   where (2) is the number of the user ASP where you are placing the save file. The library for the save file is in the system ASP and ASP 2 does not contain any libraries.

   Once the object is created, all storage for the object resides in the designated user ASP. Changes and additions to that object are also made in the user ASP. If the ASP becomes full, it overflows into the system ASP. For additional information, see "Resetting an Overflowed User ASP" on page 15-67.

   It is recommended that all journals and journal receivers on the system have unique names. RCLSTG renames them if duplicate names are found when objects are placed in library QRCL and the user cannot rename them to their original name.

   Monitor the size of objects to prevent them from overflowing into the system ASP with the MAXRCDS parameter on the CRTSAVF command, and the THRESHOLD parameter on the CRTJRNRCV command.

Transferring Objects between ASPs

You cannot directly move objects between ASPs. However, you can transfer them from one ASP into another ASP by using the save and restore commands (the RSTASP parameter on the RSTOBJ and RSTLIB commands).

If you try to restore an object to a user ASP by explicitly specifying the desired user ASP for the RSTASP parameter and the designated user ASP does not exist, or it does exist but contains a library, a message is sent to the user indicating that the object is not restored. However, if the object was originally in a user ASP and the user ASP does not exist at the time of the restore operation, and the default value RSTASP(*SAVASP) is specified, the object is restored to the system ASP and an informational message is sent to the user. The object is restored to the user ASP from which it was saved (as long as that ASP exists on your system) unless you specify a different ASP number on the restore command (with the RSTASP parameter).

If you attempt to restore objects (other than journals, journal receivers, and save files) to a user ASP other than the user ASP that contains their library, the objects are automatically restored to the ASP where their libraries are found. An informational message is sent to the user.

Private authorizations cannot be saved or restored with an object. You must manually grant private authorities to your objects after they are restored. If you have several objects to be moved that have many private authorities, consider using the Save System (SAVSYS) or the Save Security Data (SAVSECDTA) command. This command saves all private authorities and user profiles that are restored with the Restore User Profile (RSTUSRPRF) and Restore Authority (RSTAUT) commands.

To transfer a library or an object from one ASP to another ASP, do the following:

1. Save the library or the object using either the SAVLIB or SAVOBJ command.
2. Delete the library or the object.
3. Restore the library or the object to the new ASP by specifying:

   ```
   RSTLIB(ASPNLIB) RSTASP(N)
   ```

   where (N) specifies the number of the ASP to which you want the library restored.

   or

   ```
   RSTOBJ OBJ(OBJNAME) RSTASP(N)
   ```

Deleting Objects in a User ASP

To delete libraries or objects from a user ASP, use the appropriate delete commands to delete or clear the library that contains the objects.

Displaying Objects in a User ASP

There is no specific CL command to display objects contained in a user ASP. If you want to display a list:

1. Do one of the following (the first method is recommended):
   a. Create the library in the user ASP and then create the objects in the library.
   b. Create a separate library in the system ASP for each user ASP on your system before you begin placing objects in it. When you create an object, be sure that the object is created in that library.
2. To display these objects, enter the Display Library (DSPLIB) command.

   For example, you create a library called ASP2LIB to contain the objects that are in your user ASP2. If you later create a journal, journal receiver, or save file in user ASP2, you specify library ASP2LIB on the corresponding create command: CRTJRN, CRTJRNRCV, or CRTSAVF.

If you do not create separate libraries for your user ASPs, there is no direct way to display objects in a user ASP. There are indirect methods that enable you to identify objects:

1. You can determine which ASP an object is in by using the DSPOBJD command and looking at the number shown on the *Aux stg pool* field.
2. You can display all the object descriptions on your system to an out file using the DSPOBJD command, query the file, and search for objects in the ASP.

Transferring Existing Journals and Files into a User ASP

The libraries for the files being journaled and the library for the journal must be in the same ASP. Therefore, if the files being journaled are in a user ASP, it follows that the library for the journal is also in the same user ASP. For the purpose of recovery as well as performance, it is recommended that the journal receiver be placed in a different user ASP. If a failure occurs in the ASP that contains the files and the journal, you do not lose both the files and the journaled changes which are in the receiver. For maximum system performance, do not place your files and journal receiver in the same user ASP. To do so causes contention between access to the file and access to the journal receiver.

There are two methods for transferring journals and journal receivers into a user ASP. The first method is recommended. This involves creating the library for the journal (or journal receiver) in the user ASP. The objects can either be restored or newly created in the library. The second method is to keep the library for the journal (or journal receiver) in the system ASP and either restore or create the objects in the user ASP. Notice that a user ASP can contain either isolated objects (journals, journal receivers, and save files) or libraries, but not both.

Method 1: To transfer an existing journal to a user ASP using the first method, do the following:

1. If you are going to restore the journal, use the SAVOBJ or the SAVLIB command to save the journal.
2. Because the library for the journal is restored or recreated in the user ASP, the files must be moved to the same user ASP before you can resume journaling the files and access paths after the move.
3. Save any physical files being journaled and any logical files that have their access paths journaled. Restoring the files to the library in the user ASP automatically resumes journaling for the files.
4. Delete the physical files and logical files using the Delete File (DLTF) command.
5. Delete the journal using the Delete Journal (DLTJRN) command.
6. Create the library for the journal in the user ASP using the Create Library (CRTLIB) command and specify the desired ASP on the ASP parameter, or restore the library for the journal to the user ASP using the Restore Library (RSTLIB) command and specify the desired user ASP on the RSTASP parameter.
7. Create the journal in the ASP using the Create Journal (CRTJRN) command, or restore the journal to the library in the user ASP using the Restore Object (RSTOBJ) command.
8. Restore the physical files and logical files to the libraries in the user ASP. If you want to restore the files to their original libraries, you must first move those libraries to the user ASP. Restoring the files automatically resumes journaling for the files.
9. Save the files so that the journaled changes can be applied, if necessary. To save the files, see "Saving Journaled Files" on page 12-12.

Method 2: To transfer an existing journal into a user ASP using the second method, do the following:

1. If you are going to restore the journal, save the journal with either the SAVOBJ or SAVLIB command.
2. Optionally, save any physical files being journaled and any logical files whose access paths are being journaled. This allows you to begin journaling later by simply restoring the files.
3. You may want to use the SAVSYS command to save the system. By using this command now, you can save all private object authorities. If you do not do this, you must manually restore private authorities with the GRTOBJAUT command.
4. You can end journaling of access paths using the ENDJRNAP command. For physical files, use the ENDJRNPF command.

   If you saved the physical and logical files, delete those files with the DLTF command.
5. Delete the journal with the DLTJRN command.
6. Restore the journal to the user ASP by specifying:

   RSTOBJ OBJ(XXX) OBJTYPE(*JRN) RSTASP(N)

   where (N) is the number of the ASP to which the journal is restored. You can also create a journal with the ASP parameter on the Create Journal (CRTJRN) command.
7. Start journaling with the STRJRNPF and STRJRNAP commands.

   If you deleted the physical and logical files in step 4, restore these files to begin journaling using the RSTOBJ or RSTLIB command.
8. Reestablish private authorities to the journal and the database files, if they were deleted, with the GRTOBJAUT command. If you saved the system, restore your user profiles and authorities with the RSTUSRPRF and RSTAUT commands.
9. This completes the steps to transfer an existing journal to a user ASP.

Changing to Journal Receiver on a User ASP

There are two methods you can use to change to journal receivers on user ASPs. The first method is recommended.

Method 1:

1. Create a library in the user ASP using the Create Library (CRTLIB) command and specify the desired user ASP on the ASP parameter.
2. Create a new journal receiver in the library in a user ASP using the Create Journal Receiver (CRTJRNRCV) command.
3. Change journal receivers so the new journal receiver is attached and actively receiving journal entries by specifying:

   CHGJRN JRN(XX) JRNRCV(YY)

   where (XX) is the name of the journal, and (YY) is the name of the new journal receiver.

   You can save the detached journal receiver and delete it from the system.

Method 2:

1. Create a new receiver (CRTJRNRCV command) in a user ASP by specifying a user ASP name on the ASP parameter when the library for the journal receiver is in the system ASP.
2. Change journal receivers so the new journal receiver is attached and actively receiving journal entries by specifying:

   CHGJRN JRN(XX) JRNRCV(YY)

   where (XX) is the name of the journal, and (YY) is the name of the new journal receiver.

Moving Journal Receivers From an Overflowed User ASP to a Different ASP

To maintain journaling for the files, do the following steps:

1. Use the Display Journal Receiver Attributes (DSPJRNRCVA) command to determine the names of the journal receivers associated with the journal.
2. If the journal receivers to be moved are attached to a journal, create a new journal receiver on a different ASP using the CRTJRNRCV command. Consider using a name for the journal receiver that continues your naming conventions.
3. Change the journal using the Change Journal (CHGJRN) command, and specify the newly created journal receiver on the JRNRCV parameter.
4. Save the journal receivers from the overflowed user ASP that are associated with the journal using the Save Library (SAVLIB) or the Save Object (SAVOBJ) command.
5. Delete the library from the overflowed user ASP using the DLTLIB or DLTJRNRCV command.
6. Restore the library and receivers to the different ASP using the RSTLIB or RSTOBJ command.

Use the Work Journal Attributes (WRKJRNA) command to verify the restored journal receivers are now associated with the same journal and that the files are being journaled correctly.

Moving a Journal From an Overflowed User ASP to a Different ASP

To maintain journaling for the files, do the following steps:

1. Save the journal that is being moved to a different ASP using the Save Library (SAVLIB) command
2. Save the files that are currently being journaled using the SAVLIB command.

   By saving the files while they are still being journaled, the system will automatically start journaling again when you restore the files after restoring the journal.
3. End journaling for any access paths being journaled by using the End Journaling Access Path (ENDJRNAP) command.
4. End journaling for any physical files being journaled by using the End Journaling Physical Files (ENDJRNPF) command.
5. Use the Work Journal Attributes (WRKJRNA) command to determine the names of the journal receivers associated with the journal.

6. Delete the journal using the Delete Journal (DLTJRN) command.
7. Save the journal receivers that were associated with the journal being moved using the Save Object (SAVOBJ) command.
8. Delete the library for the journal using the Delete Library (DLTLIB) command.
9. Restore the library for the journal to the different ASP using the Restore Library (RSTLIB) command. This will create a new journal receiver and journal.
10. Restore the journaled files using the Restore Library (RSTLIB) command.
11. Restore the receivers saved in step 1 using the Restore Object (RSTOBJ) command.

Use the Work Journal Attributes (WRKJRNA) command to verify you have the same set of journal receivers associated with the journal in the different ASP and that the files are being journaled correctly. There will be a chain break between the newly attached receivers and the receivers that were just restored.

Chapter 16. Working with Disk Recovery

In many cases, a disk unit failure can be corrected by your service representative so that you can continue operations without replacing the disk. Replacing a disk requires a dedicated system and a special IBM program to copy the contents of the disk to either offline media and then back again to the new disk, or directly to the new disk. If data cannot be read, the system may not be able to recover it. In that case, use the OS/400 setup procedure after the disk is prepared and then restore the entire system from your save media using the restore commands. In some cases, the special program is only partially successful because it can correctly copy only some sectors of a disk. The sectors that the program cannot read are reset to zero. Sectors that are reset to zero can produce results that cannot be predicted. If the sector is part of an object description, the object may be unusable. If the sector is part of the data in a file, the file may still be usable except for the records on the sector that were reset to zero. You may have to restore the entire system. Discuss the situation with your service representative.

Types of Disk Damage: If you choose to continue operating, the next initial program load (IPL) of your system checks for sectors that were reset to zero and attempts to determine the objects to which they belong. There are two possible types of damage:

- Damage detected by the OS/400 licensed program. The system determines the damaged object, and the damage is marked in the object. Messages are sent to QHST to identify damaged objects. A single message (CPF8196 or CPF8197) is sent to QSYSOPR to notify the system operator that there are messages in QHST about objects that have been damaged.
- Damage not known to the OS/400 licensed program. The system is unable to determine which object was damaged. No message is sent. If you attempt to use this object, the system detects the damage and the object is marked as damaged.

If you try to use either type of damaged object, a damage exception occurs.

See the topic "How the Save Operation Handles Damage" on page 9-2, or "How the Restore Operation Handles Damage" on page 9-3 for more information.

Damage When Checksum or Mirrored Protection Is in Effect: For information on recovering from disk unit failures when checksum or mirrored protection is in effect, or when user auxiliary storage pools (ASPs) are used, see Chapter 15, "Working with Auxiliary Storage Pools."

Disk recovery has to do with recovering from the loss of data due to failures of the storage media contained within disk units. This chapter describes functions you can use to shorten the time it takes to recover from such failures and minimize the loss of data.

Use the following recovery functions to provide protection from disk failure:

- **Auxiliary storage pools** (ASPs). ASPs allow you to control where certain types of objects are stored on auxiliary storage devices, thus allowing you to isolate libraries and certain objects on physical disk devices of your choice. This isolation protects the libraries and objects from losing data due to disk media failures occurring in other ASPs.

- **Checksum protection**. This protects data stored in an ASP from the failure of a single disk. The system automatically reconstructs data after the disk is replaced. This protection is available for the system and user ASPs. When the system ASP has checksum protection, mirrored protection cannot be started for any other ASP.
- **Mirrored protection**. This protects data stored in an ASP from the failure of a single unit that has mirrored protection by storing identical data on another disk unit in the same user ASP. If one of the mirrored units in the mirrored pair fails, the system automatically uses the remaining unit of the mirrored pair until the failed unit can be repaired or replaced.

ASP Recovery Actions Performed by the Service Representative

The recovery actions performed by the service representative are provided here for your information. This discussion applies to a disk failure in an ASP when checksum protection or mirrored protection is not in effect.

If a disk unit in your system fails, your service representative determines whether data loss has occurred. Depending on the amount of data loss that has occurred, you and the service representative may decide that it is impractical to restore the data onto a replacement device and try to run your system. For the purpose of the ASP recovery actions discussed below, it is assumed you decide it is impractical to use the saved data. This case is referred to as a disk unit media failure.

The following steps are performed by your service representative when assisting you in recovering from a disk unit media failure:

1. The service representative attempts to use the DST Save Disk Unit Data function to save the data on the failed unit(s) to tape.
   - If all data from the failed unit is saved, it can be restored to the replacement unit. Further recovery steps are not needed beyond those required for other cases (such as power failures or other disk failures) where machine processing ends abnormally without disk media damage.
   - If less than 100% of the data is saved, you and the service representative must decide if it is practical to try to run your system using the saved data on a replacement unit. For these ASP recovery procedures, the assumption is that the save is unsuccessful to the point where you decide that a disk unit media failure has occurred.
2. The service representative attaches the replacement disk unit to the system. If the disk that failed contained unit 1 (the load source unit), it will be attached so that it is addressed correctly to indicate it contains unit 1. Otherwise, no specific address is required for the new disk.

   Note: In a mirrored ASP, the placement of the replacement unit affects the address of the unit. Because it affects the address of the unit, it can also affect the level of protection for the unit. The replacement unit should be placed in the exact location, and be given the same address, as the unit it is replacing.
3. The following occurs if the device that failed contained unit 1:
   a. The service representative performs an IPL of the system from your last SAVSYS or SAVSTG tapes. The stand-alone licensed internal code

install utility (option 24) is used to format the new unit 1 and copy the system licensed internal code from tape to the new unit 1.

Other units in the system will not be cleared by this step. However, they will be considered nonconfigured at this point. Installing licensed internal code results in a default disk configuration of just unit 1 being configured into the system.

b. The service representative uses the DST Recover Configuration function to request the system to recover the disk configuration that was in effect prior to the failure by reading information stored on the other disks in the system. This disk configuration information is placed on unit 1.

This step is necessary for the system to establish which disks were in the system ASP and which were in the user ASPs, and whether checksum protection or mirrored protection was in effect for the system.

Certain values which are normally stored on unit 1 cannot be rebuilt and will be lost. Most of these will be put back into effect automatically by IPL processing performed by the OS/400 program. You may have to set other values correctly during the recovery actions you perform.

4. The following occurs if the disk unit that failed contained a unit other than unit 1, but the unit is in the system ASP:

   a. The service representative uses the DST Replace Disk function to indicate to the system that the old (failed) unit is being replaced by the newly attached unit. The newly attached unit is formatted at this point.

   b. The service representative performs an IPL of the system from your last SAVSYS or SAVSTG tapes and uses the stand-alone utility function to restore (option 23) the licensed internal code. This allows for recovery of the PTF index when restoring the OS/400 licensed program.

5. The following occurs if the disk unit that failed contained a unit other than unit 1 and is not in the system ASP:

   a. The service representative uses the DST Replace Disk function to indicate to the system that the old (failed) unit is being replaced by the newly attached unit. The newly attached unit is formatted at this point.

When you continue the recovery actions by selecting to either IPL or to install the operating system, additional disk recovery processing may be performed, if necessary. It is at this point that the other disk units (if any) in the ASP which contained the failed disk unit are cleared.

Recovering from Disk Unit Media Failures

This topic provides three methods to recover from disk media failures:

1. Method 1 is the recovery procedure for a failure in the system ASP without user ASPs configured. Mirrored protection and checksum protection are not in effect.

2. Method 2 is the recovery procedure for a failure in the System ASP that has user ASPs configured. Mirrored protection and checksum protection are not in effect.

3. Method 3 is the recovery procedure for a failure in a user ASP without checksum protection or mirrored protection.

Method 1. Recovering the System ASP without User ASPs Configured

This example describes how to recover from a media disk failure in the system ASP when checksum or mirrored protection is not in effect, and no user ASPs are configured. It assumes that you have saved your system by using the SAVSYS, SAVLIB(*NONSYS), and SAVDLO commands, and that you have a complete backup copy of the system ASP. This information is required to restore the system after a failure.

If a unit in the system ASP fails, as described above, you and your service representative determine whether it should be treated as a disk unit media failure. If this is the case, the service representative will replace the failed disk. As a result, the system will clear all units allocated to the system ASP (all objects in the system ASP are destroyed).

After the service representative replaces the failed disk, the licensed internal code is installed or restored. The following procedure assumes that the licensed internal code has been installed or restored.

Task Overview

You will perform the following tasks during this procedure:

1. Restore the operating system
2. Restore the remaining parts of the system
   a. Restore user profiles
   b. Restore configuration objects
   c. Restore user libraries
   d. Restore changed objects
   e. Apply journaled changes
   f. Restore document library objects

Task 1. Restore the Operating System

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                         IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

```
                      Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.

 F12=Cancel
```

3. Press the Enter key.

4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                           Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .  2924

F3=Exit          F12=Cancel
```

ASP/Disk Recovery

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                    Confirm Language Feature Selection

Language feature . . . . . . . . . . . . . . . . . :   2924

Press Enter to confirm your choice for language feature.
  Installing the system will continue.
Press F12 to return to change your choice for
  language feature.









F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. The following display is shown only if disk units have been attached to the system and are in nonconfigured status.

```
                    Add All Disk Units to the System

Select one of the following:

         1. Add all disk units to the system auxiliary storage pool
         2. Keep the current disk configuration
         3. Perform disk configuration using DST










Selection
    __
```

Option 1 (Add all disk units to the system auxiliary storage pool)

Select this option if you want to add all of the nonconfigured units to the system auxiliary storage pool. Before adding the units to the system, all data stored on the non-configured units is deleted.

Note: Adding units can change the checksum set configuration of the system ASP. You can use option 3 (Perform disk configuration using DST) to calculate the effect of adding units to the system ASP.

Option 2 (Keep the current disk configuration)

Select this option if you plan to add the nonconfigured units to user ASPs or use them as spare units. This option continues the IPL without adding units to the system configuration. The disk units that are in nonconfigured status will remain so.

Option 3 (Perform disk configuration using DST)

This option starts the Dedicated Service Tools (DST). On the DST main menu, select option 4 (Work with Disk Units).

8. Press the Enter key.

   As the disk units are configured, the following display is shown:

```
                Formatting additional disk units in progress.
                      Seconds running . . . . :  ___
```

   Adding disk units takes several minutes. The time it takes depends on the size of each unit and the ability of the system to do multiple adds at the same time.

9. Status messages are displayed.

   The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                         Running IPL Step

Current IPL step . . . . : Storage Management Recovery
```

After the IPL steps complete, the Install the Operating System menu appears.

```
                        Install the Operating System

Type choices, press Enter.

    Default
      option . . . .  _             1=Take defaults, show no
                                      other installing displays
                                    2=Change installing options

    Date:
      Year . . . . .  _             00-99
      Month  . . . .  _             01-12
      Day  . . . . .  _             01-31

    Time  . . . . . . _     00-23   HH is hours
                      _     00-59   MM is minutes
                      _     00-59   SS is seconds
```

10. When the Install the Operating System display is shown, use the following information to respond to the prompts.

Default Option

Value **Description**

1 Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation.

If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.

2 If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system.

If you select 2 for *Default option*, the Installing Options display appears.

Date

The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

Time

The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

11. Press the Enter key.

    If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages are for your information only. Continue loading tapes in sequence when messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

    If you selected option 2 (Change installing options), the following display is shown.

```
                         Specify Install Options

Type choices, press Enter.

    Restore option . . . . __          1=Restore programs and language
                                         objects from current tape
                                       2=Do not restore programs or
                                         language objects
                                       3=Restore only language objects
                                         from current tape
                                       4=Restore only language objects
                                         from a different tape

    Job and output
      queue options. . . .  1          1=Clear,  2=Keep
```

12. When the Installing Options display appears, use the following information to respond to the prompts.

    Restore Option

Value	**Description**
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

Notes:

a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.

b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.

c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.

d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.

e. The system entry-point table is created again.

f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	**Description**
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                    Specify Restore Options

Type choices, press Enter.

    Restore from tape:

      System values . . . . . 2       1=Restore,  2=Do not restore
      Edit descriptions . . . 2       1=Restore,  2=Do not restore
      Message reply list. . . 2       1=Restore,  2=Do not restore
```

13. Using the following information, respond to the prompts on the Restore Options display.

System Values

Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID	Default system code page and character set
QCURSYM	Currency symbol
QDATFMT	Date editing format
QDATSEP	Date separator character
QDECFMT	Decimal data editing format
QKBDTYPE	Default work station keyboard type
QLEAPADJ	Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Recovery

Value	Description
1=Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

14. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

15. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    `Load the next volume on the tape device (1 2).`

16. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    `Operating system has been installed. IPL in progress.`

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

17. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).
18. Press the Enter key. Informational messages are displayed.
19. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.
20. When the IPL Options display is shown, respond to the prompts using the following information.

```
                                IPL Options

Type choices, press the Enter key.

  System date . . . . . . . . . . . . . .   07 / 26 / 88    MM / DD / YY
  System time . . . . . . . . . . . . . .   12 : 00 : 00    HH : MM : SS
  Clear job queues . . . . . . . . . . . .  N               Y=Yes,  N=No
  Clear output queues  . . . . . . . . . .  N               Y=Yes,  N=No
  Clear incomplete job logs  . . . . . . .  N               Y=Yes,  N=No
  Start print writers  . . . . . . . . . .  Y               Y=Yes,  N=No
  Start this device only . . . . . . . . .  N               Y=Yes,  N=No


  Set major system options . . . . . . . .  N               Y=Yes,  N=No
  Define or change system at IPL . . . . .  N               Y=Yes,  N=No






Last power-down operation was normal
                                     (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure 16-1. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a `Y` to start the console device only, or type an `N` to start all devices.

d. Enter the value for set major system options.

   Type an `N` for each option to leave them unchanged, or type `Y` to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to `N`. If you installed the licensed internal code using option 24 (Install), the default value is set to `Y`.

21. Press the Enter key.

22. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

ASP/Disk Recovery

```
                 Edit Rebuild of Access Paths              RCHAS331
                                                  05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                   ------------Access Paths----------  Unique  Rebuild
Seq   Status       File        Library     Member      Keyed   Time
25__  IPL          FILE234512  LIBRARY111  MBR1234567  No      00:00:56
25__  IPL          FILE234513  LIBRARY111  MBR1234567  No      00:00:56
75__  IPL          FILE234514  LIBRARY111  MBR1234567  Yes     00:00:56
75__  IPL          FILE234515  LIBRARY111  MBR1234567  Yes     00:00:56
88__  IPL          FILE234516  LIBRARY111  MBR1234567  No      00:00:56
99__  AFTIPL       FILE234517  LIBRARY111  MBR1234567  Yes     00:00:56
*OPN  OPEN         FILE126789  L123456789  MBR4567890  Yes     12:34:56
*OPN  OPEN         FILE346789  L123456789  MBR4567890  No      12:34:56
*HLD  HELD         F123336789  L123456789  MBR4567890  No      10:30:06
*HLD  HELD         F123456789  L123456789  MBR4567890  Yes     99:56:01
                                                          More...

F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                         Display Access Path Status

IPL Threshold . . . . . . : 88

            ----------Access Paths----------   Rebuild     Current
Status      File        Library     Member       Build Time  Run Time
RUN         F123456789  L123456789  MBR4567890               00:00:01
JRN         F123456789  L123456789  MBR4567890
JRN         F123456789  L123456789  MBR4567890
JRN         F123456789  L123456789  MBR4567890
JRN         F123456789  L123456789  MBR4567890
JRN         F123456789  L123456789  MBR4567890
JRN         F123456789  L123456789  MBR4567890
SYS         F123456789  L123456789  MBR4567890   12:34:56
SYS         F123456789  L123456789  MBR4567890   12:34:56
IPL         F123456789  L123456789  MBR4567890   12:34:56
                                                              More...



F3=Exit and continue IPL   F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN — indicates that the access path is being rebuilt.
- SYS — indicates the access path is a system access path and is waiting to be rebuilt.
- JRN — indicates that the access path is being recovered from its associated journal.
- IPL — indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

23. Press the Enter key to continue.
24. Ensure the keylock switch is in the Normal position.
25. This completes the restore operation for the operating system if you have no other restore steps to perform.
26. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

Task 2. Restoring the System

Other Considerations: Use the restore commands (not option 21 on the Restore menu) if:

- You prefer to enter the commands manually.
- You saved changed objects or have journal changes to apply.
- You performed individual save operations instead of SAVLIB(*NONSYS), you must use a RSTLIB command for each saved library. If you saved individual objects using the SAVOBJ or SAVCHGOBJ command, you must use a RSTOBJ command for each group of saved objects.
- You saved the security information with the Save Security Data (SAVSECDTA) command, you must restore the information using the restore commands.
- You saved logical file access paths using either the SAVOBJ or SAVCHGOBJ command, you must restore the logical files the same way you restored the physical files using the RSTOBJ command.
- Your documents and folders were saved in multiple tape files using SAVDLO DLO(*ALL) FLR(*ANY), use DSPTAP DATA(SAVRST) to display the beginning and ending sequence numbers. You will need this information later during the restore operation.

Use **one** of the following two methods to restore the user profiles, device configuration objects, user libraries, and authority:

1. Use option 21 (The system) on the Restore menu if you are restoring the user profiles from a SAVSYS tape and the considerations listed previously in "Other Considerations" do not apply.
2. Use the Restore commands (not option 21 on the Restore menu) if any of the consideration listed previously in "Other Considerations" apply.

Method 1. Using Option 21 (The system) on the Restore Menu

To restore user profiles, device configuration objects, user libraries, and authority, use the following steps:

1. Sign on the system as the security officer; type QSECOFR in the user prompt and the password (if password security is active) associated with that user ID on the Sign On display.
2. Press the Enter key.
3. Type the following command on the command line and press the Enter key.

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```
4. Ensure that the correct volume of your last set of save tapes is loaded and make the tape device ready. The tape should begin with a file containing the user profiles. Run the DSPTAP using *SAVRST to find the file labeled QFILEUPR.
5. Go to the Restore menu:

   ```
   GO RESTORE
   ```

   The Restore menu is shown.

```
RESTORE                              Restore
                                                          System:   RCHAS791
Select one of the following:

      1. Files
      2. Libraries
      3. Documents and folders
      4. Programs
      5. Other objects
      6. Licensed programs

     20. All libraries other than system library
     21. The system
     22. All IBM libraries other than system library
     23. All user libraries

     50. Restore from System/36 format
                                                                      More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

6. Press the Enter Key.
7. Select option 21 (The system) on the Restore menu and press the Enter key. The following display is shown.

ASP/Disk Recovery

```
                         Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .  : TAP01                Names
                                    ______
                                    ______
                                    ______

  Prompt for commands . . . . .  :    Y                  Y=Yes, N=No









F3=Exit   F12=Cancel
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device after the first tape is read.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting and uses the default values.

Note: If you have multiple tape files from the SAVDLO DLO(*ALL), type a *Y* (the default) for the *Prompt for commands* field on the Specify Command Defaults display. When the RSTDLO command prompt is displayed, enter the beginning and ending sequence numbers to restore all SAVDLO tape files.

Option 21 will guide you through the following if you selected Y for the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS *ALL *IMMED

b. RSTUSRPRF *ALL

c. RSTCFG *ALL

d. RSTLIB SAVLIB(*NONSYS)

e. RSTDLO DLO(*ALL) SAVFLR(*ANY)

f. RSTAUT

g. STRSBS

8. Continue loading the save tapes in sequence when the system sends a message to load the next volume.

If a media error occurs....

If an unrecoverable media error occurs during the RSTLIB procedure, you can restart the procedure using the STRLIB parameter on the RSTLIB command. Do one of the following:

1. If the STRLIB parameter was used on the SAVLIB command), determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). Load the next tape and enter the following:

   ```
   RSTLIB SAVLIB(*NONSYS) STRLIB(starting-library-name)
   ```

2. If the STRLIB parameter was not used on the SAVLIB command, look at the job log to determine the library where the previous RSTLIB failed.

   a. Determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). The correct starting library is normally the second library listed after the last library restored successfully.

   b. Load the first tape of the SAVLIB media and type the following command to start the restore operation again.

   ```
   RSTLIB SAVLIB(*NONSYS) STRLIB(starting-library-name)
   ```

9. If you are restoring the system using the SAVSYS tapes, this completes the restore operation. There are no other restore steps to perform.
10. Power down the system (PWRDWNSYS OPTION(*IMMED) RESTART(*YES) to perform a normal IPL and return the system to normal operations.

Method 2. Using the Restore Commands

To use the commands to restore the system, do the following:

1. Sign on the system.
2. Change the QSYSOPR message queue:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

3. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Messages are sent indicating when the subsystems have ended and the system is in a restricted state.

4. Restore user profiles from the correct save tape file (label QFILEUPR).

   Note: Use the tapes from the most recent complete save operation (SAVSYS), or if the security data was saved since the last complete save operation, use the SAVSECDTA tapes.

   If the SAVSYS tape is used:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
   ```

   If the save security data tape (SAVSECDTA) is used:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

5. Restore the device configuration objects from the SAVSYS tape:

   ```
   RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*LEAVE)
   ```

The time that this takes can vary significantly.

6. Restore the IBM and user libraries in one of the following ways:

   If you used SAVLIB LIB(*NONSYS) to save the IBM-supplied and user libraries, load the correct volume and then type the following:

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
   ```

   Or, if you used SAVLIB LIB(*IBM) and SAVLIB LIB(*ALLUSR) to save the IBM and user libraries, load the correct tape and then type the following two commands. The first command must complete before entering the second command.

   ```
   RSTLIB SAVLIB(*IBM) DEV(TAP01) ENDOPT(*LEAVE)
        MBROPT(*ALL) ALWOBJDIF(*ALL)
   ```

   ```
   RSTLIB SAVLIB(*ALLUSR) DEV(TAP01) ENDOPT(*LEAVE)
        MBROPT(*ALL) ALWOBJDIF(*ALL)
   ```

   Note: If you saved individual libraries and objects with the SAVLIB, SAVOBJ, and SAVCHGOBJ commands, then you will have to restore the individual libraries and objects with the RSTLIB command (not RSTLIB SAVLIB(*NONSYS)) and the RSTOBJ command with ALWOBJDIF(*ALL) and MBROPT(*ALL) specified.

   If a media error occurs....

   If an unrecoverable media error occurs during the RSTLIB procedure, you can restart the procedure using the STRLIB parameter on the RSTLIB command. Do one of the following:

   1. If the STRLIB parameter was used on the SAVLIB command), determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). Load the next tape and enter the following:

      ```
      RSTLIB SAVLIB(*NONSYS, *ALLUSR, or *IBM) STRLIB(starting-library-name)
      ```

   2. If the STRLIB parameter was not used on the SAVLIB command, look at the job log to determine the library where the previous RSTLIB failed.

      a. Determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). The correct starting library is normally the second library listed after the last library restored successfully.

      b. Load the first tape of the SAVLIB media and type the following command to start the restore operation again.

      ```
      RSTLIB SAVLIB(*NONSYS, *ALLUSR, or *IBM) STRLIB(starting-library-name)
      ```

7. If you have documents and folders to restore, load the correct tape and type the following:

   Note: If the saved data is contained in one or more files with folders and documents, you must specify beginning and ending sequence numbers on the RSTDLO command. Otherwise, only the first file with folders and documents will be restored.

   ```
   RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
          SEQNBR(beginning-sequence ending-sequence)
   ```

   Note: If you are not using journaling, or do not have changed objects or daily mail (SAVSECDTA tape) to restore, continue with the next step.

Otherwise, continue with "Restoring Changed Objects and Applying Journal Changes."

8. To restore the authority, type the following:

   ```
   RSTAUT
   ```

   This completes the restore operation.

9. Power down the system (PWRDWNSYS OPTION(*IMMED) RESTART(*YES)) to perform a normal IPL and return the system to normal operations.

Restoring Changed Objects and Applying Journaled Changes

If you are using journaling and need to later apply journaled changes, continue with the following steps. Otherwise ignore these steps and continue with "Restoring Changed Objects."

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily save operations using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform the steps in "Working with Journals" for the system supplied journal QUSRSYS/QAOSDIAJRN. If you specified OBJJRN(*YES) on the SAVCHGOBJ command, you do not need to apply journal changes.

Working with Journals

1. Type the following and press the Enter key:

   ```
   WRKJRN
   ```

2. The Specify Journal Name display is shown. Specify *ALL for the *Library name* prompt and press the Enter key.

3. The Work with Journals display is shown. Type a **5** in the *Opt* field for each journal that you want to apply journaled changes to display the name of the currently attached journal receiver. Write down the name of the journal and journal receiver.

   Note: If you are using OfficeVision/400 or PC Support/400, you must apply journaled changes to the system supplied journal QUSRSYS/QAOSDIAJRN.

4. You cannot restore journal receivers from the SAVCHGOBJ media if active journal receivers are attached. To later apply all journaled changes that have occurred since the last complete save operation, you must restore the receivers to the system from the SAVCHGOBJ media.

   For each journal identified in the previous step, do the following steps:

   a. Create a journal receiver that will be used as a temporary receiver. Give it a name that will identify it as a temporary receiver, for example, TEMPnn. You can enter a description in the text (TEXT parameter) that identifies it as a temporary receiver for disaster recovery.

      ```
      CRTJRNRCV JRNRCV(library-name/TEMPnn)
            TEXT('temporary journal receiver for journal xxx')
      ```

   b. To detach the current receiver and attach the new TEMPnn receiver, type the following and press the Enter key.

      ```
      CHGJRN JRN(library-name/journal-name) JRNRCV(library-name/TEMPnn)
      ```

   c. Delete the detached journal receiver (identified in the step where you wrote down the name of the journal and journal receiver) using the

ASP/Disk Recovery

Delete Journal Receiver (DLTJRNRCV) command. (This allows the journal receivers on the SAVCHGOBJ media to be restored successfully.

```
DLTJRNRCV JRNRCV(library-name/journal-receiver)
```

If you receive message CPA7025 *Receiver never fully saved*, enter an **I** to ignore and press Enter to continue the delete.

Restoring Changed Objects

Load the SAVCHGOBJ tape and enter the following to restore changed objects.

```
RSTOBJ OBJ(*ALL) DEV(tape-device) SAVLIB(library-name)
          OBJTYPE(*ALL) ENDOPT(*LEAVE) MBROPT(*ALL) ALWOBJDIF(*ALL)
```

You must repeat this step for every library saved using the SAVCHGOBJ command. If you specified SAVCHGOBJ LIB(*ALLUSR), type the following to determine the libraries that were saved:

```
DSPTAP *SAVRST
```

If you are using journaling, perform the steps in "Applying Journal Changes" for each journal you wish to apply journal changes to. Otherwise, continue with "Restoring Changed Documents and Folders."

Applying Journal Changes

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily saves using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform this step for the system supplied journal QUSRSYS/QAOSDIAJRN.

1. Determine the name of the last journal receiver (the last receiver restored) by entering the following:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

2. Press F15 (Work with receiver directory) to display the last receiver with a status of SAVED or PARTIAL. Write down the name of the receiver.

3. Create a receiver that follows the same naming convention as the last receiver but assign it a number of one greater.

   ```
   CRTJRNRCV JRNRCV(library-name/journal-receiver-nameNN)
   ```

   By doing this, you are doing what the CHGJRN command would normally do if the receiver was the current receiver being detached with a new receiver name being created. This allows your naming convention for journal receivers to continue.

4. Use the CHGJRN command to detach the temporary receiver and attach the new receiver you just created.

   ```
   CHGJRN JRN(library-name/journal-name)
         JRNRCV(library-name/journal-receiver-nameNN)
   ```

5. Determine the chain of receivers to be used in the APYJRNCHG command by entering the following command:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

   Press F15 (Work with receiver directory) to display the receivers. Write down the first and last receiver that you restored (last receiver is prior to the TEMPnn receiver). Notice that the first and last is the same receiver if only one journal receiver was restored.

6. When applying journal changes and the ending receiver has a status of PARTIAL (saved while attached), the TOENT parameter requires that a sequence number be specified on the APYJRNCHG command. Determine the last entry to be applied by entering the following command for the last receiver (identified in previous step):

   ```
   DSPJRNRCVA JRNRCV(library-name/last-journal-receiver-name)
   ```

   Write down the value for the *Last Sequence Number* field.

7. Enter the following command to apply the journaled changes using the first and last journal receivers identified on the Work with Receiver Directory display.

   ```
   APYJRNCHG JRN(library-name/journal-name)
             FILE((library-name/*ALL))
             RCVRNG(lib-name/first-receiver lib-name/last-receiver)
             FROMENT(*LASTSAVE) TOENT(last-entry)
   ```

   Attention

   Do not apply journal changes to the document and folder search index database files (QAOSSH10 through QAOSSH19) for journal QAOSDIAJRN in library QUSRSYS. You must specify individual files on the FILE parameter instead of *ALL.

   ```
   APYJRNCHG JRN(QUSRSYS/QAOSDIAJRN)
             FILE((QUSRSYS/QAOSAH05) (QUSRSYS/QAOKLY02)
                  (QUSRSYS/QAOSAH07) (QUSRSYS/QAOKLY03)
                  (QUSRSYS/QAOKDYX4) (QUSRSYS/QAOKNY06)
                  (QUSRSYS/QAOKDY01) (QUSRSYS/QAOKDY04)
                  (QUSRSYS/QAOKDY05) (QUSRSYS/QAOKDY07)
                  (QUSRSYS/QAOKDY08) (QUSRSYS/QAOKDY09))
             RCVRNG(lib-name/first-receiver lib-name/last-receiver)
             FROMENT(*LASTSAVE) TOENT(last-entry)
   ```

Restoring Changed Documents and Folders

If you performed daily save operations for documents and folders, do the following steps. Otherwise, continue with the RSTAUT command.

1. Load the last daily SAVDLO tape.

2. If you performed daily save operations to backup all new folders and new and changed documents since the last complete SAVDLO operation, type the following and press the Enter key.

   ```
   RSTDLO DLO(*ALL) DEV(TAP01) SAVFLR(*ANY) ALWOBJDIF(*ALL)
   ```

   If you saved all documents referred to by the mail using SAVDLO DLO(*MAIL), type the following and press the Enter key.

   ```
   RSTDLO DLO(*MAIL) DEV(TAP01)
   ```

3. Restore users' authority by entering:

   ```
   RSTAUT
   ```

   The time it takes for the RSTAUT command to complete can vary significantly. The time depends on the number of user profiles and private authorities that were saved during the save operation.

4. This completes the restore operation.

   Power down the system by entering the following:

```
PWRDWNSYS OPTION(*IMMED) RESTART(*YES)
```

to perform a normal IPL and return the system to normal operations.

Note: If you turned auto configuration off during the restore of the operating system, you must perform an IPL with the keylock switch in the **Normal** position.

Method 2. Recovering the System ASP with User ASPs Configured

This example describes how you recover from a media disk failure in the system ASP when checksum or mirrored protection is not in effect and user ASPs are configured. It assumes that you have saved your system by using the SAVSYS, SAVLIB(*NONSYS), and SAVDLO commands, and that you have a complete backup copy of the system ASP. This information is required to restore the system after a failure.

If a unit in the system ASP fails, as described above, you and your service representative determine whether it should be treated as a disk unit media failure. If this is the case, the service representative will replace the failed disk and, as a result, the system will clear all units allocated to the system ASP (all objects in the system ASP are destroyed and all overflowed user ASPs are cleared). All other user ASPs remain as they were before the failure occurred.

Task 1. Restore the Operating System

During this task, the system may have to perform some lengthy disk recovery processing prior to actually starting the install processing. See Appendix D, "Initial Program Load (IPL) Process" on page D-1 for more information on this disk recovery processing.

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                          IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

   Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

```
                 Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.









 F12=Cancel
```

3. Press the Enter key.
4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                       Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .  2924









F3=Exit        F12=Cancel
```

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary

language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                    Confirm Language Feature Selection

Language feature . . . . . . . . . . . . . . . . . :   2924

Press Enter to confirm your choice for language feature.
  Installing the system will continue.
Press F12 to return to change your choice for
  language feature.








F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. The following display is shown only if disk units have been attached to the system and are in nonconfigured status.

```
                      Add All Disk Units to the System

Select one of the following:

         1. Add all disk units to the system auxiliary storage pool
         2. Keep the current disk configuration
         3. Perform disk configuration using DST








Selection
    __
```

Option 1 (Add all disk units to the system auxiliary storage pool)

Select this option if you want to add all of the nonconfigured units to the system auxiliary storage pool. Before adding the units to the system, all data stored on the non-configured units is deleted.

Note: Adding units can change the checksum set configuration of the system ASP. You can use option 3 (Perform disk configuration using DST) to calculate the effect of adding units to the system ASP.

Option 2 (Keep the current disk configuration)

Select this option if you plan to add the nonconfigured units to user ASPs or use them as spare units. This option continues the IPL without adding units to the system configuration. The disk units that are in nonconfigured status will remain so.

Option 3 (Perform disk configuration using DST)

This option starts the Dedicated Service Tools (DST). On the DST main menu, select option 4 (Work with Disk Units).

8. Press the Enter key.

   As the disk units are configured, the following display is shown:

```
          Formatting additional disk units in progress.
                 Seconds running . . . . :  ___
```

   Adding disk units takes several minutes. The time it takes depends on the size of each unit and the ability of the system to do multiple adds at the same time.

9. Status messages are displayed.

   The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                          Running IPL Step

Current IPL step . . . . : Storage Management Recovery
```

Recovery

After the IPL steps complete, the Install the Operating System menu appears.

```
                     Install the Operating System

Type choices, press Enter.

   Default
     option . . . .  _             1=Take defaults, show no
                                     other installing displays
                                   2=Change installing options

   Date:
     Year . . . . .  _             00-99
     Month  . . . .  _             01-12
     Day  . . . . .  _             01-31

   Time  . . . . . . _    00-23    HH is hours
                     _    00-59    MM is minutes
                     _    00-59    SS is seconds
```

10. When the Install the Operating System display is shown, use the following information to respond to the prompts.

Default Option

Value	**Description**
1	Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation. If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.
2	If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system. If you select 2 for *Default option*, the Installing Options display appears.

Date

The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

Time

The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

11. Press the Enter key.

If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages are for your information only. Continue loading tapes in sequence when messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

If you selected option 2 (Change installing options), the following display is shown.

```
                          Specify Install Options

Type choices, press Enter.

   Restore option . . . . __          1=Restore programs and language
                                        objects from current tape
                                      2=Do not restore programs or
                                        language objects
                                      3=Restore only language objects
                                        from current tape
                                      4=Restore only language objects
                                        from a different tape



   Job and output
     queue options. . . .  1          1=Clear,  2=Keep
```

12. When the Installing Options display appears, use the following information to respond to the prompts.

    Restore Option

Value	**Description**
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

    Notes:

    a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.

    b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.

    c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.

    d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.

    e. The system entry-point table is created again.

    f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user

Recovery

decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	**Description**
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                         Specify Restore Options

Type choices, press Enter.

   Restore from tape:

     System values . . . . . 2        1=Restore,  2=Do not restore
     Edit descriptions . . . 2        1=Restore,  2=Do not restore
     Message reply list. . . 2        1=Restore,  2=Do not restore
```

13. Using the following information, respond to the prompts on the Restore Options display.

System Values Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID	Default system code page and character set
QCURSYM	Currency symbol
QDATFMT	Date editing format
QDATSEP	Date separator character
QDECFMT	Decimal data editing format
QKBDTYPE	Default work station keyboard type
QLEAPADJ	Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Value	Description
1=Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

14. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

15. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    `Load the next volume on the tape device (1 2).`

16. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    `Operating system has been installed. IPL in progress.`

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

17. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).

18. Press the Enter key. Informational messages are displayed.

19. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.

20. When the IPL Options display is shown, respond to the prompts using the following information.

```
                              IPL Options

Type choices, press the Enter key.

  System date  . . . . . . . . . . . . . .   07 / 26 / 88   MM / DD / YY
  System time  . . . . . . . . . . . . . .   12 : 00 : 00   HH : MM : SS
  Clear job queues . . . . . . . . . . . .   N              Y=Yes,  N=No
  Clear output queues  . . . . . . . . . .   N              Y=Yes,  N=No
  Clear incomplete job logs  . . . . . . .   N              Y=Yes,  N=No
  Start print writers  . . . . . . . . . .   Y              Y=Yes,  N=No
  Start this device only . . . . . . . . .   N              Y=Yes,  N=No


  Set major system options . . . . . . . .   N              Y=Yes,  N=No
  Define or change system at IPL . . . . .   N              Y=Yes,  N=No







Last power-down operation was normal
                                        (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure 16-2. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a `Y` to start the console device only, or type an `N` to start all devices.

d. Enter the value for set major system options.

   Type an `N` for each option to leave them unchanged, or type `Y` to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to `N`. If you installed the licensed internal code using option 24 (Install), the default value is set to `Y`.

21. Press the Enter key.

22. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

ASP/Disk Recovery

```
                Edit Rebuild of Access Paths                RCHAS331
                                                    05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                   ------------Access Paths----------  Unique  Rebuild
Seq   Status       File        Library     Member       Keyed   Time
25__  IPL          FILE234512  LIBRARY111  MBR1234567   No      00:00:56
25__  IPL          FILE234513  LIBRARY111  MBR1234567   No      00:00:56
75__  IPL          FILE234514  LIBRARY111  MBR1234567   Yes     00:00:56
75__  IPL          FILE234515  LIBRARY111  MBR1234567   Yes     00:00:56
88__  IPL          FILE234516  LIBRARY111  MBR1234567   No      00:00:56
99__  AFTIPL       FILE234517  LIBRARY111  MBR1234567   Yes     00:00:56
*OPN  OPEN         FILE126789  L123456789  MBR4567890   Yes     12:34:56
*OPN  OPEN         FILE346789  L123456789  MBR4567890   No      12:34:56
*HLD  HELD         F123336789  L123456789  MBR4567890   No      10:30:06
*HLD  HELD         F123456789  L123456789  MBR4567890   Yes     99:56:01
                                                           More...

F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                        Display Access Path Status

IPL Threshold . . . . . . : 88

          ----------Access Paths----------   Rebuild      Current
Status    File        Library      Member       Build Time   Run Time
RUN       F123456789  L123456789  MBR4567890                00:00:01
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
SYS       F123456789  L123456789  MBR4567890  12:34:56
SYS       F123456789  L123456789  MBR4567890  12:34:56
IPL       F123456789  L123456789  MBR4567890  12:34:56
                                                              More...



F3=Exit and continue IPL    F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN – indicates that the access path is being rebuilt.
- SYS – indicates the access path is a system access path and is waiting to be rebuilt.
- JRN – indicates that the access path is being recovered from its associated journal.
- IPL – indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

23. Press the Enter key to continue.
24. Ensure the keylock switch is in the Normal position.
25. This completes the restore operation for the operating system if you have no other restore steps to perform.
26. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

After the recovery process completes, you can proceed with the next step.

1. If the units that failed included unit 1, a message may appear during IPL processing, stating that some hardware configuration information has been lost and prompting you to enter that information again. To do this, use the Work with Hardware Products (WRKHDWPRD) command.
2. If the units that failed included unit 1, a message (CPI0916) may appear during IPL processing, stating that network attribute information has been lost and prompting you to set the correct values for these attributes on your system. Follow the instructions provided in that message to set the correct network attributes at this point.
3. If the units that failed included unit 1 and if you use the AS/400 Cryptographic Support (5728-CR1) licensed program on your system, you must reenter the master cryptographic key. To do this you use the Set Master Key (SETMSTK) command.

Task 2. Recover the Objects in the User ASP

1. End all subsystems using the following command:

```
ENDSBS SBS(*ALL)  OPTION(*IMMED)
```

   a. Type the following to reclaim storage:

```
RCLSTG
```

If you had journals, journal receivers, or save files (whose library is in the system ASP) stored in user ASPs when you began this procedure, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile.

If you had libraries and associated objects stored in a user ASP, this command will restore their addressability and transfers ownership of the objects to QDFTOWN user profile.

If objects are damaged in the ASP, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile. This happens whether you have isolated objects or an entire library in an ASP.

b. Restore user profiles from the correct save tape file (label QFILEUPR).

Note: Use the tapes from the most recent complete save operation (SAVSYS). If a SAVSECDTA command has been run since the last complete save operation, use the SAVSECDTA tapes.

If the SAVSYS tape is used, type the following:

```
RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
```

If the save security data tape (SAVSECDTA) is used, type the following:

```
RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
```

The time that this takes can vary significantly.

c. Restore the device configuration objects from your most recent SAVSYS tapes. Type the following:

```
RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*UNLOAD)
```

The time that this takes can vary significantly.

d. It is recommended that you save library QRCL and its contents to create a backup copy of the objects in the user ASPs for use in case another failure should occur. This copy may be needed in the following restore steps, depending how you chose to do the restore operation.

Load a scratch tape, type the following and press the Enter Key.

```
SAVLIB LIB(QRCL) DEV(TAP01) ENDOPT(*UNLOAD)
```

e. Restore the objects to the system in **one** of the following ways:

Method 1. Recovery When No Objects Exist in User ASPs or Old Type User ASPs Exist

If you had no objects stored in user ASPs or journals, journal receivers, or save files exist in user ASPs (where the library for the objects exists in the system ASP) at the start of this procedure, or if you want to simplify the restore process at the expense of going through restoring objects that are still intact in the user ASPs on your system, perform the following steps:

1. Use DSPTAP *SAVRST to determine the objects saved from library QRCL.
2. Delete library QRCL by typing the following and pressing the Enter key.

   ```
   DLTLIB LIB(QRCL)
   ```

3. Restore the IBM and user libraries in one of the following ways:

   If you saved the libraries using SAVLIB LIB(*NONSYS), enter the following and press the Enter key,

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) ENDOPT(*UNLOAD)
   ```

Or if you saved the IBM libraries using SAVLIB LIB(*IBM) and SAVLIB LIB(*ALLUSR), load the correct tape and then use the following two commands. The first command must complete before entering the second command.

```
RSTLIB SAVLIB(*IBM) DEV(TAP01)
```

And:

```
RSTLIB SAVLIB(*ALLUSR) DEV(TAP01)
```

4. Restore the objects from tape that were saved from library QRCL back to their original ASP libraries.

   For each group of objects, type the following and press the enter key.

   ```
   RSTOBJ OBJ(object-name) DEV(TAP01) SAVLIB(QRCL) RSTLIB(library-name)
          ALWOBJDIF(*ALL)
   ```

5. The RCLSTG command changed ownership of objects in user ASPs to QDFTOWN user profile. Transfer ownership of the objects owned by user profile QDFTOWN to the correct user profile.

   a. Type the following and press the Enter key.

      ```
      WRKOBJOWN USRPRF(QDFTOWN)
      ```

      The Work with Objects by Owner display is shown.

   b. On the Work with Objects by Owner display, type a **9** in the *Opt* column for each object in the ASP library that you want to change ownership for.

   c. If all the objects will have the same owner, type the following on the command line of the Work with Objects by Owner display and press the Enter key. Otherwise, continue with the next step.

      ```
      NEWOWN(owner-name)
      ```

      Note: If you enter NEWOWN(owner-name) on the command line of the Work with Object by Owner display, you will not have to enter an owner name in the *New owner* prompt on the Change Object Owner display.

   d. Press the Enter key.

   e. On the Change Object Owner (CHGOBJOWN) display, type the name of the new owner in the *New owner* prompt and press the Enter key. Repeat this step for all the objects that need the ownership changed.

6. If you have document library objects to restore, load the SAVDLO tape, and then type the following:

   ```
   RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
           SEQNBR(beginning-number ending-number)
   ```

   Note: If you have changed objects to restore or were journaling, do not run the RSTAUT command. Continue with "Task 3. Restoring Changed Objects and Applying Journal Changes."

7. Grant all private object authorities that existed when the system was saved by typing the following and pressing the Enter key. No media is required.

   ```
   RSTAUT
   ```

Method 2. Recovery of Objects and Libraries Existing in User ASPs

After the RCLSTG command is run, the addressability of libraries and objects in the user ASP is restored.

1. Restore the individual libraries to the system ASP from your save tapes.

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) OMITLIB(user-ASP-library-name)
   ```

   Or if you saved the IBM libraries using SAVLIB LIB(*IBM) and SAVLIB LIB(*ALLUSR), load the correct tape and then use the following two commands. The first command must complete before entering the second command.

   ```
   RSTLIB SAVLIB(*IBM) DEV(TAP01)
   ```

   And:

   ```
   RSTLIB SAVLIB(*ALLUSR) DEV(TAP01) OMITLIB(user-ASP-library-name)
   ```

   Do not restore libraries from the SAVLIB tapes that are in user ASPs (those found by reclaim processing). Specify the OMITLIB parameter to exclude up to 50 libraries that exist in the user ASPs. Journals should be restored before the journaled files, or journaling is not resumed for the restored files.

   If objects are found in QRCL, do the following:

   a. Delete the damaged objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:

      1) Display the library in the user ASP:

         ```
         WRKLIB LIB(library-name)
         ```

      2) Type a **12** (Work with objects) in the *Opt* column and press the Enter key.

      3) Find the objects to be deleted in the *Object* column.

      4) Type a **4** (Delete) in the *Opt* column for each object you want to delete.

      5) Press the Enter key.

   b. After the objects in the ASP are deleted, move the objects in library QRCL back to their original library.

      ```
      MOVOBJ OBJ(QRCL/object-name) OBJTYPE(*XXXX) TOLIB(library-name)
      ```

2. The RCLSTG command changed ownership of objects existing in user ASPs to QDFTOWN user profile. Transfer ownership of the objects in the user ASP library from QDFTOWN user profile to the correct user profile.

   a. Type the following and press the Enter key:

      ```
      WRKOBJOWN USRPRF(QDFTOWN)
      ```

      The Work with Objects by Owner display is shown.

   b. On the Work with Objects by Owner display, type a **9** in the *Opt* column for each object in the ASP library that you want to change ownership for.

   c. If all the objects will have the same owner, type the following on the command line of the Work with Objects by Owner display. Otherwise, continue with the next step.

      ```
      NEWOWN(owner-name)
      ```

Note: If you enter NEWOWN(owner-name) on the command line of the Work with Object by Owner display, you will not have to enter an owner name in the *New owner* prompt on the Change Object Owner display for each object.

d. Press the Enter key.

e. On the Change Object Owner (CHGOBJOWN) display, type the name of the new owner in the *New owner* prompt and press the Enter key. Repeat this step for all the objects that need the ownership changed.

3. Load the the SAVDLO tape, and then type the following to restore document library objects:

```
RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
       SEQNBR(beginning-number ending-number)
```

Note: If you have changed objects to restore or were journaling, do not run the RSTAUT command. Continue with "Task 3. Restoring Changed Objects and Applying Journal Changes."

4. Type the following to grant all private object authorities that existed when the system was saved:

```
RSTAUT
```

Task 3. Restoring Changed Objects and Applying Journal Changes

If you are using journaling and need to later apply journaled changes, continue with the following steps. Otherwise ignore these steps and continue with "Restoring Changed Objects."

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily save operations using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform the steps in "Working with Journals" for the system supplied journal QUSRSYS/QAOSDIAJRN. If you specified OBJJRN(*YES) on the SAVCHGOBJ command, you do not need to apply journal changes.

Working with Journals

1. Type the following and press the Enter key:

```
WRKJRN
```

2. The Specify Journal Name display is shown. Specify *ALL for the *Library name* prompt and press the Enter key.

3. The Work with Journals display is shown. Type a **5** in the *Opt* field for each journal that you want to apply journaled changes to display the name of the currently attached journal receiver. Write down the name of the journal and journal receiver.

   Note: If you are using OfficeVision/400 or PC Support/400, you must apply journaled changes to the system supplied journal QUSRSYS/QAOSDIAJRN.

4. You cannot restore journal receivers from the SAVCHGOBJ media if active journal receivers are attached. To later apply all journaled changes that have occurred since the last complete save operation, you must restore the receivers to the system from the SAVCHGOBJ media.

   For each journal identified in the previous step, do the following steps:

a. Create a journal receiver that will be used as a temporary receiver. Give it a name that will identify it as a temporary receiver, for example, TEMPnn. You can enter a description in the text (TEXT parameter) that identifies it as a temporary receiver for disaster recovery.

```
CRTJRNRCV JRNRCV(library-name/TEMPnn)
      TEXT('temporary journal receiver for journal xxx')
```

b. To detach the current receiver and attach the new TEMPnn receiver, type the following and press the Enter key.

```
CHGJRN JRN(library-name/journal-name) JRNRCV(library-name/TEMPnn)
```

c. Delete the detached journal receiver (identified in the step where you wrote down the name of the journal and journal receiver) using the Delete Journal Receiver (DLTJRNRCV) command. (This allows the journal receivers on the SAVCHGOBJ media to be restored successfully.

```
DLTJRNRCV JRNRCV(library-name/journal-receiver)
```

If you receive message CPA7025 *Receiver never fully saved*, enter an **I** to ignore and press Enter to continue the delete.

Restoring Changed Objects

Load the SAVCHGOBJ tape and enter the following to restore changed objects.

```
RSTOBJ OBJ(*ALL) DEV(tape-device) SAVLIB(library-name)
         OBJTYPE(*ALL) ENDOPT(*LEAVE) MBROPT(*ALL) ALWOBJDIF(*ALL)
```

You must repeat this step for every library saved using the SAVCHGOBJ command. If you specified SAVCHGOBJ LIB(*ALLUSR), type the following to determine the libraries that were saved:

```
DSPTAP *SAVRST
```

If you are using journaling, perform the steps in "Applying Journal Changes" for each journal you wish to apply journal changes to. Otherwise, continue with "Restoring Changed Documents and Folders."

Applying Journal Changes

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily saves using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform this step for the system supplied journal QUSRSYS/QAOSDIAJRN.

1. Determine the name of the last journal receiver (the last receiver restored) by entering the following:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

2. Press F15 (Work with receiver directory) to display the last receiver with a status of SAVED or PARTIAL. Write down the name of the receiver.

3. Create a receiver that follows the same naming convention as the last receiver but assign it a number of one greater.

   ```
   CRTJRNRCV JRNRCV(library-name/journal-receiver-nameNN)
   ```

   By doing this, you are doing what the CHGJRN command would normally do if the receiver was the current receiver being detached with a new receiver name being created. This allows your naming convention for journal receivers to continue.

Recovery

4. Use the CHGJRN command to detach the temporary receiver and attach the new receiver you just created.

```
CHGJRN JRN(library-name/journal-name)
      JRNRCV(library-name/journal-receiver-nameNN)
```

5. Determine the chain of receivers to be used in the APYJRNCHG command by entering the following command:

```
WRKJRNA JRN(library-name/journal-name)
```

   Press F15 (Work with receiver directory) to display the receivers. Write down the first and last receiver that you restored (last receiver is prior to the TEMPnn receiver). Notice that the first and last is the same receiver if only one journal receiver was restored.

6. When applying journal changes and the ending receiver has a status of PARTIAL (saved while attached), the TOENT parameter requires that a sequence number be specified on the APYJRNCHG command. Determine the last entry to be applied by entering the following command for the last receiver (identified in previous step):

```
DSPJRNRCVA JRNRCV(library-name/last-journal-receiver-name)
```

   Write down the value for the *Last Sequence Number* field.

7. Enter the following command to apply the journaled changes using the first and last journal receivers identified on the Work with Receiver Directory display.

```
APYJRNCHG JRN(library-name/journal-name)
          FILE((library-name/*ALL))
          RCVRNG(lib-name/first-receiver lib-name/last-receiver)
          FROMENT(*LASTSAVE) TOENT(last-entry)
```

Attention

Do not apply journal changes to the document and folder search index database files (QAOSSH10 through QAOSSH19) for journal QAOSDIAJRN in library QUSRSYS. You must specify individual files on the FILE parameter instead of *ALL.

```
APYJRNCHG JRN(QUSRSYS/QAOSDIAJRN)
          FILE((QUSRSYS/QAOSAH05) (QUSRSYS/QAOKLY02)
               (QUSRSYS/QAOSAH07) (QUSRSYS/QAOKLY03)
               (QUSRSYS/QAOKDYX4) (QUSRSYS/QAOKNY06)
               (QUSRSYS/QAOKDY01) (QUSRSYS/QAOKDY04)
               (QUSRSYS/QAOKDY05) (QUSRSYS/QAOKDY07)
               (QUSRSYS/QAOKDY08) (QUSRSYS/QAOKDY09))
          RCVRNG(lib-name/first-receiver lib-name/last-receiver)
          FROMENT(*LASTSAVE) TOENT(last-entry)
```

Restoring Changed Documents and Folders

If you performed daily save operations for documents and folders, do the following steps. Otherwise, continue with the RSTAUT command.

1. Load the last daily SAVDLO tape.

2. If you performed daily save operations to backup all new folders and new and changed documents since the last complete SAVDLO operation, type the following and press the Enter key.

```
RSTDLO DLO(*ALL) DEV(TAP01) SAVFLR(*ANY) ALWOBJDIF(*ALL)
```

If you saved all documents referred to by the mail using SAVDLO DLO(*MAIL), type the following and press the Enter key.

```
RSTDLO DLO(*MAIL) DEV(TAP01)
```

3. Restore users' authority by entering:

   ```
   RSTAUT
   ```

   The time it takes for the RSTAUT command to complete can vary significantly. The time depends on the number of user profiles and private authorities that were saved during the save operation.

4. This completes the restore operation.

   Power down the system by entering the following:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES)
   ```

   to perform a normal IPL and return the system to normal operations.

   Note: If you turned auto configuration off during the restore of the operating system, you must perform an IPL with the keylock switch in the **Normal** position.

This completes the steps to recover from a disk unit media failure in the system ASP with user ASPs configured.

Method 3. Recovering from a Disk Media Failure in a User ASP

This example describes two methods to recover from a media failure in a user ASP:

1. Option 1 is used to recover a user ASP if it was not in overflowed status at the time of the failure.
2. Option 2 is used to recover a user ASP if it was in overflowed status at the time of the failure.

These procedures assume that you have a backup tape or diskette containing saved objects and libraries.

If a disk unit failure occurs in a user ASP, as described above, you and your service representative will determine whether it should be treated as a disk unit media failure. If this is the case, the service representative will replace the failed disk unit and as a result, the system will clear all units in the user ASP where the unit failure occurred (all objects in the failed user ASP are destroyed). All unaffected user ASPs, as well as the system ASP, remain as they were before the media failure occurred.

After the service representative replaces the failed disk unit, do **one** the following:

Option 1. Recovering a User ASP that Was Not in Overflowed Status

```
                          IPL or Install the System

 Select one of the following:

         1. Perform an IPL
         2. Install the operating system
         3. Use Dedicated Service Tools (DST)
         4. Perform automatic installation of the operating system









 Selection
     1

 Licensed Internal Code is subject to the licensed
 granted in the Agreement for Purchase of IBM machines.
                                         © COPYRIGHT IBM CORP. 1980,
 1990.
```

1. Select option 1 (Perform an IPL) on the IPL or Install the System menu

   After IPL processing is complete, the command entry screen will appear and you can proceed to the next step.

2. Load the correct volume of the RSTLIB or RSTOBJ tapes.

3. Use the appropriate restore command to restore the desired objects to the ASP:

   ```
   RSTLIB SAVLIB(library-name) DEV(TAP01)
   ```

   Or:

   ```
   RSTOBJ OBJ(*ALL) SAVLIB(library-name) OBJTYPE(*ALL)
          ENDOPT(*REWIND) MBROPT(*ALL)
   ```

4. If any objects were restored into a different library than the one from which they were saved, manually grant private authority for the restored objects with the EDTOBJAUT command.

   Notes:

   a. If the journals are in the system ASP, create a new receiver in the user ASP, and run the CHGJRN command to attach the new journal.

   b. If the user ASP contained journals, and you did not delete the old files and then restore them from the media, create a new journal receiver and the journal in the user ASP. Use the Start Journal Access Paths (STRJRNAP) and the Start Journal Physical Files (STRJRNPF) commands to start journaling again.

   c. If the user ASP contained journals and you did not delete the old files and then restore them from the save media, you must delete and restore the files that were journaled which will start journaling again.

5. This completes the steps to recover from a disk unit media failure in a user ASP.

6. Saving of the libraries in the ASP is recommended.

Option 2. Recovering a User ASP that Was in Overflowed Status

1. Select option 1 (Perform an IPL) on the IPL or Install the System menu

   After IPL processing is complete, the command entry screen will appear and you can proceed to the next step.

2. End all subsystem using the following command:

   ```
   ENDSBS *ALL *IMMED
   ```

3. Type the following to reclaim storage:

   ```
   RCLSTG
   ```

4. Delete the objects that were placed in the user ASP but may still be known to the system after reclaim processing. Although the initial disk extents for such objects may have been allocated in the system ASP, portions of the objects may still have been lost. Therefore the integrity of the object cannot be predicted.

   If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:

   a. Display the library in the user ASP:

      ```
      WRKLIB LIB(library-name)
      ```

   b. Type a **12** (Work with objects) in the *Opt* column and press the Enter key.

   c. Find the objects to be deleted in the *Object* column.

   d. Type a **4** (Delete) in the *Opt* column for each object you want to delete.

   e. Press the Enter key.

5. Load the correct volume of the RSTLIB or RSTOBJ tapes.

6. Use the appropriate restore command to restore the desired objects to the ASP:

   ```
   RSTLIB SAVLIB(library-name) DEV(TAP01)
   ```

   Or:

   ```
   RSTOBJ OBJ(*ALL) SAVLIB(library-name) OBJTYPE(*ALL)
          ENDOPT(*REWIND) MBROPT(*ALL)
   ```

7. If any objects were restored into a different library than the one from which they were saved, manually grant private authority for the restored objects with the EDTOBJAUT command.

   Notes:

   a. If the journals are in the system ASP, create a new receiver in the user ASP, and run the CHGJRN command to attach the new journal.

   b. If the user ASP contained journals, and you did not delete the old files and then restore them from the media, create a new journal receiver and the journal in the user ASP. Use the Start Journal Access Paths (STRJRNAP) and the Start Journal Physical Files (STRJRNPF) commands to start journaling again.

   c. If the user ASP contained journals and you did not delete the old files and then restore them from the save media, you must delete and restore the files that were journaled which will start journaling again.

8. This completes the steps to recover from a disk unit media failure in a user ASP.

Part 5. Checksum Protection

Chapter 17. Using Checksum Protection

Checksum protection is an option you can use to recover from a disk unit media failure in an ASP without having to reload the entire system. This option minimizes your vulnerability to data loss and reduces recovery time from disk unit media failures.

Note: When you start checksum protection for an ASP, all storage units in the ASP are cleared. Therefore, it is important to understand checksum protection and how it might fit in your recovery strategy before you start using it.

Checksum protection should not be used as a replacement for system backup procedures. You should continue to back up your system on a regular basis because there are instances where checksum protection cannot be used to reconstruct the lost data.

How Checksum Protection Works

When checksum protection is configured, the system automatically groups the disk units in an ASP into checksum sets. Space equivalent to approximately one disk unit in each set is used to store checksum data that provides protection for the user data stored on the other units in the set. Checksum protection requires additional storage capacity (both main storage and auxiliary storage), as well as added processing unit capacity to achieve comparable performance to a system without checksum protection.

Checksum protection takes the data residing on a storage unit and distributes the data onto other storage units in the checksum set. The data is distributed in such a way that if any one of the units in the checksum set fails, its contents may be recovered by recombining the data on remaining units. Checksum protection operates like a parity bit. The parity bit is normally set based on the contents of a byte of data and allows the recovery of any bit within the byte that cannot be read. Checksum protection operates in the same manner except that the parity is dependent on input from more than one disk unit.

When a disk unit fails, the system becomes unusable regardless of whether checksum protection is active or not. Checksum protection does not prevent the system from ending abnormally. Rather, its key advantage is that it helps avoid restoring the licensed internal code, the operating system, and user data if the data on a failed unit is lost and the unit must be replaced.

When you first establish the checksum environment in the system ASP, you must restore the entire system. When starting checksum protection in a user ASP, you must save and then restore all libraries in that user ASP. Thereafter, any changes you make to permanent objects are automatically updated and maintained in the checksum data (associated with the disk extents allocated to the objects). If any single storage unit in a checksum set is lost, the system reconstructs the contents of the lost storage unit after it is replaced. The reconstructed data reflects the most up-to-date information on the disk at the time of the failure.

Figure 17-1 on page 17-2 illustrates a system ASP configuration with checksum protection. In this example, assume that two of the three devices contain user application data, and the other device contains checksum data.

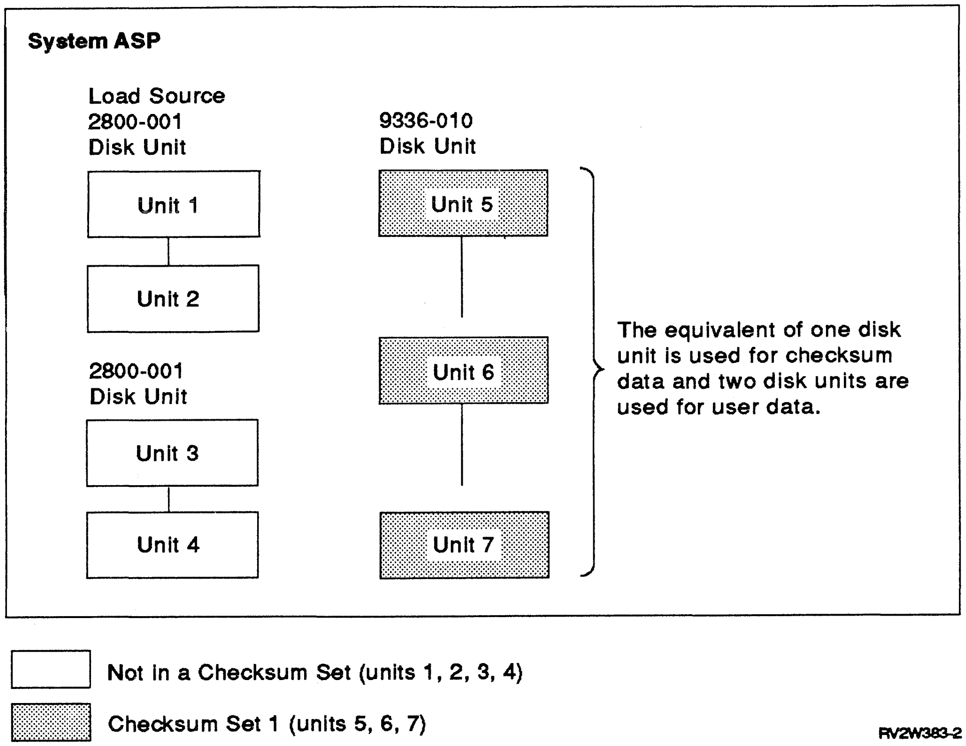


Figure 17-1. System ASP with Checksum Protection

The assignment of storage units to checksum sets is shown for example purposes only. The actual assignment of checksum sets can vary.

The implementation of checksum protection is more complex because there are multiple checksum areas spread across all units in a checksum set to distribute the disk activity more evenly over all units in a checksum set.

If the user data on storage unit 6 is lost, the system automatically reconstructs that data from the checksum data on storage units 5 and 7 after unit 6 is replaced.

Additional Considerations

Checksum protection gives you the advantage of being able to recover from a disk unit failure, but there is a performance and resource cost. You may need to do one or more of the following before you start using checksum protection:

- Add more disk units.
- Add more main storage.
- Upgrade to a higher performance processor model.

Checksum Recovery Limitations:

Although checksum protection can fully recover from several disk unit media failures, it is not a substitute for save procedures. There are some types of disk unit failures that do require backup media. For example:

- Checksum protection cannot recover data if a second storage unit in the same checksum set fails before the data from the first failure is fully recovered.

  In this case, all units in the ASP must be cleared. You must be prepared to restore the ASP from your backup media and apply your journal changes. For more information about how to reload the system, see "Recovering from Disk Unit Media Failures" on page 16-3.
- If the system abnormally ends without saving main storage, disk sectors affected by I/O operations in progress may not be recovered, and you may have some damaged objects.

After the system recovers data using checksum protection, message queue QHST contains messages identifying any objects that could not be completely recovered. This is especially useful if a power loss prevents the saving of main storage.

System Performance When Using Checksum Protection: When checksum protection is active, additional storage capacity (both main and auxiliary storage) and added processing unit capacity are required to achieve performance comparable to a system without checksum protection. Even with the addition of more resources to your system, you can still experience a decrease in performance with checksum protection.

Because of the additional requirements, you must plan carefully to determine the additional system resources that are required to get a similar level of performance. Planning for additional auxiliary storage requirements should be done using the information provided later in this chapter. Use the capacity planning tool to plan for additional main storage and requirements processing which is a part of the AS/400 Performance Tools package. Using the *Performance Tools/400 Guide*, you can determine the effect checksum protection does have on system performance. Use MDLSYS, the tool's capacity planning help, to predict the effect that checksum protection will have.

In addition to the capacity planner, the following are some general guidelines that can help determine what effect checksum protection does have on main storage and processor utilization. However, these guidelines should not be used as a replacement for the capacity planning tool. The capacity planning tool should still be your primary source for obtaining information about checksum protection.

- Checksum protection uses approximately 5% of main storage for internal operations. This amount should be added to the machine pool size. In addition, checksum protection uses an additional 40KB of main storage for each storage unit that is part of a checksum set.
- Checksum protection requires at least 10% additional system processing power or increased system utilization. Because checksum protection can slow system performance, you may need additional processing capacity.

  For example, if you monitor your current processor usage and find that it is running at approximately 60% of its capability during heavy usage, you can expect to add at least another 10% with checksum protection. Your system would run at 70% of its processing capability during heavy usage with checksum protection specified. This percentage leaves little room for additional usage or growth. In this situation, you may decide a faster processing unit is needed.

It is important to understand that, even with the addition of more resource to your system, you can still experience a decrease in performance because of checksum protection. The amount can vary for each job, depending on how many disk write operations are done by the job. The more write operations done by a job, the greater the decrease in performance for the job when checksum protection is in effect. Some of the operations that generally experience a decrease in performance are:

- Extensive updating of the database
- Journaling
- Commitment control
- Copying a file (Copy File (CPYF) command)
- Restoring
- Saving to save files
- Replacing edited source members
- Reorganizing physical file members (Reorganize Physical File Member (RGZPFM) command)

Because performance is reduced as the number of write operations increases, a restore of the system ASP or of all the libraries in a user ASP can take two to three times longer than your normal restore time. When planning to implement checksum protection, make sure you add the extra time into your estimate of setting up checksum protection.

It is recommended that you implement user ASPs without checksum protection for journaling, commitment control, and saving to save files. Because checksum protection is done by ASP, the unprotected ASPs will not be affected by the performance impact of checksum protection. Journals, journal receivers, and save files are all forms of protection from data loss. It is not necessary to protect an ASP containing these objects using checksum protection.

Batch jobs must also be given special consideration when implementing checksum protection. Because batch jobs are often disk input/output intensive, they can perform many more write operations than an interactive job. Therefore batch jobs can experience a significant decrease in performance. Batch jobs can take two to three times longer, depending on the number of write operations. To predict more accurately the effects of checksum protection for interactive and batch jobs, use the *Performance Tools/400 Guide*.

Additional IPL Time: In some cases, if checksum protection is active and the last system end was abnormal, the system cannot be sure that the most up-to-date information was written to disk before the system ended. In that case, the system does read all the data in the ASP to verify checksum data. This validation process occurs during the IPL following the abnormal end, and it can be a lengthy process. If the system can save a copy of main storage before it ends, the lengthy checksum validation process may be avoided.

If the system loses power frequently, it is strongly recommended that you install an uninterruptible power supply when you start checksum protection. Should main power be lost, an uninterruptible power supply allows the system to stop normally, preventing most long checksum validations.

Planning Storage Capacity

This section provides information on your main storage requirements, understanding your current storage use, determining the amount of unprotected and protected storage you need, and disk unit requirements.

Main Storage Requirements

Checksum protection uses approximately 40 000 bytes of main storage for each storage unit in a checksum set.

Adding additional main storage may lessen, but not avoid, slower performance. Generally, operations that do several write operations (such as updating database records) slow system performance on a checksum-protected system unless extra main storage is added.

Moving your journal receivers or selected user libraries to an unprotected or mirror protected user ASP can reduce the number of write operations required for checksum protected data. This reduces the amount of main storage required to provide for acceptable performance. However, it is not recommended to use both checksum and mirrored protection on the same system.

Understanding Current Storage Use

After checksum protection is started, portions of auxiliary storage are used for checksum redundancy information, reducing the overall storage capacity of your disk configuration. Accordingly, you must understand how much auxiliary storage space you are using for your normal operations before you start checksum protection.

Auxiliary storage for the system ASP with checksum protection is divided into two distinct areas of protected storage (for permanent data) and unprotected storage (for temporary data):

- Protected storage contains permanent objects, such as database files and program objects.
- Unprotected storage contains machine data and temporary objects, such as file open data paths and compiler work areas.

Note: When checksum protection is started for the system ASP, you define the amount of unprotected storage for each storage unit in a checksum set.

Auxiliary storage in a user ASP that has checksum protection does not have this division of protected and unprotected storage. All available storage is protected storage.

Before configuring checksum protection for an ASP, monitor your system for a representative period to determine the total amount of protected and unprotected storage you are currently using. To determine the amount of unprotected storage, see the topic "Determining the Amount of Unprotected Storage You Need for the System ASP" on page 18-4.

The value shown on the Work with System Status display when calculating the maximum amount of unprotected storage allocated since the last IPL is useful only if it is a representative example of your system's work. Also consider any special applications you may run, as well as peak activity needs, when planning for your system storage.

It is recommended that, when deciding how to configure your system, you allow a large buffer of space for unprotected storage for growth. It is better to overestimate the amount of space needed. If you incorrectly estimate the amount, it is easier to decrease the unprotected storage space than it is to increase it. Increasing the amount causes the ASP to be cleared of all data. You must save and restore all data in the ASP in this case.

How Unprotected Storage Is Used

To determine the amount of space to allocate for unprotected storage, you must first consider how the space is to be used.

Note: Unprotected storage has meaning only for the system ASP that has checksum protection.

Unprotected storage is used for:

- Running jobs. The Work with Active Jobs (WRKACTJOB) display shows, at the top of the display, the number of jobs currently active in the system. Each job requires unprotected storage for internal work areas. You can display this value for a job using the Work with Jobs (WRKJOB) command and selecting the Job Run Attributes display. The temporary (unprotected) storage used value appears at the bottom of the Job Run Attributes display, and is shown as kilobytes.

  Sample several jobs to determine the averages for your system. Certain jobs (for example, certain office products) may require a large amount of temporary storage.

- Storing machine data. Unit 1 is used mainly for licensed internal code and machine data. The entire size of unit 1 is included in the unprotected amount used values shown on the Work with System Status (WRKSYSSTS command) display (when checksum protection is not active).

Your *unprotected storage* allocation should be the total of:

- The amount of unprotected storage already used immediately after you do an IPL, and before other users sign on.
- The average amount of temporary (unprotected) storage used per job multiplied by the total number of jobs you expect to support.

To verify that the value you calculate is appropriate for your needs, use the Work with System Status (WRKSYSSTS) command to show the current amount of unprotected storage used and the maximum amount of unprotected storage used since the last IPL of the system.

In addition to growth and peak levels of activity, also consider whether you now use, or plan to use, group jobs, advanced program-to-program communications (APPC), Structured Query Language/400 (SQL/400*), AS/400 Query, or the dynamic selection (DYNSEL) attribute on any of your files. Group jobs and APPC increase the number of active jobs for your system. SQL/400, Query, and the DYNSEL attribute increase the amount of temporary storage used by the system.

Disk Unit Requirements

After you have assessed your storage capacity requirements, you must determine the equipment you need to satisfy those requirements. Use the SST Calculate Checksum Configuration display to determine how much protected (permanent) and unprotected (temporary) storage in the system ASP will be available for a given disk configuration when checksum protection is started. For more information about doing this, see the topic "Calculating a Disk Configuration for Checksum Protection" on page 18-4.

You can evaluate whether the amount of auxiliary storage on your system is sufficient to meet your permanent and temporary storage requirements. If your current disk configuration does not support a checksum environment, use this option to help determine the disk configuration that does support that environment.

Using the checksum protection function requires that you have enough storage for the checksum data area for each checksum set. The amount of additional storage capacity required depends on the size and number of the checksum sets created. Each checksum set requires the equivalent of approximately one unit for checksum protection purposes.

For example, a checksum set with three storage units requires the equivalent storage of one of those units to be used for the checksum data. Therefore, only 67 percent of the storage space is actually available for protected and unprotected storage. The other 33 percent of the storage space contains the checksum data.

Table 17-1 illustrates the space available for protected and unprotected storage as additional storage units are added.

Table 17-1. Space Available per Number of Units

Number of Storage Units in a Checksum Set	Storage Available per Checksum Set	Checksum Storage Used per Checksum Set
2	50%	50%
3	67%	33%
4	75%	25%
5	80%	20%
6	84%	16%
7	86%	14%
8	88%	12%

The number of storage units in a checksum set is limited by:

- The requirement that each checksum set contains from two to eight units.
- The similarity of the disk unit sizes. Each storage unit in the checksum set must be of the same size. For example, you could have one checksum set with four 9332 model 200 disk unit. You cannot, however, have a checksum set with one 9332 and one 9335 disk unit.

  The 9332 models 200 and 400 disk units can be in the same checksum set. You can have multiple sets of different disk unit types (for example, you can have three 9332 units in a checksum set and four 9335 units in a different checksum set in the same ASP).
- The restriction that no two storage units of a checksum set can be on the same replaceable disk unit applies to the 9332 and 9335 disk units only.

  Note: Storage units in the same replaceable disk unit have the same serial number. Therefore, you should verify that both storage units in the same replaceable disk unit are assigned to the same ASP.
- The restriction that unit 1 and all other 2800 model 001 units that contain the licensed internal code and data areas are not eligible for checksum protection.

For example, if you currently have one 9335 (two storage units total), you need at least one additional 9335 disk unit to begin checksum protection. This additional disk unit must be added because a minimum of two units must be in each checksum set, and because no two units can be on the same replaceable disk unit.

The 9336 model 010 disk unit contains from two to four storage units that can be replaced separately within the same disk unit. Therefore, the storage units within the disk unit can be in the same checksum set.

Because the 2800 model 001 units cannot be placed in a checksum set, the additional capacity they provide (beyond that needed for the licensed internal code and data areas) can only be used for the storage of unprotected (temporary) objects. The 2800-001 units that are not used for the licensed internal code and data areas can be assigned to a user ASP.

When checksum protection is in effect, storage units in the system ASP that are not in a checksum set are used for storage of unprotected data. For example, if you have only one disk unit of a particular type, the disk unit is not checksum protected. Only unprotected data (such as temporary objects) is stored on that device.

Storage units that are not in a checksum set in a user ASP that has checksum protection are never used. These storage units should be either moved or removed from the user ASP. Before this type of configuration is allowed to occur, a warning is sent informing you of this situation.

Disk Configuration for Checksum Protection

You can use the SST Calculate Checksum Configuration display to calculate the amount of protected and unprotected storage capacity that would be provided if checksum protection were in effect for a particular disk configuration. You can calculate the storage capacity of your current system configuration or the storage capacity of additional disk units that you may add later by using a different configuration.

To calculate disk storage capacity, see the topic "Calculating a Disk Configuration for Checksum Protection" on page 18-4.

Examples of Checksum Protection Configurations

The following examples are provided to illustrate how different combinations of disk units can result in different checksum set configurations. The examples do not necessarily match the configuration put into effect on your system for the same set of disk units because the system uses an algorithm to determine the checksum sets. The unit numbers may be different.

Example of Checksum Configuration Using Three Checksum Sets: Assume your system ASP has two 2800 model 001 disk units configured, three additional 9335 disk units, and a 9336 model 010 disk unit configured for a total of fourteen storage units. After checksum protection is configured, the system ASP could have the configuration shown in Figure 17-2.

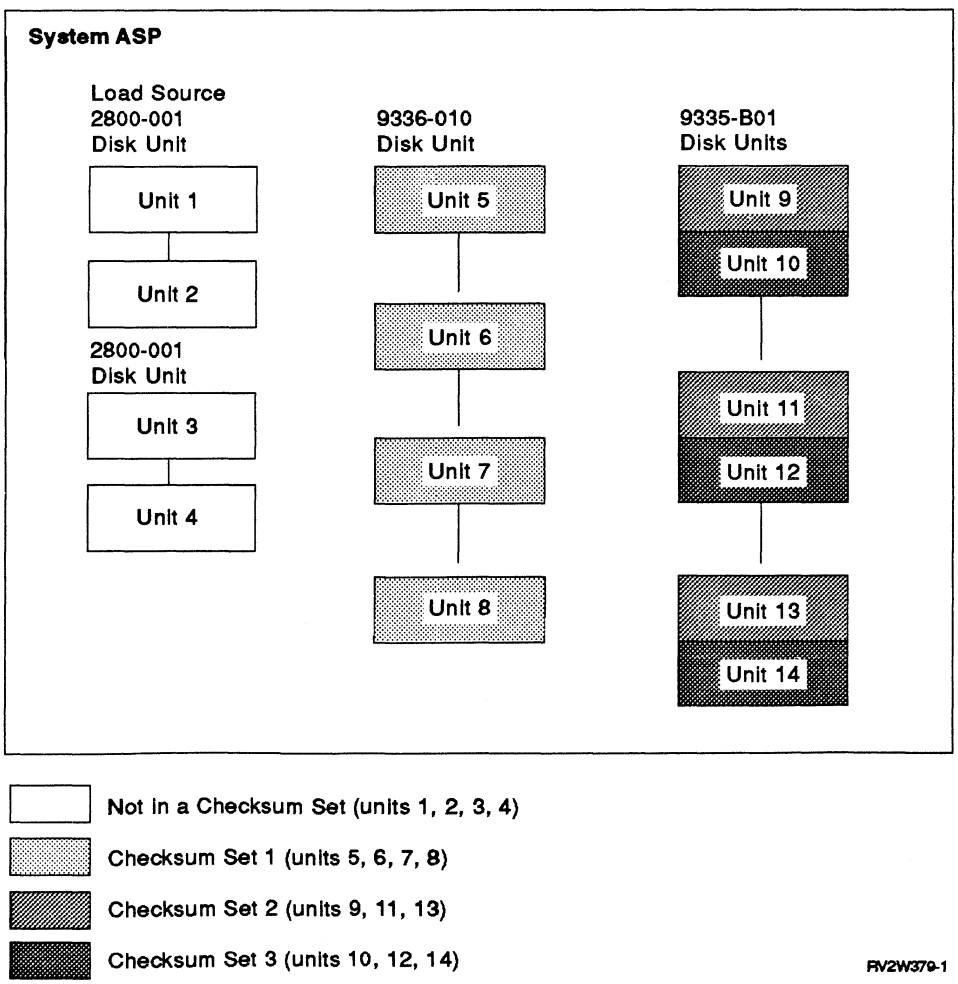


Figure 17-2. Example of Checksum Configuration Using Three Checksum Sets

Storage units 5, 6, 7, and 8 are configured in checksum set 1. and storage units 9, 11, and 13 are configured in checksum set 2, and storage units 10, 12, and 14 are configured in checksum set 3. Thus, if the data in storage unit 12 is lost, the system automatically reconstructs the lost data from the checksum data on storage units 10 and 14. The 2800 Model 001 storage units (1, 2, 3, and 4) that contain the licensed internal code and data areas cannot be placed in a checksum set.

Example of Checksum Configuration Using Two User ASPs: Assume your system ASP has two 2800 Model 001 disk units configured, that have a total of four storage units to contain the licensed internal code and data areas, and has a 9336 disk unit configured, with 4 storage units, for a total of 8 storage units.

Also, assume that you have three additional 9335 Model B01 and 9332 Model 400 disk units configured into two user ASPs. Assume ASP 3 contains a journal object, journal receivers, or one or more save files. ASP 2 contains other object types allowed in user ASPs.

The system could have the configuration shown in Figure 17-3 on page 17-9.

Placing the journal receiver in a user ASP gives you two performance benefits:

- If extensive journaling is done, journal performance is better because the actuator arm is dedicated to the journal receiver.
- Write operations to a user ASP without checksum protection do not have the performance overhead of checksum protection.

Placing the save file in a user ASP also reduces the number of write operations in the system ASP that has checksum protection.

Dual journal receivers can further improve the recovery capabilities. If you use this approach, the best performance occurs if the active journal receivers are in different user ASPs that do not have checksum protection. The best recovery occurs if the active journal receivers are not on the same replaceable disk unit.

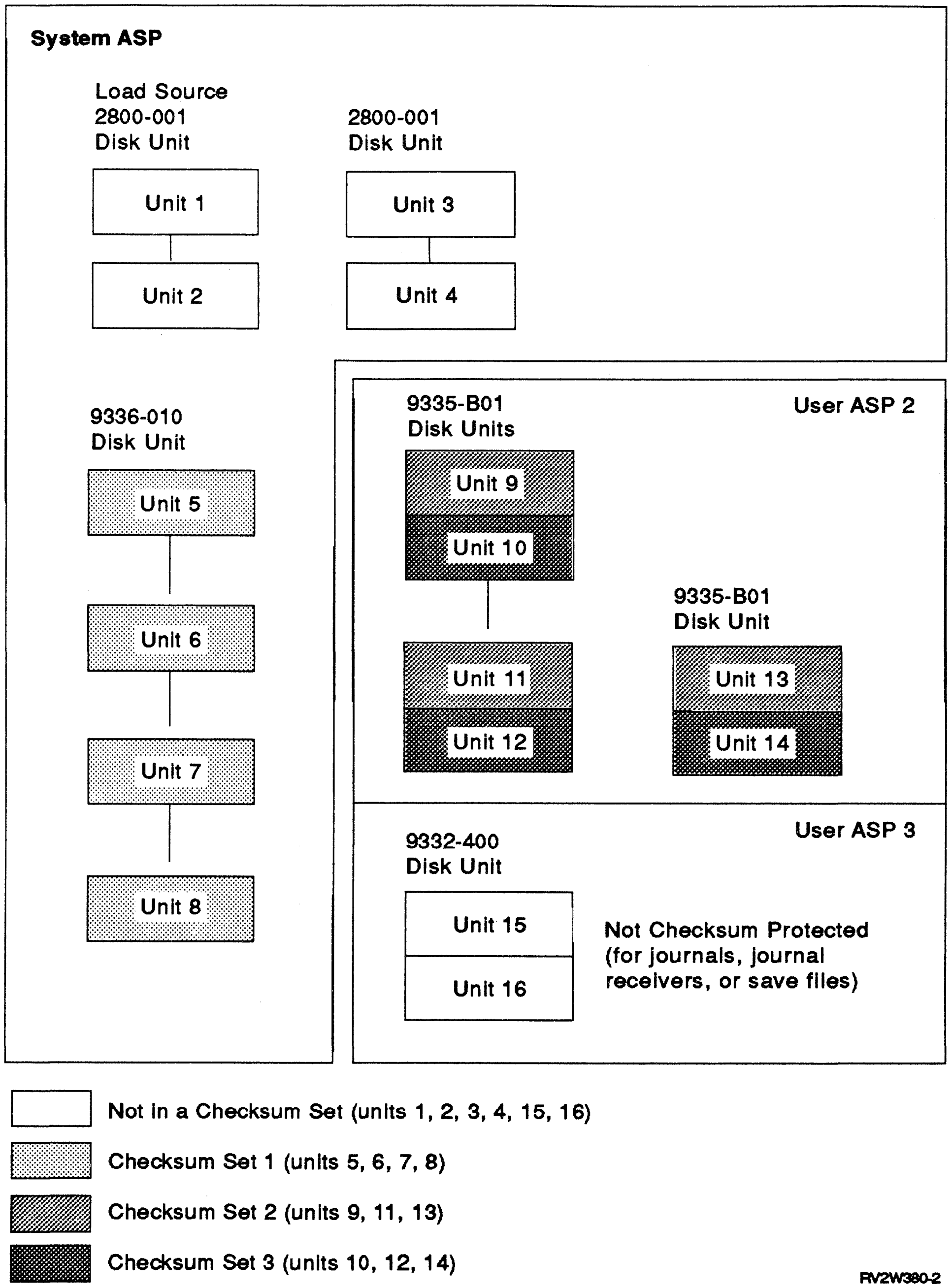


Figure 17-3. Example of Checksum Configuration Using Two Checksum Sets with Two User ASPs

Example of an Ineligible Configuration for Checksum Protection: Assume your system ASP has the required 2800 model 001 units. In addition, you have the following configured disk units:

- One 9332 Model 400 disk unit (two storage units)
- One 9335 B01 disk unit (two storage units)

This configuration shown in Figure 17-4 on page 17-10 is not eligible for checksum configuration because you must have at least two disk units containing the same sized storage units allocated to your system ASP. To qualify for checksum protection, add a second 9335 or 9332 disk unit.

The 2800-001 units cannot be in a checksum set because they are reserved for use by the system for the licensed internal code, dump space, and so forth.

Example of an Ineligible Configuration for Checksum Protection

If you try to configure the units in this example, you will receive an error message when you use the DST Start Checksum Protection option.

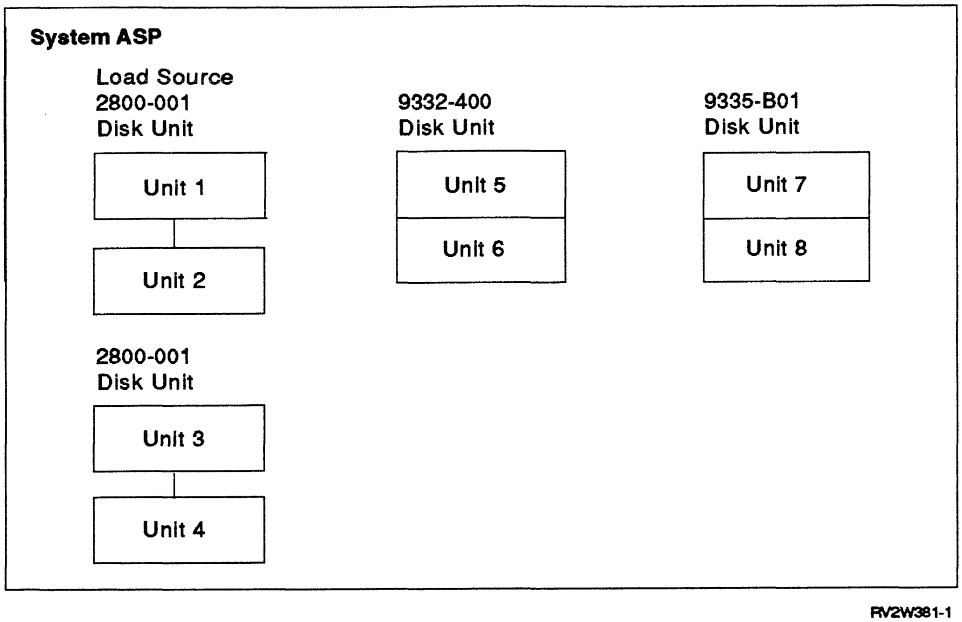


Figure 17-4. Example of an Ineligible Checksum Configuration

Example of an Inefficient Checksum Configuration: Assume your system has the required 2800 model 001 storage units configured for the licensed internal code and data areas in addition to the following:

In ASP 1:

- Three 9332 Model 400 disk units (six storage units)
- One 9335 Model B01 disk unit (two storage units)

In ASP 2:

- One 9332 Model 400 disk unit (two storage units)
- One 9332 Model 200 disk unit (one storage unit)

Although this is a valid checksum configuration, this is not efficient use of storage because the space on the 9335 model B01 disk unit is used only for unprotected storage (your permanent user objects are not stored on the 9335 disk unit). Also, the 2800 model 001 storage units that are not used for the licensed internal code, data areas, and dump space provide unprotected storage. This can be inefficient in disk accessing performance because it can allow too much temporary data in one storage unit, risking contention. You could let the system use the space for unprotected storage, or you could place the 9335 disk unit into a separate user ASP.

In Figure 17-5 on page 17-12, unit 14 in the user ASP is not included in the checksum set because it is in the same replaceable disk unit as storage unit 13, in checksum set 3. There are no permanent user objects or temporary data stored on unit 14. You will get a warning message if you attempt to configure a user ASP with a storage unit that cannot be in a checksum set. If your configuration is already in this state, you may remove the unit using DST without clearing the ASP.

Example of an Inefficient Checksum Configuration

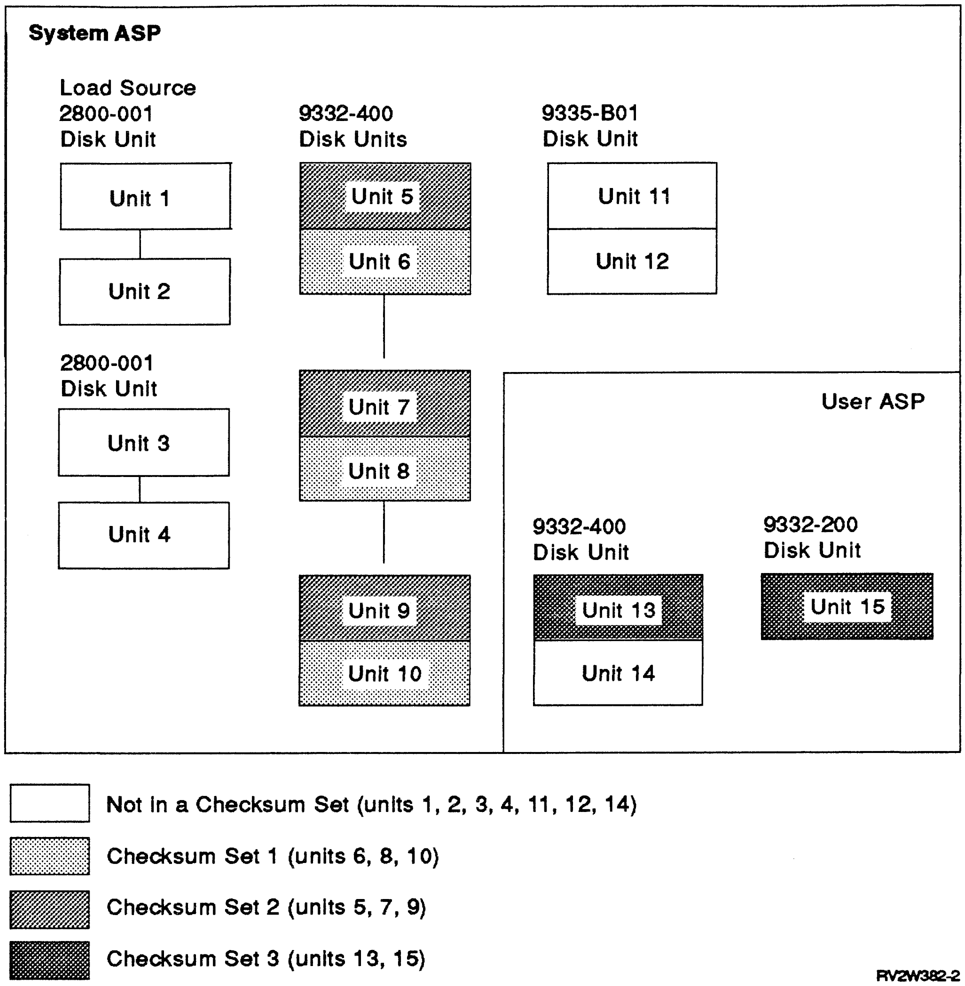


Figure 17-5. Example of an Inefficient Checksum Configuration

Changing an Existing Checksum Configuration

When checksum protection is in effect, you can change your existing configuration in several ways. Some of the changes destroy the ASP data and require that you save and restore the entire ASP. Other changes can be made without having to do this, although, any time you perform configuration changes to your system, you should save your system to protect against unanticipated problems or errors that can occur during the reconfiguration.

The following changes do not require that you save and restore the ASP:

- Decreasing the amount of unprotected storage in the system ASP
- Adding units to the system ASP or a user ASP while checksum protection is in effect.
- Moving a storage unit that is not in a checksum set from an ASP that has checksum protection to a different ASP, or removing such a unit from an ASP that has checksum protection.
- Replacing one storage unit in a checksum set with a nonconfigured unit.

The following changes do require that you save and restore the system ASP:

- Increasing the amount of unprotected storage in the system ASP destroys the data in the system ASP
- Moving a unit that is in a checksum set from an ASP with checksum protection to another ASP, or removing such a unit from an ASP with checksum protection destroys the data in the ASP.

Decreasing the Amount of Unprotected Storage in the System ASP

Assume your current unprotected storage amount is 10MB per unit, and you want to decrease it to 9MB. To do this use the DST Change Unprotected Storage for ASP1 option and specify the new value of 9 for the unprotected storage per unit value.

Chapter 18. Working with Checksum Protection

Before configuring checksum protection, monitor your system for a representative period to determine the total amount of protected and unprotected storage you are currently using:

1. Enter the Work With System Status (WRKSYSSTS) command to monitor system storage. (For additional information on this command, refer to the *CL Reference*.) This display shows the storage capacity in megabytes. Megabytes is represented by M on the display.

   - *System*: The capacity of auxiliary storage in the system ASP.
   - *% Used:* The current percentage of system ASP storage capacity that is allocated.

     When checksum protection is not in effect, these fields relate to the capacity in the system ASP for storage of both permanent and temporary objects.

     When checksum protection is in effect, these fields relate to the capacity in the system ASP for the storage of permanent objects only.
   - *Current unprotected used*: All temporary objects and machine data currently used on the system.
   - *Max unprotected*: The maximum amount of unprotected storage allocated since the last IPL of the system.

2. Use the values in the unprotected used fields to determine your current storage use for what would be unprotected storage under checksum protection:

   - When checksum protection is not in effect, these fields show the amount of unprotected storage that would be used if checksum protection were in effect.
   - When checksum protection is in effect, these fields show the amount of unprotected storage actually used.

   The value shown for the maximum amount of unprotected storage allocated since the last IPL is useful only if it is a representative example of your system's work. Also consider any special applications you may run as well as peak activity needs when planning for your system storage.

It is recommended that, when deciding how to configure your system, you allow a large buffer of space for unprotected storage for growth. It is better to overestimate the amount of space needed. If you incorrectly estimate the amount, it is easier to decrease the unprotected storage space than it is to increase it. Increasing the amount causes the ASP to be cleared of all data. You must save and restore all data in the ASP in this case.

Warning: When considering the removal of units from the system ASP (ASP 1), remember to leave at least two 2800-001 storage units in the system ASP for the 9406 model D70 and D80 system units that contain 224MB or more of main storage.

Determining the Amount of Protected Storage You Need for Checksum Protection in the System ASP

In the following example, the system ASP storage value is 1803M, 60.63% of the storage is being used, and 262M is the current amount of unprotected storage being used.

To determine the amount of protected storage needed, do the following:

1. Look at the Work with System Status (WRKSYSSTS command) display.

```
                          Work with System Status              RCH38342
                                                     02/26/90  14:31:46
% CPU used . . . . . . . :       29.5  Auxiliary storage:
Elapsed time . . . . . . :   00:00:00   System . . . . . . . . . :  1803 M
Jobs in system . . . . . :        159   % used . . . . . . . . . : 60.6241
% addresses used:                       Total  . . . . . . . . . :  1803 M
  Permanent  . . . . . . :      2.125   Current unprotect used :     262 M
  Temporary  . . . . . . :       .304   Maximum unprotect  . . :     265 M

Type pool size and activity level changes, press Enter.

System    Pool    Reserved    Max   -----DB-----  ---Non-DB---
 Pool   Size (K)  Size (K)  Active  Fault  Pages  Fault  Pages
   1       9183      4392     +++     .0     .0     .0     .0
   2       4000         0       6     .0     .0     .0     .0
   3      35619         0      59     .0     .0    7.0   16.3
   4        350         0       5     .0     .0     .0     .0

                                                                   Bottom
Command
===>
F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve     F10=Restart
F11=Transition data   F12=Cancel   F14=Subsystems   F24=More keys
```

2. Determine the system ASP storage amount from the value in the *System* field (1803M), and multiply that value by the value in the *% used* field. The *% uses* is the storage currently used.

```
WRKSYSSTS---System ASP Storage  =    1803M
               Storage being used  =    x60.63
                                        ______
                                        109,316
```

3. Divide the value calculated in step 2 by 100.

```
Divide 109,316 by 100          =    1093.16

Round this value to            =    1093
```

4. Look at the Work with Disk Status (WRKDSKSTS command) display.

```
                              Work with Disk Status              RCH38342
                                                        02/26/90  14:30:54
Elapsed time:    00:00:00

            Size      %    I/O   Request  Read  Write   Read  Write    %
Unit  Type   (M)   Used   Rqs  Size (K)   Rqs    Rqs    (K)    (K)   Busy
  1   9332   200   98.2   .0        .0    .0     .0     .0     .0     0
  2   9332   200   57.4   .0        .0    .0     .0     .0     .0     0
  3   9332   200   58.6   .0        .0    .0     .0     .0     .0     0
  4   9332   200   58.0   .0        .0    .0     .0     .0     .0     0
  5   9332   200   59.0   .0        .0    .0     .0     .0     .0     0
  6   9332   200   57.0   .0        .0    .0     .0     .0     .0     0
  7   9332   200   59.8   .0        .0    .0     .0     .0     .0     0
  8   9332   200   48.7   .0        .0    .0     .0     .0     .0     0
  9   9332   200   48.6   .0        .0    .0     .0     .0     .0     0



                                                                      Bottom
Command
===>
F3=Exit   F5=Refresh   F12=Cancel   F24=More keys
```

5. Determine the size of the load source unit (unit 1).
6. Subtract the value of unit 1 from the value in the *Maximum unprotected used* field on the Work with System Status display.

```
Maximum unprotected storage use =   265M
Size of unit 1                  = - 200M
                                    ____
Net unprotected storage         =    65M
```

7. Subtract the value in step 6 from the value calculated in step 3, since the *% used* value includes the total amount of protected and unprotected storage when checksum protection is not in effect.

```
Total storage used              =   1093M
Net unprotected storage used        - 65M
                                    _____
Protected storage used              1028M
```

8. Multiply this value by a factor (for example, 1.5) to allow for growth and peak storage needs. You may want to use a larger value, depending on your future storage needs.

```
Protected storage used              1028M
Buffer for future growth            x1.5
                                    _____
Protected storage needed            1542M
```

Determining the Amount of Protected Storage You Need for Checksum Protection in a User ASP

User ASPs that have checksum protection do not reserve unprotected storage for temporary objects. You only need to calculate the protected storage needed in the ASP.

Use the Display Disk Configuration Capacity option in SST or DST to determine the amount of storage currently used in the user ASP. To allow for growth and peak storage needs, multiply this value by a factor (for example 1.5), depending on your future storage needs. Use this value when calculating a disk configuration for checksum protection for a user ASP.

Determining the Amount of Unprotected Storage You Need for the System ASP

To determine the amount of space to allocate for unprotected storage, you must first consider how the space is to be used. Unprotected storage is used for:

- Running jobs. The Work with Active Jobs (WRKACTJOB) display shows, at the top of the display, the number of jobs currently active in the system. Each job requires unprotected storage for internal work areas. You can display this value for a job using the Work with Jobs (WRKJOB) command and selecting the Job Run Attributes display. The temporary (unprotected) storage used value appears at the bottom of the Job Run Attributes display, and is shown as kilobytes.

  Sample several jobs to determine the averages for your system. Certain jobs (for example, certain office products) may require a large amount of temporary storage.

- Storing machine data. The unit 1 is used mainly for licensed internal code and the system data. The entire size of the unit 1 is included in the unprotected used values shown by WRKSYSSTS when checksum protection is not active.

Your *unprotected storage* allocation should be the total of:

- The amount of unprotected storage already used immediately after you IPL your system, and before other users sign on.
- The average amount of temporary (unprotected) storage used per job multiplied by the total number of jobs you expect to support.

To verify that the value you calculate is appropriate for your needs, use the Work with System Status (WRKSYSSTS) command to show the current amount of unprotected storage used and the maximum amount of unprotected storage used since the last IPL of the system.

Calculating a Disk Configuration for Checksum Protection

Use the SST/DST Calculate Checksum Configuration display to calculate the amount of protected and unprotected storage capacity that would be provided if checksum protection were in effect for a particular disk configuration. You can calculate the storage capacity of your current system configuration or the storage capacity of additional disk units that you may add later by using a different configuration.

To calculate disk storage capacity, perform the following steps:

1. After you have determined your protected and unprotected storage needs (see "Determining the Amount of Protected Storage You Need for Checksum Protection in the System ASP" on page 18-2 and "Determining the Amount of Unprotected Storage You Need for the System ASP"), use the option 5 (Calculate checksum configuration display) on the Work with Checksum Protection display to display the capacities of protected and unprotected storage that would be provided by the disk units currently in your system ASP, or a set of disk units of your choosing.
2. This step applies only to the system ASP.

   Assume that:

- You need 560MB of storage allocated for unprotected storage.
- You have two 2800 Model 001 disk units, each with 2 storage units configured to ASP 1.
- You have four other disk units, each with 2 storage units.

Calculate the unprotected storage per unit value needed as input to the Calculate Checksum Configuration function by:

a. Subtracting the capacity of unit 1 (since it is assigned to unprotected storage primarily for system data) from the amount of unprotected storage that you have determined you need.

   Subtract 320MB from the unprotected storage (560MB) to get the adjusted amount of unprotected storage.

```
(unprotected storage) - (unit 1)          = X

                 560MB  - 320MB           = 240MB
```

b. Divide the adjusted amount of unprotected storage by the number of storage units that have checksum protection. Do not include the load source unit or any 2800-001 units because they do not have checksum protection.

   Divide 240M (560MB needed less the 320MB for the 2800-001 units) by 8.

```
(X)     divided by (checksum units) = (unprotected storage per unit)

240MB   divided by       8          = 30
```

Note: Storage per unit cannot be more than 100. If your calculated value is more than 100, use 100.

3. Select an option on the Calculate Checksum Configuration display:
   - Existing ASP
   - Existing ASP and user-specified units
   - User-specified units
4. Enter the number of the ASP and the amount of unprotected storage per unit. Enter the information in the remaining fields.
5. Press the Enter key.
6. When the Possible Checksum Configuration display is shown, count the total number of units contained in all checksum sets. If this number is different from the number you divided by in your original calculation (8 in the above example), figure your calculation for the unprotected storage per unit value again, using the new number and repeating step 2 on page 18-4 above.

   This must be done to ensure adequate performance of disk accesses for the temporary objects stored in unprotected storage by evenly spreading the space allocations for unprotected storage across all the units in checksum sets. Otherwise, if space for unprotected storage is allocated on just a few storage units, contention is possible and system performance will suffer.

   If the value shown for protected storage available under that checksum configuration is adequate for your needs, the number of disk units you specified in the calculation should be adequate.

   If the value shown for protected storage is less than the value for your protected storage needs, you have to add disk units to the ASP.

Determining the Number of Additional Disk Units You Need: If you determine that you need additional disk units, specify different disk units on the DST Calculate Checksum Configuration display to try different combinations. Use the calculation examples in the previous section to determine the amount of unprotected storage per unit to specify on the display.

For example, assume you have two 9335 disk units and unit 1 configured on your system. You have decided to add an additional 9335 disk unit and want to determine if there will be adequate protected and unprotected storage. Also, assume that you determined you will need 60M of unprotected storage.

To determine the amount of protected and unprotected storage that would be provided by these disk units if checksum protection were in effect:

1. Use the DST Calculate Checksum Configuration display to request a calculation for your base configuration plus the 9335 disk unit to be added. Specify 10 for the unprotected storage per unit value (60M divided by 6 units).
2. You then see the Calculate Storage Configuration display for ASP 1. It shows what the total protected and unprotected storage capacity per storage unit would be if checksum protection were in effect for the specified devices. Checksum set assignments for each unit are also shown.

Based on your earlier calculations ("Determining the Amount of Unprotected Storage You Need for the System ASP" on page 18-4) you can determine whether this amount of storage is adequate for your needs. Refer to the disk capacity information in Table 14-1 on page 14-4 to help determine which devices to add to your system.

Starting Checksum Protection for the System ASP

Read the topic entitled "Calculating a Disk Configuration for Checksum Protection" on page 18-4 to verify storage capacity needs before you perform the following.

Task Overview

You will perform the following tasks during this procedure:

1. Save the system
2. Access DST
3. Add disk units (optional)
4. Start checksum protection
5. Restore the licensed internal code (function code 23)
6. Restore the operating system
7. Restore the objects to the system

Before You Begin

- ☐ To recover the PTF index, you will use function code 23 (Restore) to restore the licensed internal code.
- ☐ Clean the read and write head of the tape unit.
- ☐ Initialize enough tapes for the save operation.

Task 1. Save the Entire System

Save the entire system with the appropriate save commands (for example, SAVSYS, SAVLIB LIB(*NONSYS), SAVDLO) using the following steps. It is recommended that you also save the access paths.

There are two methods you can use to save the entire system:

1. Using option 21 (Entire system) on the Save menu allows you to save the entire system without entering the commands.
2. Using the Save commands allows you to save the entire system by entering the commands from the command line.

Method 1. Using Option 21 (Entire system) on the Save Menu

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:

   ```
   WRKACTJOB
   ```
3. Display the system log QHST to verify it is up to date:

   ```
   DSPLOG  LOG(QHST)
   ```

   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:

   ```
   WRKF FILE(QSYS/QHST*)
   ```

   Look at the list to verify that you saved all copies of the log that will be needed later.
5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.
6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```
7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)
   ```

   Keep this list with your backup log or your save system tapes for future reference.
8. Go to the Save menu:

   ```
   GO SAVE
   ```

   The Save menu is shown.

```
SAVE                                  Save
                                                         System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Changed objects only
     7. Licensed programs
     8. Security data
     9. Storage

    20. All libraries other than system library
    21. Entire system
    22. All IBM libraries other than system library
                                                                    More...
Selection or command
===>


F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
     (C) COPYRIGHT IBM CORP. 1980, 1991.
```

9. Select option 21 (Entire system) from the Save menu and press the Enter key.

```
                              Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .   TAP01          Names
                                   _______
                                   _______
                                   _______

  Prompt for commands  . . . . .   Y              Y=Yes, N=No

  Check for active files . . . .   Y              Y=Yes, N=No
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device when the first tape is full.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting (for unattended save operations) and uses the default values.

Check for active files

Allows you to specify whether or not you want to check for active files. If you specify Y=Yes, the system sends a message when active files are encountered. You can end the checking process or clear the existing files and continue. If N=No is specified, all active files encountered during the save are cleared.

Option 21 will guide you through the following if you selected Y on the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS SBS(*ALL) OPTION(*IMMED)

b. SAVSYS

c. SAVLIB LIB(*NONSYS) ACCPTH(*YES)

d. SAVDLO DLO(*ALL) FLR(*ANY)

e. STRSBS SBSD(controlling-subsystem)

If you want to be notified when the subsystems are ended, change the QSYSOPR message queue by typing the following and pressing the Enter key.

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
```

Messages are sent to the QSYSOPR message queue indicating when the subsystems have ended and the system is in a restricted state.

10. Continue loading tapes when the system sends a message asking you to load the next volume.

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).

2. Determine the next library to be restored beyond the failure by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

```
DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
        OUTPUT(*PRINT)
```

The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.

3. Load a new tape and type the following:

```
SAVLIB LIB(*NONSYS) STRLIB(library-name)
```

Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

Method 2. Using the Save commands

If you do not want to use option 21, you can do the following steps from the command line of a menu:

1. Sign on the system as QSECOFR.

2. Verify that no users are on the system and that no batch jobs are running:

```
WRKACTJOB
```

3. Display the system log QHST to verify it is up to date:

```
DSPLOG  LOG(QHST)
```

Displaying the QHST log automatically brings it up to date.

4. Display all copies of the system log:

   ```
   WRKF FILE(QSYS/QHST*)
   ```

   Look at the list to verify that you saved all copies of the log that will be needed later.

5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.

6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```

7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)
   ```

   Keep this list with your backup log or your save system tapes for future reference.

8. Change the QSYSOPR message queue:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK)  SEV(60)
   ```

9. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Messages are sent to the QSYSOPR message queue indicating that the subsystems have ended and the system is in a restricted state.

10. Load the first tape and make the tape device ready.

11. Save the system:

    ```
    SAVSYS DEV(TAP01) ENDOPT(*LEAVE)
    ```

12. When a message similar to the following appears, load the next tape or make the device ready, and then enter R.

    ```
    Device was not ready or next volume was not loaded (C R)
    ```

13. Save all user and IBM libraries:

    ```
    SAVLIB LIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
          ACCPTH(*YES)
    ```

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be restored beyond the failure by by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
           OUTPUT(*PRINT)
   ```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.
3. Load a new tape and type the following:

   ```
   SAVLIB LIB(*NONSYS) STRLIB(library-name)
   ```

   Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

14. Save the documents, folders, and distribution documents:

    ```
    SAVDLO DLO(*ALL) FLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
    ```
15. Start the subsystems:

    ```
    STRSBS SBSD(controlling-subsystem)
    ```

Task 2. Access DST

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```
3. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
   ```
7. When the system has powered down, the IPL or Install the System display appears.

```
                      IPL or Install the System

Select one of the following:

         1. Perform an IPL
         2. Install the operating system
         3. Use Dedicated Service Tools (DST)
         4. Perform automatic installation of the operating system

Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                    © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                  Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                    Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage




Selection
    4

F3=Exit          F12=Cancel
```

Task 3. Add Units to the System ASP (Optional)

If you have units to add to the system ASP, do the following. Otherwise continue with "Task 4. Start Checksum Protection" on page 18-16.

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                    Work with Disk Units

 Select one of the following:

      1. Work with disk configuration
      2. Analyze disk problem
      3. Work with disk recovery
      4. Work with disk unit information




 Selection
      1

 F3=Exit        F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

```
                        Work with Disk Configuration

 Select one of the following:

      1. Display disk configuration status
      2. Work with ASP threshhold
      3. Work with ASP configuration
      4. Work with checksum protection
      5. Work with mirrored protection










 Selection
      3

 F3=Exit          F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display.

```
                             Work with ASP Configuration

 Select one of the following:

      1. Display disk configuration capacity
      2. Create User ASP
      3. Delete user ASP
      4. Add units to existing ASP
      5. Reset overflowed user ASPs
      6. Change ASP storage threshold
      7. Move units from one ASP to another
      8. Remove unit from configuration







 Selection
      4

 F3=Exit          F12=Cancel
```

4. Select option 4 (Add units to existing ASP) on the Work with Disk Configuration display.

```
                         Select ASP to Add Units to

 Type option, press Enter.
   1=Select

                                      --Protected--  --Unprotected--
 Option  ASP  Threshold  Overflow      Size  %Used      Size  %Used
   _      1      90%      No             0   0.00%       600  77.84%
   _      2      90%      No             0   0.00%       200   0.53%
```

5. Type a 1 in the *Option* column to select the ASP.

   After you select an ASP number, the Select Units to Add to ASP display appears.

```
                         Select Units to Add to ASP

 Type options, press Enter.
   1=Add unit to ASP

         Serial
 Option  Number        Type  Model  Address
   _     10-00A7503    9332   400    0010-0100FFFF
   _     10-00A7503    9332   400    0010-0101FFFF
   _     10-00A3651    9332   400    0010-0400FFFF
   _     10-00A3651    9332   400    0010-0401FFFF

 F3=Exit                       F12=Cancel
```

6. Type a **1** in the *Option* column and to select the disk units to place in the specified ASP and pressing the Enter key.

   The Confirm Add Units display shows what the the entire system configuration will be when you add the units. Use the serial number of the disk unit to verify that you are selecting the correct disk units to add to the ASP.

   Note: It is important that you verify that you want the disk units in the specified ASP. If you add a unit to the wrong ASP and want to move it, the source ASP is cleared after the move operation and will require a restore operation to recover the source ASP.

```
                          Confirm Add Units

Units will be added to ASP number  . . . :    _

Add will take several minutes for each unit.  The system will
have the displayed protection after the unit(s) are added.

Press Enter to confirm your choice for 1=Add units.
Press F12=Cancel to return and change your choice.


           Serial
ASP  Unit  Number      Type  Model  Address        Protection      CSS
 1                                                 Unprotected
        1  10-00A5703  9332   200   0010-0100FFFF  Unprotected
        2  10-00A4293  9335   B01   0110-0100FFFF  Unprotected
 2                                                 Unprotected
        5  10-00A3651  9332   400   0010-0400FFFF  Unprotected
        6  10-00A3651  9332   400   0010-0401FFFF  Unprotected



F12=Cancel
```

7. If you are satisfied with the configuration, press the Enter key to add the disk units to the ASP.

   Adding units will take several minutes. The time it takes depends on the size of each unit being added and the ability of the system to do multiple adds at the same time.

Task 4. Start Checksum Protection

Note: If any user ASP has overflowed, reset the ASP using "Resetting an Overflowed User ASP" on page 15-67.

1. Return to the Work with Disk Unit display by pressing F3 (Exit) two times.

```
                      Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk device problem
     3. Work with disk unit recovery
     4. Work with disk unit information














Selection
     1

F3=Exit          F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

```
                         Work with Disk Configuration

Select one of the following:

     1. Display disk configuration
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection

Selection
     4

F3=Exit          F12=Cancel
```

3. Select option 4 (Work with checksum protection) on the Work with Disk Configuration display.

```
                       Work with Checksum Protection

Select one of the following:

     1. Display checksum configuration
     2. Start checksum protection
     3. Stop checksum protection
     4. Change unprotected storage
     5. Calculate checksum configuration

Selection
     2

F3=Exit          F12=Cancel
```

4. Select option 2 (Start checksum protection) on the Work with Checksum Protection display to start checksum protection for the ASP.

```
                    Select ASP for Start Checksum Protection

Type option, press Enter.
  1=Select

Option  ASP    Protection
  _      1     Unprotected
  _      2     Unprotected




F3=Exit    F11=Display disk configuration status    F12=Cancel
```

5. Select the ASP by typing a 1 in the *Option* and pressing the Enter key.

   The Specify Unprotected Storage display is shown.

```
                         Specify Unprotected Storage

ASP number. . . . . . . . . . . .:

      Serial
Unit  Number      Type  Model   Address
 1    10-00A5307  9332   200   0010-0100FFFF
 2    10-00A4937  9332   200   0010-0101FFFF
 5    10-00A3651  9332   400   0010-0400FFFF
 6    10-00A3651  9332   400   0010-0401FFFF



Type Choice, Press Enter.

   Unprotected storage per unit: . . .  ___  5-100 Megabytes

F3=Exit        F12=Cancel
```

6. Specify the desired value for the unprotected storage per unit and press the Enter key.
7. The Confirm Start of Checksum Protection display is shown.
8. Verify your choice by pressing F10 to confirm your choice of 1=Start checksum protection.

   Note: Starting checksum protection destroys all data in the ASP. You can change your choice by pressing F12 (Cancel) to return and change the configuration.

9. The Display Checksum Configuration display is shown. Press the Enter key to continue.
10. Press F3 (Exit) until you return to the Use Dedicated Service Tools menu.

11. Select option 7 (Start a service tool) on the Use Dedicated Service Tools menu and press the Enter key.
12. Select option 8 (Power off the system) on the Start a Service Tool menu. The Confirm Power Off the System display is shown.
13. Press F10 to confirm your choice to power off the system.

Task 5. Restore the Licensed internal Code

This step is necessary to recover the PTF index during the restore of the OS/400 licensed program.

1. Ensure the key is in the keylock switch on the control panel.
2. Turn the key to the Manual position.
3. Press the Function Select switch to display **02** in the Function display on the control panel.
4. Press the Enter button on the control panel.
5. Select IPL type D (this specifies that the IPL source comes from tape) by pressing the Function Select switch on the control panel until **D** is shown on the Data display.
6. Press the Enter button on the control panel.
7. For the 9406 system unit, ensure that the power switches for the IPL tape unit and all disk units are in the On position.
8. Find the licensed internal code tape, which is the first volume of the most recent set of save tapes.
9. Place the tape in the tape unit used for the IPL. See the setup manual for the device for more information on loading the tape.

   Note: If your tape unit cannot be loaded when the power is off, continue with the next step. You will be prompted later by an SRC code for the tape.
10. Turn on the power to the system by pushing the Power switch up. The switch returns to center after you push it. The 9402 system unit has a green button labeled Power On.
11. Wait as explained below for the tape unit to power on. See the following explanations:

    Notes:

    a. SRC A100 1938 (Tape not found) may be displayed along with the System Attention light until the tape unit is turned on. Continue with the next step.
    b. SRC A100 1933 or 1934 (Tape not ready) may also be displayed along with the System Attention light. When the tape is ready, continue with the next step. If this SRC is displayed for more than 1 or 2 minutes, call your service representative.
    c. SRC 2507 XXXX appears if manual select IPL intervention is required. If intervention is required, press the Function Select switch until 03 appears. Press the Enter button on the control panel.
12. Ensure that the tape is online or ready. No action is required for tape units that perform this step automatically (such as the tape cartridge unit).
13. Ensure that the console display is turned on.

14. Wait for the yellow System Attention light on the control panel to light up.

    There is a delay while the system loads information from the tape. SRCs showing status are continuously updated on the control panel while processing occurs. This can take from 5 to 20 minutes; the time varies depending on the speed of the tape unit and the processor speed for the specific system model.

    When SRC A6xx 6001 is displayed, the system is prepared to start installing or restoring the licensed internal code on the disk unit containing unit 1. Continue with the next step.

15. Select the correct function code by pressing the Function Select switch on the control panel until the correct function code is displayed. (See the "Before You Begin" section in this topic to determine the correct function code.)

    Warning: Option 24 (Install) is used only to recover from the loss of unit 1 in the system ASP, or to recovery from a disaster. Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

    If you selected function code 24, the System Attention light light may appear in one or two minutes and SRC A6xx 6002 is shown. If you are sure you want to install the licensed internal code, select function code 24 again.

    Notes:

    a. When SRC A6xx 6002 is displayed, option 23 can also be selected. If option 23 is selected, a restore of the licensed internal code is performed (not an install). Data on the disk units will not be lost.

    b. If another SRC appears after A6xx 6001 that is not in the A6xx xxxx format, then the system needs additional attention. Call your service representative.

       If the following SRCs appear after SRC A6xx 6001 is displayed, see Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" for an explanation of these SRCs and the steps to follow.

       - A6xx 6002 Disk unit may contain a valid system
       - A6xx 6003 Disk unit not currently a load source
       - A6xx 6004 Disk unit not currently a load source
       - A6xx 6005 Disk unit not found

16. Press the Enter button on the control panel.

    Note: The system will start displaying status SRCs again, which will be continuously updated to show the status of installing the licensed internal code. An example of a status SRC is D6xx 6201 (stand-alone install operations is running).

17. If the yellow System Attention light is on again, and SRC A6xx 6048 (New tape volume needs to be loaded) appears, the system needs the next tape. The xx tells which volume needs to be loaded. Load the correct tape and make the device ready. The install operation will automatically continue.

    If SRC A6xx 6051 appears, the stand-alone function is requesting the model-unique licensed internal code tape found in the service kit. Unload the current tape from the tape device and load the model-unique licensed internal code tape.

- A6xx 6051 Model-unique licensed internal code tape needs to be loaded
- A6xx 6052 Tape loaded was not the model-unique licensed internal code,

If another SRC A6xx xxxx is displayed, look up the displayed SRC in Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" on page A-1 and follow the instructions. For all other SRCs call your service representative.

18. After the install or restore operation of licensed internal code or the model-unique licensed internal code is complete, the system will automatically perform an IPL.
19. Remove the model-unique licensed internal code tape from the tape unit (if it is loaded) when the IPL or Install the System menu is shown.
20. If you have other restore steps to perform and the model-unique licensed internal code tape was loaded, load the previous save tape.

Task 6. Install the Operating System

When the IPL or Install the System display appears, select option 2 (Install the operating system).

During this procedure, the system performs some lengthy disk formatting of the units placed into checksum sets. This process can vary depending on the disk unit type and model. A 9332 disk unit model 200 can take five minutes. A 6100 disk unit can take up to 40 minutes. For more information on disk formatting, see the IPL recovery step 4220 (Checksum configuration change) in Appendix D, "Initial Program Load (IPL) Process."

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                          IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

   Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

Checksum Protection

```
                        Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.







 F12=Cancel
```

3. Press the Enter key.
4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                           Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .  2924







F3=Exit          F12=Cancel
```

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary

language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                    Confirm Language Feature Selection

Language feature . . . . . . . . . . . . . . . . . :   2924

Press Enter to confirm your choice for language feature.
  Installing the system will continue.
Press F12 to return to change your choice for
  language feature.


F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. Status messages are displayed.

   The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                          Running IPL Step

Current IPL step . . . . : Storage Management Recovery
```

After the IPL steps complete, the Install the Operating System menu appears.

Checksum Protection

```
                          Install the Operating System

 Type choices, press Enter.

     Default
        option . . . .  _             1=Take defaults, show no
                                        other installing displays
                                      2=Change installing options

     Date:
        Year . . . . .  _             00-99
        Month  . . . .  _             01-12
        Day  . . . . .  _             01-31

     Time  . . . . . .  _    00-23    HH is hours
                        _    00-59    MM is minutes
                        _    00-59    SS is seconds
```

8. When the Install the Operating System display is shown, use the following information to respond to the prompts.

   Default Option

Value	Description
1	Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation. If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.
2	If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system. If you select 2 for *Default option*, the Installing Options display appears.

   Date

   The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

   Time

   The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

9. Press the Enter key.

   If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages are for your information only. Continue loading tapes in sequence when

messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

If you selected option 2 (Change installing options), the following display is shown.

```
                         Specify Install Options

Type choices, press Enter.

    Restore option . . . . _          1=Restore programs and language
                                        objects from current tape
                                      2=Do not restore programs or
                                        language objects
                                      3=Restore only language objects
                                        from current tape
                                      4=Restore only language objects
                                        from a different tape




    Job and output
      queue options. . . .  1         1=Clear,  2=Keep
```

10. When the Installing Options display appears, use the following information to respond to the prompts.

Restore Option

Value	Description
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

Notes:

a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.

b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.

c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.

d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.

e. The system entry-point table is created again.

f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user

decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	Description
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                              Specify Restore Options

Type choices, press Enter.

    Restore from tape:

       System values . . . . . 2        1=Restore,  2=Do not restore
       Edit descriptions . . . 2        1=Restore,  2=Do not restore
       Message reply list. . . 2        1=Restore,  2=Do not restore
```

11. Using the following information, respond to the prompts on the Restore Options display.

System Values

Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID	Default system code page and character set
QCURSYM	Currency symbol
QDATFMT	Date editing format
QDATSEP	Date separator character
QDECFMT	Decimal data editing format
QKBDTYPE	Default work station keyboard type
QLEAPADJ	Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Value	Description
1=Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

12. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

13. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    `Load the next volume on the tape device (1 2).`

14. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    `Operating system has been installed.  IPL in progress.`

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

15. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).
16. Press the Enter key. Informational messages are displayed.
17. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.
18. When the IPL Options display is shown, respond to the prompts using the following information.

```
                              IPL Options

 Type choices, press the Enter key.

   System date . . . . . . . . . . . . . .   07 / 26 / 88   MM / DD / YY
   System time . . . . . . . . . . . . . .   12 : 00 : 00   HH : MM : SS
   Clear job queues . . . . . . . . . . . .   N              Y=Yes,  N=No
   Clear output queues . . . . . . . . . .   N              Y=Yes,  N=No
   Clear incomplete job logs . . . . . . .   N              Y=Yes,  N=No
   Start print writers . . . . . . . . . .   Y              Y=Yes,  N=No
   Start this device only . . . . . . . . .   N              Y=Yes,  N=No


   Set major system options . . . . . . . .   N              Y=Yes,  N=No
   Define or change system at IPL . . . . .   N              Y=Yes,  N=No







 Last power-down operation was normal
                                         (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure 18-1. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a `Y` to start the console device only, or type an `N` to start all devices.

d. Enter the value for set major system options.

   Type an `N` for each option to leave them unchanged, or type `Y` to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to `N`. If you installed the licensed internal code using option 24 (Install), the default value is set to `Y`.

19. Press the Enter key.

20. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

```
                  Edit Rebuild of Access Paths                RCHAS331
                                                    05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                  ------------Access Paths----------  Unique  Rebuild
Seq   Status      File        Library     Member      Keyed   Time
25__  IPL         FILE234512  LIBRARY111  MBR1234567  No      00:00:56
25__  IPL         FILE234513  LIBRARY111  MBR1234567  No      00:00:56
75__  IPL         FILE234514  LIBRARY111  MBR1234567  Yes     00:00:56
75__  IPL         FILE234515  LIBRARY111  MBR1234567  Yes     00:00:56
88__  IPL         FILE234516  LIBRARY111  MBR1234567  No      00:00:56
99__  AFTIPL      FILE234517  LIBRARY111  MBR1234567  Yes     00:00:56
*OPN  OPEN        FILE126789  L123456789  MBR4567890  Yes     12:34:56
*OPN  OPEN        FILE346789  L123456789  MBR4567890  No      12:34:56
*HLD  HELD        F123336789  L123456789  MBR4567890  No      10:30:06
*HLD  HELD        F123456789  L123456789  MBR4567890  Yes     99:56:01
                                                          More...

F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                     Display Access Path Status

IPL Threshold . . . . . . : 88

         ----------Access Paths----------   Rebuild     Current
Status   File         Library     Member       Build Time  Run Time
RUN      F123456789  L123456789  MBR4567890              00:00:01
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
JRN      F123456789  L123456789  MBR4567890
SYS      F123456789  L123456789  MBR4567890  12:34:56
SYS      F123456789  L123456789  MBR4567890  12:34:56
IPL      F123456789  L123456789  MBR4567890  12:34:56
                                                          More...




F3=Exit and continue IPL   F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN—indicates that the access path is being rebuilt.
- SYS—indicates the access path is a system access path and is waiting to be rebuilt.
- JRN—indicates that the access path is being recovered from its associated journal.
- IPL—indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

21. Press the Enter key to continue.
22. Ensure the keylock switch is in the Normal position.
23. This completes the restore operation for the operating system if you have no other restore steps to perform.
24. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

After the install process is complete, the command entry display or the Main menu appears and you can proceed with the next step.

Task 7. Restore the Objects to the System

1. Change the QSYSOPR message queue:

   ```
   CHGMSGQ QSYSOPR *BREAK SEV(60)
   ```

2. End all subsystems using the following command:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   A message is sent to the QSYSOPR message queue indicating that all subsystem have ended and the system is in a restricted state.

3. Type the following to reclaim storage:

   ```
   RCLSTG
   ```

   If you had journals, journal receivers, or save files (whose library is in the system ASP) stored in user ASPs when you began this procedure, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile.

   If you had libraries and associated objects stored in a user ASP, this command will restore their addressability and transfers ownership of the objects to QDFTOWN user profile.

If objects are damaged in the ASP, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile. This happens whether you have isolated objects or an entire library in an ASP.

4. Restore user profiles from the correct save tape file (label QFILEUPR).

   Note: Use the tapes from the most recent complete save operation (SAVSYS). If a SAVSECDTA command has been run since the last complete save operation, use the SAVSECDTA tapes.

   If the SAVSYS tape is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
   ```

   If the save security data tape (SAVSECDTA) is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

5. Restore the device configuration objects from your most recent SAVSYS tapes. Type the following:

   ```
   RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

6. It is recommended that you save library QRCL and its contents to create a backup copy of the user ASPs containing isolated objects for use in case another failure should occur. This copy may be needed in the following restore steps, depending how you chose to do the restore operation.

   Load a scratch tape, type the following and press the Enter Key.

   ```
   SAVLIB LIB(QRCL) DEV(TAP01) ENDOPT(*UNLOAD)
   ```

7. Restore the objects to the system in **one** of the following ways:

Method 1. Recovery When No Objects Exist in User ASPs or Old Type ASPs Exist

If you had no objects stored in user ASPs or had journals, journal receivers and save files in user ASPs (where the library for the objects are in the system ASP) at the start of this procedure, or if you want to simplify the restore process at the expense of going through restoring objects that are still intact in the user ASPs on your system, perform the following steps:

1. Delete library QRCL by typing the following and pressing the Enter key.

   ```
   DLTLIB LIB(QRCL)
   ```

2. Restore the IBM and user libraries:

   Type the following and press the Enter key,

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01)  ENDOPT(*UNLOAD)
   ```

3. If you have document library objects to restore, load the SAVDLO tape, and then type the following:

   ```
   RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
            SEQNBR(beginning-number ending-number)
   ```

4. Grant all private object authorities that existed when the system was saved by typing the following and pressing the Enter key. No media is required.

   ```
   RSTAUT
   ```

Method 2. Recovery of Objects and Libraries Existing in User ASPs

After the RCLSTG command is run, the addressability of libraries and objects in the user ASP is restored.

1. Restore the individual libraries to the system ASP from your save tapes.

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) OMITLIB(user-ASP-library-name)
   ```

   Do not restore libraries from the SAVLIB tapes that are in user ASPs. Specify the OMITLIB parameter to exclude up to 50 libraries that exist in the user ASPs. Journals should be restored before the journaled files, or journaling is not resumed for the restored files.

2. Display library QRCL to determine if damaged objects were found.

   ```
   DSPLIB LIB(QRCL)
   ```

   If objects are found in QRCL, do the following:

   a. Delete the damaged objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:

      1) Display the library in the user ASP:

         ```
         WRKLIB LIB(library-name)
         ```

      2) Type a **12** (Work with objects) in the *Opt* column and press the Enter key.

      3) Find the objects to be deleted in the *Object* column.

      4) Type a **4** (Delete) in the *Opt* column for each object you want to delete.

      5) Press the Enter key.

   b. After the objects in the ASP are deleted, move the objects in library QRCL back to their original library.

      ```
      MOVOBJ OBJ(QRCL/object-name) OBJTYPE(*XXXX) TOLIB(library-name)
      ```

3. The RCLSTG command changed ownership of objects existing in user ASPs to QDFTOWN user profile. Transfer ownership of the objects in the user ASP library from QDFTOWN user profile to the correct user profile.

   a. Type the following and press the Enter key:

      ```
      WRKOBJOWN USRPRF(QDFTOWN)
      ```

      The Work with Objects by Owner display is shown.

   b. On the Work with Objects by Owner display, type a **9** in the *Opt* column for each object in the ASP library that you want to change ownership for.

   c. If all the objects will have the same owner, type the following on the command line of the Work with Objects by Owner display. Otherwise, continue with step 3d.

      ```
      NEWOWN(owner-name)
      ```

      Note: If you enter NEWOWN(owner-name) on the command line of the Work with Object by Owner display, you will not have to enter an owner name in the *New owner* prompt on the Change Object Owner display for each object.

   d. Press the Enter key.

e. On the Change Object Owner (CHGOBJOWN) display, type the name of the new owner in the *New owner* prompt and press the Enter key. Repeat this step for all the objects that need the ownership changed.

4. Load the the SAVDLO tape, and then type the following to restore document library objects:

```
RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
        SEQNBR(beginning-number ending-number)
```

5. Type the following to grant all private object authorities that existed when the system was saved:

```
RSTAUT
```

This completes the steps to configure checksum protection.

Starting Checksum Protection for a User ASP

Read the topic "Calculating a Disk Configuration for Checksum Protection" on page 18-4 to verify the storage capacity needs before you perform the following.

Task Overview

You will perform the following tasks during this procedure:

1. Save the security data
2. Save all objects in the user ASP
3. Delete the objects in the user ASP
4. Access DST
5. Add units to the ASP (Optional)
6. Start checksum protection
7. Restore objects back to the user ASP

Task 1. Save the Security Data

Use the Save Security Data (SAVSECDTA) command to save the security data. By entering this command now, you can save all private object authorities. If you do not do this now, you must later manually restore private authorities to every object (with the GRTOBJAUT command) that resides on the user ASP containing the units to be reassigned.

To save the security data, do the following:

1. To change the system operator message queue so all messages will appear on the display, type the following and press the Enter key.

   ```
   CHGMSGQ QSYSOPR *BREAK SEV(60)
   ```

2. Load the first tape, and make the tape device ready.
3. Save the security data:

   If you are saving distribution (mail) objects, type the following and press the Enter key.

   ```
   SAVSECDTA DEV(TAP01) MAIL(*YES)
   ```

   If one or more tapes have not been initialized, a message similar to the following may appear:

   ```
   Volume on device TAP01 wrong type (C INZ R)
   ```

   Enter `INZ` and press the Enter key to initialize the tape.

4. When the following message appears, load the next tape, make the device ready, and then enter `R`.

   ```
   Device was not ready or next volume not loaded (C R)
   ```

Task 2. Save all objects on the user ASP

If the user ASP contains data, save the objects in the user ASP by entering the following:

```
SAVLIB LIB(library-name) DEV(TAP01) ENDOPT(*UNLOAD)
```

Task 3. Delete all objects from the user ASP

1. Delete the objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:
   a. Display the library in the user ASP:
      ```
      WRKLIB LIB(library-name)
      ```
   b. Type a **12** (Work with objects) in the *Opt* column and press the Enter key.
   c. Find the objects to be deleted in the *Object* column.
   d. Type a **4** (Delete) in the *Opt* column for each object you want to delete.
   e. Press the Enter key.

Task 4. Access DST

Start the Work with Disk Units function by using the the following steps:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:
   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```
3. End all subsystems:
   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```
   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:
   ```
   PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
   ```
7. When the system has powered down, the IPL or Install the System display appears.

```
                          IPL or Install the System

Select one of the following:

        1. Perform an IPL
        2. Install the operating system
        3. Use Dedicated Service Tools (DST)
        4. Perform automatic installation of the operating system









Selection
   3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                              © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                    Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage





Selection
    4

F3=Exit          F12=Cancel
```

Task 5. Add Units to the ASP (Optional)

If you have units to add to the ASP, continue with the following. Otherwise, continue with task 6.

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                         Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information






Selection
     1

F3=Exit        F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

Checksum Protection

```
                     Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshhold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection









Selection
     3

F3=Exit          F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display.

```
                          Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration








Selection
     4

F3=Exit          F12=Cancel
```

4. Select option 4 (Add units to existing ASP) on the Work with Disk Configuration display.

```
                        Select ASP to Add Units to

 Type option, press Enter.
   1=Select

                                       --Protected--  --Unprotected--
 Option  ASP  Threshold  Overflow       Size  %Used      Size  %Used
   _      1      90%      No              0   0.00%      600   77.84%
   _      2      90%      No              0   0.00%      200    0.53%
```

5. Type a 1 in the *Option* column to select the ASP.

   After you select an ASP number, the Select Units to Add to ASP display appears.

```
                        Select Units to Add to ASP

 Type options, press Enter.
   1=Add unit to ASP

         Serial
 Option  Number       Type  Model  Address
   _     10-00A7503   9332   400    0010-0100FFFF
   _     10-00A7503   9332   400    0010-0101FFFF
   _     10-00A3651   9332   400    0010-0400FFFF
   _     10-00A3651   9332   400    0010-0401FFFF

 F3=Exit                       F12=Cancel
```

6. Type a **1** in the *Option* column and to select the disk units to place in the specified ASP and pressing the Enter key.

   The Confirm Add Units display shows what the the entire system configuration will be when you add the units. Use the serial number of the disk unit to verify that you are selecting the correct disk units to add to the ASP.

   Note: It is important that you verify that you want the disk units in the specified ASP. If you add a unit to the wrong ASP and want to move it, the source ASP is cleared after the move operation and will require a restore operation to recover the source ASP.

Protection

```
                              Confirm Add Units

Units will be added to ASP number  . . . :   _

Add will take several minutes for each unit.  The system will
have the displayed protection after the unit(s) are added.

Press Enter to confirm your choice for 1=Add units.
Press F12=Cancel to return and change your choice.


           Serial
ASP  Unit  Number       Type  Model  Address         Protection      CSS
 1                                                   Unprotected
        1  10-00A5703   9332   200   0010-0100FFFF   Unprotected
        2  10-00A4293   9335   B01   0110-0100FFFF   Unprotected
 2                                                   Unprotected
        5  10-00A3651   9332   400   0010-0400FFFF   Unprotected
        6  10-00A3651   9332   400   0010-0401FFFF   Unprotected



F12=Cancel
```

7. If you are satisfied with the configuration, press the Enter key to add the disk units to the ASP.

   Adding units will take several minutes. The time it takes depends on the size of each unit being added and the ability of the system to do multiple adds at the same time.

Task 6. Start Checksum Protection

1. From the Work with ASP Configuration display, return to the Work with Disk Units display by pressing F12 (Cancel) two times

```
                         Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk device problem
     3. Work with disk unit recovery
     4. Work with disk unit information









Selection
     1

F3=Exit          F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection

Selection
     4

F3=Exit         F12=Cancel
```

3. Select option 4 (Work with checksum protection) on the Work with Disk Configuration display.

```
                    Work with Checksum Protection

Select one of the following:

     1. Display checksum configuration
     2. Start checksum protection
     3. Stop checksum protection
     4. Change unprotected storage
     5. Calculate checksum configuration

Selection
     2

F3=Exit         F12=Cancel
```

4. Select option 2 (Start checksum protection) on the Work with Checksum Protection display.

```
                    Select ASP for Start Checksum Protection

Type option, press Enter.
  1=Select

Option  ASP    Protection
  _      1     Unprotected
  _      2     Unprotected











F3=Exit    F11=Display disk configuration status    F12=Cancel
```

5. Select the ASP by typing a 1 in the *Option* and pressing the Enter key.
6. The Confirm Start of Checksum Protection display is shown. For a user ASP, the confirmation will notify you if any units in the ASP will not be part of a checksum set.

```
                    Confirm Start of Checksum Protection

Warning: Starting checksum protection will destroy all the
data from the ASP.  You may save the data before selecting
this option.  A subsequent stop checksum protection request
on this ASP before an IPL will ignore this start checksum
protection request.

Press F10 to confirm your choice for 1=Start checksum.
Press F12 to return to change your choice.

ASP . . . . . . . . . . :  __









F10=Confirm    F12=Cancel
```

7. Verify your choice by pressing F10 to confirm your choice of 1=Start checksum.

   Note: Starting checksum protection destroys all data in the ASP. You can change your choice by pressing F12 (Cancel) to return and change the configuration.

8. The Display Checksum Configuration display is shown. Press the Enter key to continue.
9. Press F3 (Exit) until you return to the Use Dedicated Service Tools Menu

Task 7. Restore Objects to the User ASP

1. From the IPL or Install the System menu, select option 1 (Perform an IPL)

   After IPL processing is complete, the command entry screen will appear and you can proceed to the next step.

2. Load the correct volume of the SAVSECDTA tapes.

3. Restore the user profiles.

   ```
   RSTUSRPRF USRPRF(*ALL)
   ```

4. Load the correct volume of the RSTLIB or RSTOBJ tapes.

5. Use the appropriate restore command to restore the desired objects to the ASP:

   ```
   RSTLIB SAVLIB(library-name) DEV(TAP01)
   ```

   Or:

   ```
   RSTOBJ OBJ(*ALL) SAVLIB(library-name) OBJTYPE(*ALL)
          ENDOPT(*REWIND) MBROPT(*ALL)
   ```

6. Restore the private authorities:

   ```
   RSTAUT
   ```

   If any objects were restored into a different library than the one from which they were saved, manually grant private authority for the restored objects with the EDTOBJAUT command.

   Notes:

   a. If the journals are in the system ASP, create a new receiver in the user ASP, and run the CHGJRN command to attach the new journal.

   b. Or do one of the following:

      If the user ASP contained journals and you did not delete the old files and then restore them from the media, create a new journal receiver and the journal in the user ASP. Use the Start Journal Access Paths (STRJRNAP) and the Start Journal Physical Files (STRJRNPF) commands to start journaling again.

      1) If the user ASP contained journals and you did not delete the old files and then restore them from the save media, you must delete and restore the files that were journaled which will start journaling again.

7. This completes the steps to start checksum protection for a user ASP.

Adding Storage Units to an ASP while Checksum Is in Effect

You can add units to an ASP while it is under checksum protection to increase the capacity:

1. Use DST Calculate Checksum Configuration display to verify that the new disk will provide the desired capacity.
2. Your service representative attaches the new disk.
3. After the disk is attached, the units for it appear as nonconfigured units on the DST Display Non-configured Units display.
4. Use the DST Add Units to Existing ASP display to select the new units to add to the system ASP.

Protection

The new units are automatically added to the existing checksum sets or a new checksum set is created as part of the add operation.

5. Use the DST Display Checksum Configuration display to verify the new configuration.

Moving a Storage Unit Not in a Checksum Set from the System ASP to a User ASP

Units (other than unit 1) in the system ASP that are not in a checksum set when checksum protection is started do not contain permanent user data. They can be moved without causing destruction of user data in the system ASP.

Warning: When considering the removal of units from the system ASP (ASP 1), remember to leave at least two 2800-001 storage units in the system ASP for the 9406 model D70 and D80 system units that contain 224MB or more of main storage.

To move a unit from the system ASP, perform the following steps:

1. Use the DST Display Checksum Configuration display to determine which units are not in checksum sets. If any units listed under the *Checksum set number* column are marked with an asterisk, they can be moved into a user ASP (with the exception of the unit 1, which must remain in the system ASP).

   Carefully mark down the identification of the units to be moved.
2. Use the DST Move Unit from One ASP to Another display to move the unit from the system ASP to the desired user ASP.
3. Use the DST Display Disk Configuration displays and the Display Checksum Configuration displays to verify the configuration changes.

Replacing One Disk Unit in a Checksum Set with Another Unit

You can take advantage of checksum protection's ability to rebuild the data on a unit in a checksum set when replacing one unit in a checksum set with another unit. For example, when a disk is experiencing intermittent errors that seem to indicate a hard failure might occur, you may want to replace the unit. Your service representative will assist you in analyzing the problem and deciding if the disk should be replaced. For more information about replacing the unit, see "Replacing a Failed Disk Unit in the System ASP" on page 18-50.

Increasing the Amount of Unprotected Storage in the System ASP

You can increase the amount of unprotected storage in the system ASP, but to do this you must go through essentially the same configuration procedure as you did for your first checksum protection configuration. There is a difference in that you use DST to increase the amount of unprotected storage instead of initially configuring checksum protection. For more information, see "Starting Checksum Protection for the System ASP" on page 18-6.

Handling Unprotected Storage Overflows

If you underestimate the amount of unprotected storage needed, temporary objects automatically overflow into the protected storage area. A message automatically notifies you when this occurs. Overflow slows system performance because checksum protection is performed on these temporary objects, which ordinarily are stored in the unprotected storage area.

When you IPL the system, unprotected objects are always deleted and this overflow condition is reset. You may want to change your checksum configuration and allocate a larger amount of unprotected storage space if this occurs frequently. This type of reconfiguration requires a save and restore of your system ASP because it destroys all data in the system ASP.

You may also consider installing additional devices to give you more unprotected storage without clearing the system ASP. For each unit added, you get the amount of unprotected storage currently specified for the unprotected storage per unit value you specified for the checksum configuration.

1. Select option 4 (Change unprotected storage) on the Work with Checksum Protection display. Use this option to change the amount of unprotected storage for the system after it has checksum protection. This display automatically shows the configuration of ASP 1 if it has checksum protection.

   Note: Increasing the amount of unprotected storage causes the ASP to be cleared.

```
                           Change Unprotected Storage

ASP number . . . . . . . . . . . . . :   1
Current unprotected storage per unit :  22    Megabytes

       Checksum  Serial
Unit   Set No.   Number       Type  Model    Address
 1        *      10-0001234   9332   200
 2        1      10-00031     9332   200
 3        1      10-00031     9332   200
 6        2      57-000451    9335   B01
 8        2      57-000451    9335   B01

* - Indicates that unit is not part of any checksum set

Type choice, press Enter.

    New unprotected storage per unit. :  ___    5-100  Megabytes



F3=Exit          F12=Cancel
```

   - *Current unprotected storage per unit.* The amount of unprotected storage that exists now.
   - *New unprotected storage per unit.* The amount of unprotected storage you want to enter.

Handling Protected Storage That Has Reached Maximum Storage Capacity

If objects placed in the ASP cause the protected storage area to exceed the threshold value specified for the ASP, a message is sent to the system operator every hour warning that the threshold was reached.

When the actual storage space is exceeded in a user ASP, the data overflows into the system ASP.

When the actual storage space is exceeded in the system ASP, and the protected area reaches its maximum capacity, the system completely stops, resulting in a device error. When this occurs, you must do one of the following after you IPL your system:

- Delete objects from auxiliary storage.
- Reduce the amount of unprotected storage.
- Move user ASP devices to the system ASP.
- Stop checksum protection.
- Configure additional devices to increase the amount of protected storage.

It is possible for protected storage to reach its maximum capacity because temporary (unprotected) objects overflow into protected storage. If this occurs, you can IPL your system (which deletes all temporary objects), freeing protected storage. For information on what to do if this occurs, see "Handling Unprotected Storage Overflows" on page 18-47.

If checksum protection is in effect, the system requires at least 2MB of available protected storage to IPL the system. If protected storage reaches its maximum capacity, and you have less than 2MB of protected storage, you must use DST options to increase the protected storage capacity. You should either add units to the system ASP, which can be allocated to checksum sets, or reduce the amount of unprotected storage per unit. While the system is designed to allow for recovery when protected storage becomes full, there may be instances when you are unable to IPL your system.

Stopping Checksum Protection

You do not need to save or restore the system to stop checksum protection. The units you specified for checksum protection are available for normal use and the storage space for checksum data is freed.

To stop checksum protection, perform these steps:

1. Select option 3 (Stop checksum protection) on the Work with Checksum Protection display.
2. Select the ASP to stop checksum protection by typing a 1 (Select) in the Option column.
3. Press F10 to confirm your choice of 1 = Stop Checksum protection.
4. Use the DST Display ASP Configuration display to confirm your changes.

Checksum Recovery Actions

Recovering from disk unit media failures involves steps the service representative performs as well as steps you perform. When using checksum recovery consider the following.

Checksum Recovery Actions Performed by the Service Representative

The recovery actions performed by the service representative are provided here so you know what happens before you recover from a disk unit media failure. This discussion applies to a disk failure in an ASP where checksum protection is in effect.

If a disk unit in your system fails, your service representative determines whether data loss has occurred. For the purpose of the checksum recovery actions discussed below, any case of data loss is considered a disk unit media failure.

Following are the steps your service representative follows when assisting you in recovery from a disk unit media failure:

1. The service representative attempts to use the DST Save Disk Unit Data function to save the data on the failed unit(s) to tape.
   - If all data from the failed unit is saved, it can be restored to the replacement unit. Further recovery steps are not needed beyond those required for other cases (such as power failures or other disk failures) where machine processing ends abnormally without disk media damage.
   - If some of the data is saved and the unit that failed is not unit 1, the chances of partial data loss are reduced and overall recovery time is reduced by restoring data to the replacement unit because checksum protection will not have to rebuild any data that has been successfully copied.
   - If all data is not saved and the unit that failed is unit 1, your service representative will go through a procedure to rebuild the data stored on unit 1 to its replacement unit rather than restoring the saved data.
2. The service representative attaches the replacement disk unit to the system. If the disk that failed contained unit 1, it will be attached so that it is addressed correctly to indicate it contains unit 1. Otherwise, no specific address is required for the new disk.
3. The following occurs if the disk that failed contained unit 1:
   a. The service representative performs an IPL of the system from your last SAVSYS or SAVSTG tapes and uses the stand-alone licensed internal code install (option 24) utility to format the new unit 1 and copy the system licensed internal code from tape to the new unit 1.

      Other units in the system will not be cleared by this step. However, they will be considered nonconfigured at this point. Installing microcode results in a default disk configuration of just unit 1 being configured into the system.
   b. The service representative uses the DST Recover Configuration function to request the system to recover the disk configuration that was in effect prior to the failure by reading information stored on the other disks in the

system. This disk configuration information is placed on unit 1 as a result.

This step is necessary for the system to reestablish which disks were in the system ASP and which were in the user ASPs, and whether or not checksum or mirrored protection was in effect for the ASP.

Certain values that are normally stored on unit 1 cannot be rebuilt and will be lost. Most of these will be put back into effect automatically by IPL processing performed by the OS/400 program. You may have to set other values correctly during the recovery actions you perform.

4. The following occurs if the disk unit that failed contained a unit other than unit 1 and some of the data for the unit was saved to tape:
   a. The service representative uses the DST Restore Disk Unit Data function to restore the saved data onto the replacement unit.
5. The following occurs if the disk unit that failed contained a unit other than unit 1 and none of the data for the unit was saved to tape:
   a. The service representative uses the DST Replace Disk function to indicate to the system that the old (failed) unit is being replaced by the newly attached unit. The newly attached unit is formatted at this point.

When you continue the recovery actions by selecting to IPL the operating system, additional disk recovery processing may be performed, if necessary, to go through checksum recovery processing to rebuild the data previously contained on the failed units.

Replacing a Failed Disk Unit in the System ASP

After the service representative has replaced the disk unit, follow these steps to recover from replacement of failed disk units in the ASP when checksum protection is in effect:

If the Units That Failed Did Not Include Unit 1:

1. Select option 1 (Perform an IPL) on the Use Dedicated Service Tools menu.
2. Select option 1 (Perform an IPL) on the IPL or Install the System menu

   The system will attempt to rebuild the contents of the disk units that were replaced and are members of a checksum set. The IPL may take several hours, but you will not need to reload your system, data, or other objects residing in the ASP. Unit 1 is never a member of a checksum set and is not operated on by this rebuild processing.

If the Units That Failed Did Include Unit 1:

1. If the units that failed included unit 1 and you had previously changed the DST passwords from their defaults to values of your own choosing, use the DST security options to set your password values back into effect. For more information about DST passwords, see *Security Concepts and Planning*.

   Also, if you had previously changed the system value for the date format from the default, the default format will be in effect until the OS/400 program completes IPL processing which sets the format back to the value that was in effect when you saved your system. It is assumed the LIC has been restored.

   a. Perform an abbreviated install of the operating system to reinstall certain data needed by IPL onto unit 1. For more information on install, refer to

"Procedure for Restoring the Operating System" on page 10-18. To perform the abbreviated install, follow these steps:

a. Select the option to install the operating system.

b. On the Install Operating System display, select the option named *Change Installing Options*.

c. On the Installing Options display, select the option named *Do not Restore Programs or Language Objects*.

On this display you normally want to specify *NO* for the option named *Clear Job and Output Queues*.

The system will then perform a relatively short amount of processing to load the necessary data from tape onto the system. After this processing is complete, the system will automatically continue to the step of doing an IPL of the operating system.

The system will attempt to rebuild the contents of the disk units that were replaced and are members of a checksum set. The IPL may take several hours, but you will not need to reload your system, data, or other objects residing in the system ASP. Unit 1 is never a member of a checksum set and is not operated on by this rebuild processing.

2. If the units that failed included unit 1, a message may appear during IPL processing, stating that some hardware configuration information has been lost and prompting you to enter that information again. To do this, use the Work with Hardware Products (WRKHDWPRD) command.

3. If the units that failed included unit 1, a message (CPI0916) may appear during IPL processing, stating that network attribute information has been lost and prompting you to set the correct values for these attributes on your system. Follow the instructions provided in that message to set the correct network attributes at this point.

4. If the units that failed included unit 1 and if you use the AS/400 Cryptographic Support (5728-CR1) licensed program on your system, you must reenter the master cryptographic key. To do this you use the Set Master Key (SETMSTK) command.

Recovering From a Disk Unit Media Failure in a User ASP That Has Checksum Protection

After the service representative has replaced the disk unit, perform the following steps to recover from a replacement of a failed disk unit in the user ASP when checksum is in effect.

1. Select option 1 (Perform an IPL) on the Use Dedicated Service Tools (DST) menu.

2. Select option 1 (Perform an IPL) on the IPL or Install the System menu.

   The system will attempt to rebuild the contents of the disk unit that was replaced and is a member of a checksum set. The IPL may take several hours, but you will not need to restore the system, data, or other objects residing in the user ASP.

Estimating Checksum Recovery Time

The number of storage units in the checksum set in which the disk unit failure occurs has a direct bearing on how long it takes to reconstruct the data on the lost unit. To estimate the length of time to reconstruct data:

1. Determine the number of units in the checksum set that contained the unit that failed.
2. Multiply this number by the per-unit rebuild time: 5 to 10 minutes for 9332 units, 10 to 15 minutes for 9335 and 9336 010 units, 12 to 20 minutes for 6100 units, 20 to 30 minutes for 9336 020 units.

   For example, rebuilding a 9335 unit in a 3-unit checksum set should take from 30 to 45 minutes (3 units multiplied by 10 to 15 minutes for each unit).

If a 9332 model 400 or 600, or a 9335 model B01 disk unit fails and is replaced, it usually means that a unit in two different checksum sets must be rebuilt (because these disk units have two storage units). To estimate the time to reconstruct the data on an entire drive (with two storage units), perform the calculation described above and then double the time.

The time estimates given in these examples are just approximate. The actual times experienced on your system could vary significantly, depending on the particular characteristics of your disk configuration.

These time estimates do not include the time for other recovery processing, such as rebuilding access paths, which may be needed when the system ends abnormally.

Part 6. Mirrored Protection

Chapter 19. Introduction to Mirrored Protection

The information in Part 6 is intended to be used by you, your IBM marketing representative or remarketing representative, and your service representative. The procedures in this part require the support of your IBM marketing representative and your service representative.

This chapter provides the following:

- Overview of mirrored protection
- Terminology used with mirrored protection.
- An overview of how mirrored protection works, including information about levels of protection, concurrent maintenance, system performance, and protection limitations.
- A review of how the system identifies disk units.

Overview of Mirrored Protection

Mirrored protection is a function that increases the availability of the AS/400 system in the event of a failure of a disk-related hardware component. It can be used on most models of the AS/400 system and is a part of the licensed internal code. Different levels of mirrored protection are possible, depending on what hardware is duplicated. The system remains available during a failure of a disk-related hardware component, such as a disk unit, a disk controller, a disk I/O processor, or a bus, if the failing hardware component and hardware components attached to it are duplicated. For the 9406 system unit, some failed hardware components can be serviced while the system remains available.

Mirrored protection provides these benefits:

- It keeps the system available when most disk-related hardware failures occur. For a disk unit failure, you do not need to restore your data and wait for a long recovery.
- It reduces the possibility of data loss by keeping the two copies of each unit's data on two different storage units. Changes to the data are written to both storage units of the mirrored pair. Data is read from either storage unit.
- For the 9406, it allows a degree of concurrent maintenance. Disk-related hardware failures can often be repaired while using the system.
- System performance is comparable to a non-mirrored system. System performance can improve, depending on the number of disk-related hardware components that are duplicated and depending on the application environment.
- Starting mirrored protection is fast and easy. You do not need to restore your data after starting mirrored protection.

Mirrored protection should not be used as a replacement for system backup procedures. You should continue to save the system on a regular basis because there are times when mirrored protection cannot be used to recover data. Mirrored protection reduces the times you have to restore from your backup copy. For more information, see "Mirrored Protection Limitations" on page 19-5.

Mirrored Protection Terminology

The following is a list of terms used in referring to mirrored protection.

ASP (auxiliary storage pool): A group of units defined from the disk storage devices. ASPs provide a means of isolating objects on specific units to prevent loss of data because of disk failures of other disk units in different ASPs. Performance may be improved by isolating heavily referenced objects in a user ASP. Every system has a system ASP which is ASP 1. You can create ASPs 2 through 16.

Bus: One or more conductors used for transmitting signals or power. It provides all signal, power, and ground connections to the internal adapters.

Concurrent maintenance: The process of repairing or replacing a failed disk-related hardware component while using the system. Concurrent maintenance is only possible on the 9406 with mirrored protection.

Controller: A device that coordinates and controls the operations of one or more I/O devices and synchronizes the operations of such devices with the operation of the system as a whole.

Deferred maintenance: The process of waiting to repair or replace a failed disk-related hardware component until the system can be powered down. The system is available, although mirrored protection is reduced by whatever hardware components have failed. Deferred maintenance is only possible with mirrored protection.

Disk Unit: The physical enclosure containing one or more storage units.

Examples:

- A 6105 disk unit has a single storage unit.
- A 9335-B01 disk unit has 2 storage units in its enclosure.

I/O processor: The I/O processor is attached to the bus and controls information between the bus and specific groups of I/O controllers and disk units. The I/O processor is also called a storage controller or a magnetic storage device controller.

Mirrored protection: A function that protects data by duplicating all disk data on one storage unit in an auxiliary storage pool on another storage unit in the same auxiliary storage pool. If a disk failure occurs, the system continues to run using the remaining storage unit of the mirrored pair.

Mirrored pair: Two storage units that contain the same data and are referred to by the system as one unit.

Mirrored unit: The storage unit that is half of a mirrored pair.

Storage Unit: The defined space within a disk unit that is addressed by the system.

Unit: The defined division of single-level storage. This space is the smallest disk location addressable by the user. An ASP is one or more units which are identified by unique unit numbers. A unit in a non-mirrored ASP is one storage unit. A unit in a mirrored ASP is a mirrored pair which is two storage units.

Certain create commands (CRTPF, CRTJRNRCV, etc) can create an object on a specified unit. In the non-mirrored environment this is a single storage unit. In the mirrored environment, the UNIT parameter value means a mirrored *pair*.

How Mirrored Protection Works

Because mirrored protection is configured by auxiliary storage pool (ASP), you can mirror one, some, or all ASPs on the system. By default, every system has a system ASP. It is not necessary to create user ASPs in order to use mirrored protection. Although mirrored protection is configured by ASP, all ASPs must be mirrored to provide for maximum system availability. If a disk unit fails in an ASP that is not mirrored, the system cannot be used until the disk unit is repaired or replaced.

The start mirrored pairing algorithm automatically selects a mirrored configuration that provides the maximum protection at the bus, I/O (input/output) processor, or controller level for the hardware configuration of the system. When storage units of a mirrored pair are on separate buses, they have maximum independence or protection. Because they do not share any resource at the bus, I/O processor, or controller levels, a failure in one of these hardware components allows the other mirrored unit to continue operating. The pairing algorithm does not allow mirroring between storage units in the same disk unit when replacement of a common part could cause data loss for both storage units. For example, the two storage units in a 9332 or a 9335 disk unit are not paired to make a mirrored pair.

Any data written to a unit that is mirrored is written to both storage units of the mirrored pair. When data is read from a unit that is mirrored, the read operation can be from either storage unit of the mirrored pair. It is transparent to the user which mirrored unit the data is being read from. A user is not aware of the existence of two physical copies of the data.

If one storage unit of a mirrored pair fails, the system *suspends* mirrored protection to the failed mirrored unit. The system continues to operate using the remaining mirrored unit. The failing mirrored unit can be physically repaired or replaced.

After the failed mirrored unit is repaired or replaced, the system *synchronizes* the mirrored pair by copying current data from the storage unit that has remained operational to the other storage unit. During synchronization, the mirrored unit to which the information is being copied is in the *resuming* state. If repairing the failed storage unit requires replacement of the other storage unit in the disk unit, (as in a 9335 Model B01), synchronization occurs to both storage units in the repaired disk unit. Synchronization does not require a dedicated system and runs concurrently with other jobs on the system. System performance is affected during synchronization. When synchronization is complete, the mirrored unit becomes *active*.

Level of Protection

The level of mirrored protection determines if the system keeps running when different levels of hardware fail. Mirrored protection always provides disk unit-level protection which keeps the system available for a single disk unit failure. To keep the system available for failures of other disk-related hardware requires higher levels of protection. For example, to keep the system available when an I/O processor fails, all of the disk units attached to the failing I/O processor must have I/O processor level protection. That is, all of the disk units attached to the failing I/O processor must have active mirrored units attached to a different I/O processor.

For the 9406, the level of mirrored protection also determines if concurrent maintenance can be done for different types of failures. Certain types of failures require concurrent maintenance to diagnose hardware levels above the failing hardware component. For example, to diagnose a power failure in a disk unit requires resetting the I/O processor to which the failed disk unit is attached. Therefore, I/O processor-level protection is required. The higher the level of mirrored protection, the more often concurrent maintenance is possible.

The level of protection you get depends upon the hardware you duplicate. If you duplicate disk units, you will have disk unit-level protection. If you duplicate disk unit controllers as well, you have controller-level protection. If you duplicate I/O processors, you have I/O processor-level protection. If you duplicate busses, you have bus-level protection.

Figure 19-1 on page 19-4 shows four hardware configurations of mirrored protection. In each configuration, the two storage units shown make a mirrored pair.

1. The first (upper left) configuration shows disk unit-level protection. With disk unit-level protection, the system continues to operate after a disk unit failure. If the controller or I/O processor fails, the system cannot access data on either of the storage units of the mirrored pair, so the system is unusable.
2. The next (upper right) configuration shows disk controller-level protection. With controller-level protection, the system can continue to operate after a disk controller failure. If the I/O processor fails, the system cannot access data on either of the disk units, and the system is unusable.
3. The next (lower left) configuration shows I/O processor-level protection. With I/O processor-level protection, the system can continue to operate after an I/O processor failure. Only if the bus fails will the system become unusable.
4. The last (lower right) configuration shows bus-level protection. With bus-level protection, the system can continue to operate after a bus failure. However, the system cannot continue to operate if bus 0 fails.

During the start mirrored protection operation and during the add operation to a mirrored ASP, the system pairs the disk units to provide the maximum level of protection for the current hardware configuration. The hardware configuration is both the hardware and how the hardware is connected.

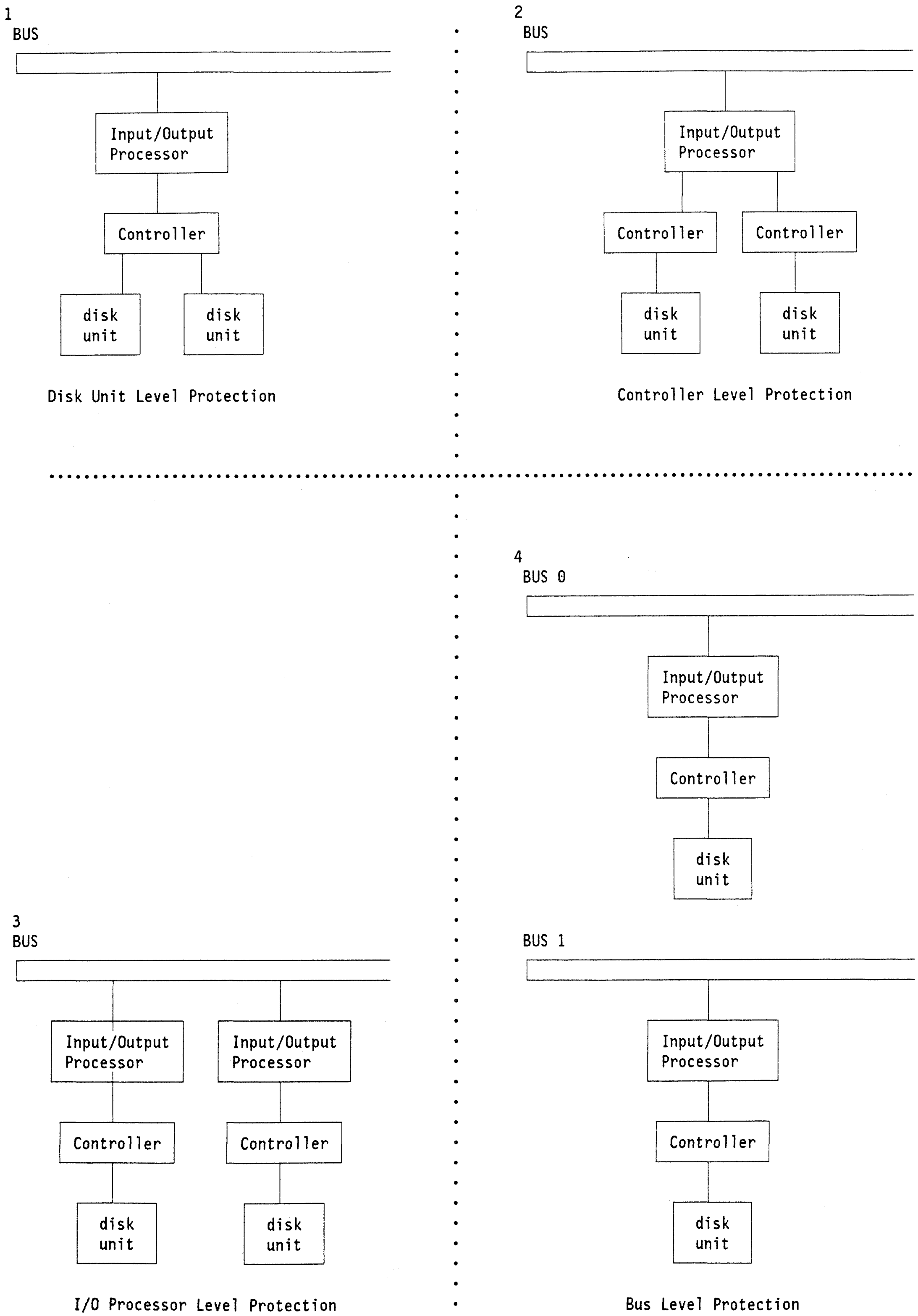


Figure 19-1. Mirrored Protection Configurations

Mirrored Protection Limitations

Although mirrored protection can keep the system available after disk-related hardware failures occur, it is not a replacement for save procedures. There can be multiple types of disk-related hardware failures, or disasters (such as flood or sabotage) that require backup media.

Mirrored protection cannot keep your system available if the remaining storage unit in the mirrored pair fails before the first failing storage unit is repaired and mirrored protection is resumed. If two failed storage units are in different mirrored pairs, the system is still available and normal mirrored protection recovery is done because the mirrored pairs are not dependent on each other for recovery. If a second storage unit of the same mirrored pair fails, the failure may not result in a data loss. If the failure is limited to the disk electronics, or if the service representative can successfully use the Save Disk Unit Data function to recover all of the data, no data is lost.

If both storage units in a mirrored pair fail causing data loss, the entire ASP is lost and all units in the ASP are cleared. You must be prepared to restore your ASP from the backup media and apply any journal changes. (See "Recovering from Disk Unit Media Failures" on page 16-3 if the system ASP is lost, or see "Method 3. Recovering from a Disk Media Failure in a User ASP" on page 16-43 if a user ASP is lost.)

When starting the mirrored protection operation, objects created on a preferred unit may be moved to another unit. The preferred unit may no longer exist after mirror protection is started.

System Performance When Mirrored Protection is in Effect

Most systems should show little difference in performance. In some cases, mirrored protection can improve performance. Mirrored protection normally requires additional disk units and input/output processors. Since mirrored protection uses some main storage, some systems may require additional main storage to maintain the same level of performance.

Generally, functions that do mostly read operations see equal or better performance with mirrored protection. This is because read operations have a choice of two storage units to read from, and the one with the faster expected response time is selected. Operations that do mostly write operations (such as updating database records) may see slightly reduced performance on a system that has mirrored protection because all changes must be written to both storage units of the mirrored pair. Thus, restore operations are slower.

For details on system performance with mirrored protection, see Chapter 23, "Performance Considerations for Mirrored Protection" on page 23-1 and "Capacity Planning Tools" on page 24-2.

Additional IPL Time after an Abnormal System End: In some cases, if the system ends abnormally, the system cannot determine whether the last updates were written to both storage units of each mirrored pair. If the system is not sure that the last changes were written to both storage units of the mirrored pair, the system synchronizes the mirrored pair by copying the data in question from one storage unit of each mirrored pair to the other storage unit. The synchronization occurs during the IPL following the abnormal system end. If the system can save a copy of main storage before it ends, the synchronization process takes just a few minutes. If not, the synchronization process can take much longer. The extreme case could be close to a complete synchronization.

If you have frequent power outages, you may want to consider adding an uninterruptible power supply to your system. Should main power be lost, the uninterruptible power supply allows the system to continue. A basic uninterruptible power supply allows the system time to save a copy of main storage before ending which avoids long recovery. Both storage units of the load source mirrored pair must be powered by the basic uninterruptible power supply.

Spare Disk Units

The amount of time that the system runs with reduced mirrored protection after a disk-related hardware failure can be minimized by keeping spare storage units. If a disk unit fails, a spare may be used to replace the failed storage unit while the system is running. The new mirrored pair is then synchronized and mirrored protection is available while the failed storage unit is being serviced.

Overview of Concurrent Maintenance

On systems without mirrored protection, the system is not available when a disk-related hardware failure occurs and remains unavailable until the failed hardware is repaired or replaced. However, with mirrored protection the failing hardware can often be repaired or replaced while the system is being used.

Concurrent maintenance is supported only for a 9406 system unit that has mirrored protection. The best hardware configuration for mirrored protection also provides for the maximum amount of concurrent maintenance.

It is possible for the system to operate successfully through many failures and repair actions. For example, a failure of a disk head assembly will not prevent the system from operating. A replacement of the head assembly and synchronization of the mirrored unit can occur while the system continues to run. The greater your level of protection, the more often concurrent maintenance can be performed.

On some models, the system restricts the level of protection for unit 1 and its mirrored unit to only controller-level protection. See rule 6 on page 21-1 for restrictions.

Under some conditions, diagnosis and repair can require active mirrored units to be suspended. You may prefer to power down the system to minimize the exposure of operating with less mirrored protection. Some repair actions require that the system be powered down. Both of these are *deferred maintenance* situations.

Review of How the System Addresses Storage

Disk units are assigned to an auxiliary storage pool (ASP) on a storage unit basis. The system treats each storage unit within a disk unit as a separate unit of auxiliary storage. When a new disk unit is attached to the system, the system initially treats each storage unit within it as nonconfigured. Through dedicated service tools (DST) options you can add these nonconfigured storage units to either the system ASP or a user ASP of your choosing. When adding nonconfigured storage units, use the serial number information assigned by the manufacturer to ensure you are selecting the correct physical storage unit. Additionally, the individual storage units within the disk unit can be identified through the *Address* field on the DST Display Disk Configuration display. For more information about displaying disk configuration, see Chapter 15, "Working with Auxiliary Storage Pools"

When you add a nonconfigured storage unit to an ASP, the system assigns a unit number to the storage unit. The unit number can be used instead of the serial number and address. The same unit number is used for a specific storage unit even if you connect the disk unit to the system in a different way.

When a unit has mirrored protection, the two storage units of the mirrored pair are assigned the same unit number. The serial number and the address distinguish between the two storage units in a mirrored pair.

To determine which physical disk unit is being identified with each unit number, make note of the unit number assignment to ensure correct identification. If a printer is available, print the DST or SST display of your disk configuration. If you need to verify the unit number assignment, use the DST or SST Display Configuration Status display to show the serial numbers and addresses of each unit.

The address has twelve digits and is identified as follows:

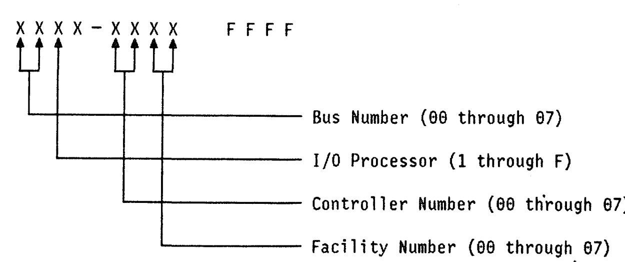


Figure 19-2. Resource Address

The storage unit addressed by the system as unit 1 is always used by the system to store licensed internal code and data areas. The amount of storage used on unit 1 is quite large and varies depending on the configuration of your system. Unit 1 contains a limited amount of user data. Because unit 1 contains the initial programs and data used during an IPL of the system, it is also known as the *load source unit*.

The system reserves a fixed amount of storage on units other than unit 1. The size of this reserved area is 1.08MB per unit, reducing the space available on each unit by that amount.

Chapter 20. Planning for Mirrored Protection

When considering mirrored protection, contact your IBM marketing representative to guide you through these steps.

You will need to do the following steps before you can start mirrored protection:

1. Decide whether to use mirrored protection.
2. Decide which ASP or ASPs to protect with mirrored protection.
3. Determine disk storage capacity requirements.
4. Determine the level of protection you want for each mirrored ASP.
5. Determine what extra hardware you need for mirrored protection.
6. Determine what extra hardware you need for performance, if any.
7. Order your hardware.
8. Plan the installation of your system and the configuration of new units.
9. Install the new hardware.

Step 1: Deciding Whether to Use Mirrored Protection

If you have a multi-bus system or a large single-bus system, you should consider using mirrored protection. The greater the number of disk units that are attached to a system, the more frequent disk-related hardware failures are, simply because there are more individual pieces of hardware that can fail. Therefore, the possibility of data loss or loss of availability as a result of a disk or other hardware failure becomes more likely. Also, as the amount of disk storage on a system increases, the recovery time after a disk storage subsystem hardware failure increases significantly. Downtime becomes more frequent, more lengthy, and more costly.

Benefits of Mirrored Protection

With the best possible mirrored protection configuration, the system continues to run after a single disk-related hardware failure. On a 9406 system unit, the failed hardware can sometimes be repaired or replaced without having to power down the system. If the failing component is one that cannot be repaired while the system is running, such as a bus or an I/O processor, the system usually continues to run after the failure. Maintenance can be deferred, the system can be shut down normally, and a long recovery time can be avoided.

Even if your system is not a large one, mirrored protection can provide you valuable protection. A disk or disk-related hardware failure on an unprotected system leaves your system unusable for several hours. The actual time depends on the kind of failure, the amount of disk storage, your backup strategy, the speed of your tape unit, and the type and amount of processing the system performs. If you or your business cannot tolerate this loss of availability, you should consider mirrored protection for your system, regardless of your system's size.

Costs of Mirrored Protection

The main cost of using mirrored protection is in additional hardware. To achieve high availability and prevent data loss when a disk unit fails, you need mirrored protection for all the ASPs. This normally requires twice as many disk units. If you want continuous operation and prevention of data loss when a disk unit, controller, or I/O processor fails, you need duplicate disk controllers and I/O processors. A model upgrade can be done to get nearly continuous operation and to prevent data loss when any of these failures occur, as well as the failure of bus 1. If bus 0 fails, the system cannot continue to operate. Because bus failures are rare, and bus-level protection is not significantly greater than I/O processor-level protection, you may not find a model upgrade to be cost-effective for your protection needs.

Mirrored protection has a minimal effect on performance. If the busses, I/O processors, and controllers are no more heavily loaded on a system with mirrored protection than they are on an equivalent system without mirrored protection, then the performance of the two systems should be approximately the same.

In deciding whether or not to use mirrored protection on your system, you must evaluate the cost of potential downtime against the cost of additional hardware, over the life of the system. The additional cost in performance or system complexity is usually negligible. You should also consider other availability and recovery alternatives, such as checksum protection.

Comparison of Availability and Recovery Options

A simple comparison of selected availability and recovery options for journaling, checksum protection, and mirrored protection follows. The comparison discusses the three options separately, even though journaling can also be used in combination with checksum protection or mirrored protection. Mirrored protection and checksum protection cannot be used together on an ASP.

For journaling, the comparison assumes your system has an unprotected system ASP plus a user ASP for the journal. For checksum protection, your system assumes a checksum-protected system ASP and no user ASPs. For mirrored protection, it assumes for your system that all ASPs have mirrored protection.

- Data loss after a single disk unit failure
  - Journaling — minimal loss to files being journaled if good backups are available.
  - Checksum protection — none.
  - Mirrored protection — none.
- Recovery time after a single disk unit failure
  - Journaling — potentially many hours. After the failed disk unit is repaired, the system must be reloaded, long system recoveries must be run, data must be restored, and journal updates must be applied.
  - Checksum protection — a few hours. After the failed disk unit is repaired, the system can rebuild the lost data.
  - Mirrored protection — none. The system continues to run while the failed disk unit is repaired on-line.
- Performance impacts
  - Journaling — varies from minimal to significant.
  - Checksum protection — varies from minimal to significant.
  - Mirrored protection — negligible. In some environments there is a performance improvement. After repair actions, there is a temporary decrease in performance.
- Planning complexity
  - Journaling — minimal. Additional disk units are needed for the user ASP.
  - Checksum protection — careful planning is necessary to determine the size of the unprotected area. Additional disk units may be needed.
  - Mirrored protection — Careful planning is necessary to determine hardware and configuration needed to provide the best possible protection.
- Setup complexity and time required
  - Journaling — minimal.
  - Checksum protection — considerable. New disk units and other hardware must be installed and the disk units must be added to the system ASP. The protected and unprotected storage must be defined for all disk units in the ASP. The system must be completely saved and restored. This requires considerable time and operator intervention. The system is in a restricted state and is unusable until the process is completed.
  - Mirrored protection — minimal. New disk units and other hardware must be installed and the disk units that will have mirrored protection must be added to the ASP. Saving and restoring of the system is not required. Starting mirrored protection is a simple operation that does not require user intervention after it is started. The process takes a maximum of a few hours on the largest systems.
- Operational and management complexity
  - Journaling — continuous management of the journal receivers is required.
  - Checksum protection — negligible unless the checksum configuration or the protected-unprotected storage boundary must be changed. (The boundary between protected storage and unprotected storage is changed; the storage is

not changed.) It can be difficult and time consuming if a change is required.

- Mirrored protection — negligible. Mirrored protection is generally transparent to all parts of the system except at the lowest levels.

- Additional hardware required
  - Journaling — one or two storage units for the user ASP.
  - Checksum protection — one additional storage unit for each checksum set (this is a minimum of one extra storage unit for each seven existing storage units). However, to provide good checksum protection and avoid having units not in a checksum set, often more than the minimum number of extra storage units is required.
  - Mirrored protection — normally twice as many storage units. For concurrent maintenance and higher availability, other disk-related hardware may be required.

Step 2: Deciding Which ASPs to Protect with Mirrored Protection

Mirrored protection is configured by auxiliary storage pool, because the ASP is the user's level of control over single-level storage. Mirrored protection can be used to protect one, some, or all ASPs on a system. However, multiple ASPs are not required in order to use mirrored protection. Mirrored protection works well if all disk units on a system are configured into a single ASP (the default on the AS/400). In fact, mirroring reduces the need to partition auxiliary storage into ASPs for data protection and recovery. However, ASPs may still be desirable for performance and other reasons.

To provide the best protection and availability for the entire system, all ASPs in the system should have mirrored protection:

- If the system has a mixture of some ASPs with and some ASPs without mirrored protection, a disk unit failure in an ASP without mirrored protection severely limits the operation of the entire system. Data can be lost in the ASP in which the failure occurred. A long recovery may be required.
- If a disk fails in a mirrored ASP, and the system also contains ASPs that are not mirrored, data is not lost. However, in some cases, concurrent maintenance may not be possible.

The disk units used in user ASPs should be selected carefully. For best protection and performance, an ASP should contain disk units attached to several different I/O processors. The number of disk units in the ASP that are attached to each I/O processor should be the same (that is, balanced).

Warning: If your system contains 224MB or more of main storage, and the system ASP (ASP 1) contains only two 2800-001 storage units, two additional 2800-001 storage units must be added to the system ASP before mirrored protection can be started. See note 3 on page 14-7 for additional information. (The maximum number of 2800-001 storage units allowed on the system is four. You may need to move 2800-001 storage units from a user ASP to the system ASP.)

Mirrored protection and checksum protection should not be used in different ASPs on the same system. A disk unit failure in an ASP with checksum protection leaves the entire system unusable until the failure is repaired and checksum recovery is completed. For a somewhat larger hardware investment, the ASP can be mirrored.

Step 3: Determining Disk Units Needed for Mirrored Protection

A mirrored ASP requires twice as much auxiliary storage as an ASP that is not mirrored, because the system keeps two copies of all the data in the ASP. Also, mirrored protection requires an even number of storage units of each type of disk unit and model so that storage units can be made into mirrored pairs. On an existing system, it should be noted that it is not necessary to add the same types of disk units already attached in order to provide the required additional storage capacity. Any new disk units may be added as long as sufficient total storage capacity and an even number of storage units of each type are present. The system will assign mirrored pairs and automatically move the data as necessary. If an ASP does not contain sufficient storage capacity, or if storage units cannot be paired, mirror protection cannot be started for that ASP.

The process of determining the disk units needed for mirrored protection is similar for existing or new systems. For each ASP that is to be mirrored, you and your IBM marketing representative should:

1. Plan how much data the ASP will contain.
2. Plan a target percent of storage used for the ASP (how full the ASP will be).
3. Plan the number and type of disk units needed to provide the storage required. For an existing ASP, you can plan a different type and model of disk unit to provide the required storage. You must ensure an even number of each type of disk unit and model.

After planning for all ASPs is completed, plan for spare units, if desired.

Planning for Storage Capacity

For a new system, your IBM marketing representative or remarketing representative can help you analyze your system storage requirements. For an existing system, the current amount of data in the ASP being planned is a useful starting point. The DST or SST Display Disk Configuration Capacity option shows the total size (in megabytes) and the percent of storage used for each ASP on the system. Multiply the size of the ASPs by the percent used to calculate the number of megabytes of data currently in the ASP. In planning future storage requirements for an ASP, system growth and performance should also be considered.

The planned amount of data and the planned percent of storage used work together to determine the amount of actual auxiliary storage needed for a mirrored ASP. For example, if an ASP is to contain 1GB (GB equals 1 073 741 824 bytes) of actual data, it requires 2GB of storage for the mirrored copies of the data. If 50% full is planned for that ASP, the ASP needs 4GB of actual storage. If the planned percent of storage used is 66%, 3GB of actual storage are required. One gigabyte of real data (2GB of mirrored data) in a 5GB ASP results in a 40% auxiliary storage utilization.

Calculating Mirrored Capacity

The Calculate mirrored capacity option does the arithmetic to determine the capacity provided by different numbers and types of disk units for an ASP. The ASP can currently be unprotected, checksum protected, or mirror protected. To plan for a new ASP or for an ASP on a different system, you can enter the number of an ASP that does not currently exist. You can change the amount of data in the ASP and the number of storage units of each disk type to calculate the percent of storage used for various storage units and amounts of data. The amount of data is in megabytes and is the amount of actual data, not the amount of mirrored data (which would be twice the actual data). The number of each disk unit type planned refers to individual storage units, not disk units, because mirroring is by storage unit. The Calculate mirrored capacity option is accessed using either DST or SST.

Planning for Spare Storage Units

Spare storage units can reduce the time the system runs without mirrored protection for a mirrored pair after a storage unit failure. If a storage unit fails and a spare unit of the same disk unit type and model is available, that spare unit can be used to replace the failed unit. Using the DST or SST replace option, the user selects the failed storage unit to replace, then selects a spare storage unit to replace it. The system logically replaces the failed unit with the selected spare unit, then synchronizes the new unit with the remaining good unit of the mirrored pair. Mirrored protection for that pair is again active when synchronization completes (in usually less than an hour). However, it might take several hours from the time a service representative is called until the failed unit is repaired and synchronized, and mirrored protection is again active for that pair.

To make full use of spare units, you need at least one spare unit of each type and model of disk unit you have on your system. This provides a spare for any type of disk unit that may fail. A failed unit must be replaced by a spare of the same disk unit type and model. An exception is the 9332 model 200s and model 400s, which can be used as spares for one another.

Total Planned Storage Capacity Needs

After planning for the number and type of storage units needed for each ASP on the system, and for any spare storage units, add up the total number of storage units of each disk unit type and model. Remember that the number planned is the number of storage units of each disk unit type, not the number of disk units. You and your IBM marketing representative will need to convert the planned number of storage units to disk units before ordering hardware.

The preceding procedure helps you plan the total number of disk units needed for your system. If you are planning for a new system, this is the number that needs to be ordered. If you are planning for an existing system, subtract the number of each disk type currently on your system from the number planned. This is the number of new disk units that should be ordered.

Step 4: Determining the Level of Protection

The amount of duplicate disk-related hardware determines your level of protection. The more mirrored pairs that have higher levels of protection, the more often your system will be usable when disk-related hardware fails. You may decide that a lower level of protection is more cost effective for your system than a higher level.

Figure 19-1 on page 19-4 shows the different levels of protection possible. When determining what level of protection is adequate, you should consider the relative advantages of each level of protection with respect to the following:

- The ability to keep the system operational during a disk-related hardware failure.
- For the 9406, the ability to perform maintenance concurrently with system operations. To minimize the time that a mirrored pair is unprotected after a failure, you may want to repair failed hardware while the system is operating.

Disk Unit-Level Protection

Mirrored protection always provides disk unit-level protection because the storage units are duplicated. If your main concern is protection of data and not high availability, then disk unit-level protection may be adequate. The disk unit is the most likely hardware component to fail, and disk unit-level protection keeps your system available after a disk unit failure.

For the 9406, concurrent maintenance is possible for certain types of disk unit failures with disk unit-level protection.

Controller-Level Protection

If your system has 9335 disk units, then determine if you want controller-level protection based on the following:

- To keep your system available when a 9335 A01 controller fails.
- To concurrently repair a failed disk unit or controller. To use problem recovery procedures in preparation for isolating a failing item or to verify a repair action, the controller (9335 A01) must be dedicated to the repair action. If any disk units attached to the controller do not have controller-level protection, then this part of concurrent maintenance is not possible.

To achieve controller-level protection, all 9335 B01 disk units attached to a 9335 A01 controller must have a mirrored unit attached to a different 9335 A01 controller.

I/O Processor-Level Protection

Determine if you want I/O processor-level protection based on the following:

- To keep your system available when an I/O processor fails.
- To keep your system available when the cable attached to the I/O processor fails.
- To concurrently repair certain types of disk unit failures or cable failures. For these failures, concurrent maintenance needs to reset the I/O processor. If any disk units attached to the I/O processor do not have I/O processor-level protection, then concurrent maintenance is not possible.

To achieve I/O processor-level protection, all disk units attached to an I/O processor must have a mirrored unit attached to a different I/O processor. On many systems, I/O processor-level protection is not possible for the mirrored pair for unit 1. See rule 6 on page 21-1 for restrictions.

Bus-Level Protection

Bus-level protection may allow the system to run when a bus fails. However, bus-level protection is often not cost-effective because of the following:

- If bus 0 fails, the system is not usable.
- If a bus fails, disk I/O operations may continue, but so much other hardware is lost, such as work stations, printers and communication lines, that from a practical standpoint, the system is not usable.
- Bus failures are rare compared with other disk-related hardware failures.
- Concurrent maintenance is not possible for bus failures.

To achieve bus-level protection, all disk units attached to a bus must have a mirrored unit attached to a different bus. On many systems, bus-level protection is not possible for the mirrored pair for unit 1. Rule 6 on page 21-1 has more information on the restrictions for unit 1.

Step 5: Determining the Extra Hardware You Need for Mirroring

In order to communicate with the rest of the system, disk units are attached to controllers, which are attached to I/O processors, which are attached to busses. The number of each of these types of disk-related hardware available on the system directly affects the level of protection that is possible.

To provide the best protection and performance, each level of hardware should be balanced under the next level of hardware. That is, the disk units of each device type and model should be evenly distributed under their controllers. The same number of controllers should be under each I/O processor for that disk type. The I/O processors should be balanced among the available busses.

To plan what disk-related hardware is needed for your mirrored system, you must plan the total number and type of disk units (old and new), that will be needed on the system, as well as the level of protection for the system. It is not always possible to plan for and configure a system so that all mirrored pairs meet the planned level of protection. See 6 on page 21-1 for restrictions. However, it is possible to plan a configuration in which a very large percentage of the disk units on the system achieve the desired level of protection.

When planning for additional disk-related hardware, you should first determine the minimum hardware needed for the planned disk units to function. Plan for one disk unit type at a time. Next, plan the additional hardware needed to provide the desired level of protection for each disk unit type.

Step 5A: Planning the Minimum Hardware Needed to Function.

Various rules and limits exist on how storage hardware can be attached together. The limits may be determined by hardware design, architecture restrictions, performance considerations, or support concerns. Your IBM marketing representative can explain these configuration limits and help you use them in your planning. For a listing of the configuration limits and rules, see *9406 System Installation and Upgrade Guide*.

For each disk unit type, first plan for the controllers needed and then for the I/O processors needed. After planning the number of I/O processors needed for all disk unit types, use the total number of I/O processors to plan for the number of busses needed.

Step 5B: Planning Additional Hardware to Achieve the Level of Protection.

- Bus-level protection

  If you want bus-level protection and already have a multiple-bus system, you need to do nothing. If your system is configured according to standard configuration rules, the mirrored pairing function pairs up storage units to provide bus-level protection for as many mirrored pairs as possible. If you have a single-bus system, a model upgrade is probably not cost-effective merely to achieve bus-level protection. You should plan instead for I/O processor-level protection.

- I/O processor-level protection

  If you want I/O processor-level protection and do not already have the maximum number of I/O processors on your system, add as many I/O processors as possible, keeping within the defined system limits. Then balance the disk units among them according to the standard system configuration rules. It generally is not cost-effective to add another bus (model upgrade) merely to attach more I/O processors, although a model upgrade may be desirable for performance or other reasons.

- Controller-level protection

  If the planned disk units do not require a separate controller (such as the 9332), you will already have controller-level protection for as many units as possible and you do not need to do anything more. If your planned disk units do require a separate controller (such as the 9335 B01), add as many controllers (9335 A01s) as possible, keeping within the defined system limits. Then balance the disk units among them according to the standard system configuration rules.

- Disk unit-level protection

  If you have planned for disk unit-level protection, you do not need to do anything more. All mirrored ASPs have a minimum of disk unit-level protection if they meet the requirements for starting mirrored protection.

Example of Planning for Additional Hardware

In this example, assume you are upgrading your 9406 Model B40. You plan to use all disk units in a single ASP and mirror that ASP.

- "Step 3: Determining Disk Units Needed for Mirrored Protection" on page 20-3:

  You planned for thirteen 9332 600 disk units and four 9335 B01 disk units.

- "Step 4: Determining the Level of Protection" on page 20-5:

  You decided that you want I/O processor-level protection on as many storage units as possible.

- "Step 5A: Planning the Minimum Hardware Needed to Function." on page 20-6:

  For the 9332 model 600 disk units that contain an internal controller, no extra controllers need be planned. Eight 9332 model 600 disk units can be attached to one I/O processor. Therefore, a minimum of two I/O processors are needed for the 9332 disk units.

  The 9335 model B01 disk units require a 9335 model A01 controller. A maximum of four 9335 model B01 disk units can be attached to a 9335 A01 controller. Therefore, you need to plan for one 9335 model A01 controller. The one 9335 A01 can be attached to a single I/O processor.

  After Step 5A, you know that you need a minimum of three I/O processors and one 9335 model A01 controller to go with your disk units planned.

- "Step 5B: Planning Additional Hardware to Achieve the Level of Protection." on page 20-6:

  You planned for only two I/O processors, but have an uneven number of disk units. This provides I/O processor-level protection on all but one mirrored pair of the 9332 disk units. To achieve I/O processor-level protection on all of the 9332 disk units requires a minimum of three I/O processors. However, adding a third I/O processor for the 9332 disk units requires a second bus and a model upgrade.

  In Step 5B, you planned for only one I/O processor for the 9335 disk units. At least two are needed to provide I/O processor-level protection. Since the 9335 B01 disk unit requires a 9335 A01 controller between the disk unit and the I/O processor, you must also plan for an additional 9335 A01 controller. You can attach both 9335 A01 controllers to the single I/O processor previously planned for and achieve controller-level protection for the 9335 disk units. If you add the second I/O processor, you need a second bus and a model upgrade.

  You have two choices:

- Be satisfied with less than I/O processor-level protection for all units. All but one mirrored pair of the 9332 disk units will have I/O processor protection. The 9335 disk units and the other pair of 9332 disk units will have controller-level protection.
- Upgrade to a model B50 and buy two more I/O processors and one more 9335 A01 controller. This provides I/O processor protection for all mirrored pairs except for unit 1. Even though a model upgrade is often not cost-effective merely to achieve a higher level of protection, a model upgrade may be worth consideration if your general processing and capacity needs may grow in the near future.

Step 6: Determining Extra Hardware for Performance

In addition to the hardware required to provide the level of mirror protection you want, your system may need additional hardware to achieve the level of performance you want.

Processing Unit Requirements

Mirrored protection causes a minor increase in central processing unit usage (approximately 1% to 2%).

Main Storage Requirements

Mirrored protection uses the following amounts of main storage:

- Mirrored protection uses approximately 30KB of main storage from the machine pool for general purposes.
- Mirrored protection uses 4KB from the machine pool for each mirrored pair.

During synchronization, mirrored protection uses an additional 68KB for each mirrored pair that is being synchronized. The system uses the pool with the most storage.

I/O Processor Requirements

To maintain equivalent performance after starting mirrored protection, your system should have the same ratio of disk units to I/O processors as it did before. To add I/O processors, you may also need additional controllers (for example, 9335 A01). To add I/O processors, you may need to upgrade your system for additional busses.

Because of the limit on busses and I/O processors, you may not be able to maintain the same ratio of disk units to I/O processors. In this case, system performance may be less.

Step 7: Ordering Your New Hardware

Your IBM marketing representative will assist you in ordering your new hardware using the usual order process. That ordering process allows for any other hardware that may be needed as part of your upgrade, such as additional racks and cables.

Step 8. Planning Your Installation

You must work with your IBM marketing representative to plan for the installation of mirrored protection on your system. The marketing representative will help you determine whether your system is balanced and meets standard configuration rules, as defined in the *9406 System Installation and Upgrade Guide*. The system must be configured according to the standard rules in order for the mirrored pairing function to pair up storage units to provide the best protection possible from the hardware that is available. Your marketing representative will also help you plan for the new units needed to add for each ASP.

If you are planning to start mirrored protection on a new system, that system is already configured according to standard configuration rules. If you are using an older system, it may not follow the standard rules. However, wait until after attempting to start mirrored protection before reconfiguring any hardware.

Planning What ASPs to Create

Plan the user ASPs that will have mirrored protection and determine what units to add to the ASPs. When selecting units to add to the configuration, consider reviewing "Mirrored Protection Configuration Rules" on page 21-1. Also review "Creating a User ASP and Adding Disk Units to the New User ASP" on page 15-1.

In general, the units in an ASP should be balanced across several I/O processors, rather than all being attached to the same I/O processor. This provides better protection and performance.

Step 9. Installing Your New Hardware

When the hardware arrives, your service representative will install the hardware. After the hardware is installed, continue with Chapter 21, "Setting Up Mirrored Protection" on page 21-1 for information on how to add new units and start mirrored protection.

Chapter 21. Setting Up Mirrored Protection

This chapter provides information about mirrored protection configuration rules and the procedures to start mirrored protection.

Mirrored Protection Configuration Rules

The following rules apply when setting up a mirrored protection configuration:

1. Mirrored protection can be started through dedicated service tools (DST) not through system service tools (SST).
2. Mirrored protection is configured by ASP number. The two units of a mirrored pair are configured by the system within an ASP.
3. If the system ASP currently has checksum protection, the option to start mirrored protection cannot be selected for any ASP. To start mirrored protection on an ASP when the system ASP has checksum protection, you must stop checksum protection, continue the IPL past storage management recovery, perform an IPL to DST again, and select the Start Mirrored Protection option for the ASP. After mirrored protection is started on one or more user ASPs, you can perform an IPL to DST and select Start Checksum Protection for the system ASP, and then restore the entire system.
4. Mirrored protection requires an even number of storage units for each type of disk unit in the ASP being mirrored. An uneven number of storage units for any type of disk unit prevents mirrored protection from starting.
5. Two storage units cannot be mirrored if the failure of one unit can cause a data loss on the other unit. For the following current types of disk units, the two units of a mirrored pair are not allowed within the same disk enclosure:
   - 9332 model 400
   - 9332 model 600
   - 9335 model B01

   Additionally, the system attempts to assign the two storage units of a mirrored pair in such a way that if one fails, it can be repaired while the system continues to use the other mirrored unit. For a hardware configuration where this is not possible, the repair of the failing unit must be delayed until the system can be powered down. This is always true on the 9404 and 9402 system units. It may also be true for a failing mirrored unit sharing the same controller or I/O processor as its mirrored unit.
6. The mirrored units for unit 1 of the system ASP can only be at specific input and output addresses on the system. Therefore, the system attempts to assign the mirrored units for unit 1 of the system ASP first. Mirrored protection will not start if valid mirrored units for unit 1 cannot be found.

   The following rules apply to stage 1 hardware

   For 9406 system units:

   a. Both storage units of the mirrored pair for unit 1 of the system ASP must be on Bus 0.

   b. Both storage units of the mirrored pair for unit 1 of the system ASP must be under I/O processors in logical card slot 1 or 2.

Protection

c. If you have the new service processor card, then unit 1 can be attached to the I/O processor attached to logical card slot 1, 2, or 3.

System Model	Service Processor Card Number	Feature Number
B50	21F8195	2505
B60	21F8195	2505
B70	21F8197	2521

d. If the input/output (I/O) processor in logical card slot 1 is not a 6110, 6111, or 6112; or if the I/O processors in logical card slots 1 and 2 control disk units of different types, then the two storage units of the mirrored pair for unit 1 are under the same I/O processor. They are not protected against an I/O processor failure, and concurrent disk maintenance is not always possible.

e. Both storage units of the mirrored pair for unit 1 must be on controllers at address 0 or 1, under their respective I/O processors.

f. The storage units of the mirrored pair for unit 1 may be at any device address under the appropriate I/O processors and controllers.

For 9402 and 9404 systems units:

a. These systems only use the disk unit at address 1 to perform an IPL. The other storage unit for unit 1 can be anywhere.

b. If unit 1 of the system ASP at address 1 fails, the system continues to run. In order for an IPL of the system to occur, the other mirrored unit of the mirrored pair for unit 1 must be moved to address 1.

The following applies to all system units with stage 2 hardware

Both storage units for unit 1 must be attached to the first I/O processor (service processor). Therefore, I/O processor-level protection is not possible.

Starting Mirrored Protection

Before you start mirrored protection for an ASP, the ASP must exist and must have sufficient storage units. If necessary, first create any new ASPs and add the nonconfigured storage units. You do not need to save and restore the system to start mirrored protection. It is assumed that your system backup is current. When you have completed these tasks, complete the following tasks:

Any new disk units installed on your system will be nonconfigured. Use the DST Work with ASP Configuration displays to add the selected new storage units to the ASPs to be mirrored. If you planned for spare storage units, leave those storage units in a nonconfigured status.

Task Overview

You will perform the following steps during this task

1. Access DST options
2. Display disk configuration
3. Add new units
4. Start Mirrored protection

Task 1. Access DST Options

To use DST, do the following:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   `CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)`
3. End all subsystems:

   `ENDSBS SBS(*ALL) OPTION(*IMMED)`

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.
4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   `PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)`
7. When the system has powered down, the IPL or Install the System display appears.

```
                           IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system









Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                    © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

Task 1. Access DST Options

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                       Use Dedicated Service Tools (DST)

Select one of the following:

           1. Perform an IPL
           2. Install the operating system
           3. Work with licensed internal code
           4. Work with disk units
           5. Work with DST environment
           6. Select DST console mode
           7. Start a service tool
           8. Perform automatic installation of the operating system
           9. Work with save storage and restore storage




Selection
    4

F3=Exit          F12=Cancel
```

Task 2. Display the Disk Configuration

```
                          Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information









Selection
     1

F3=Exit         F12=Cancel
```

1. Select option 1 (Work with disk configuration) on the Work with Disk Units display after the disk is attached.

```
                     Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshhold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection









Selection
     1

F3=Exit         F12=Cancel
```

2. Select option 1 (Display disk configuration status) on the Work with Disk configuration display.

Task 2. Display the Disk Configuration

```
                           Display Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Display disk configuration capacity
     3. Display disk configuration protection
     4. Display non-configured units

Selection
     1

F3=Exit         F12=Cancel
```

Note: Disk units on the system are either configured or nonconfigured. Options 1, 2, and 3 on the Display Disk Configuration Status display show information about the configured units. Option 4 shows the non-configured units attached to the system. When you physically attach a unit to the system, it becomes part of the nonconfigured set until you place it in an ASP.

3. Select option 1 (Display disk configuration status) and press the Enter key.

```
                      Display Disk Configuration Status

           Serial
ASP  Unit  Number       Type  Model  Address        Status
 1                                                  Mirrored
        1  10-00A7529   9332   400   0010-0000FFFF  Active
        1  10-00A4936   9332   400   0010-0100FFFF  Active
        2  10-00A7498   9332   400   0010-0300FFFF  Resuming
        2  10-00A7529   9332   400   0010-0001FFFF  Active
        3  10-00A7498   9332   400   0010-0600FFFF  Active
        3  10-00A4936   9332   400   0010-0101FFFF  Active
 2                                                  Unprotected
        8  10-00A7530   9332   400   0020-0701FFFF  Configured
 3                                                  Unprotected
        6  10-00A7530   9332   400   0010-0700FFFF  Configured

Press Enter to continue.

F3=Exit   F5=Refresh    F11=Display non-configured units    F12=Cancel
```

Note: Non-configured units are not shown on this display. To display non-configured units, press F11 (Display non-configured units).

The following fields appear on the Display Disk Configuration Status display:

- *ASP*. The auxiliary storage pool number.

- *Unit*. The number assigned by the system to identify a specific disk unit.
- *Serial Number*. The number assigned by the manufacturer to identify a specific disk unit.
- *Type*. The number assigned by the manufacturer to identify a type of disk unit.
- *Model*. The numbers or letters used to identify the feature level of a specific product type.
- *Address*. Identifies the following:
  - Location of the storage device controller card (columns 1 through 4)
  - Functional controller for the disk unit (columns 5 and 6)
  - Disk unit itself (columns 7 and 8)
  - FFFF (columns 9 through 12)
- *Status*. The valid values for this field are:
  - For ASPs:
    - *Unprotected*. No protection exists for this ASP.
    - *Checksummed*. The units in the ASP are protected by checksum protection if the units are a part of a checksum set.
    - *Mirrored*. All the units in the ASP are protected by mirrored protection.
  - For units in an unprotected ASP.
    - *Configured*.
  - For units in an ASP that has checksum protection.
    - *Checksummed*. Indicates that the unit is part of a checksum set.
    - *Configured*. Indicates that the unit is not part of a checksum set.
  - For units in a mirrored ASP.
    - *Active*. This unit is capable of having data written to it, or read from it.
    - *Suspended*. This unit is not capable of having data written to it, or read from it. The data on this unit is not current. For example, if the disk needs repair action or has been manually suspended, it would be in a *Suspended* state.
    - *Resuming*. The current data is being copied (or will be copied) to this unit from the other active unit of the mirrored pair.
    - *Unknown*. The system configuration mechanism cannot determine what the valid configuration should be.

4. Press F11 (Display non-configured unit) on the Display Disk Configuration Status display.

```
                        Display Non-Configured Units

Serial
Number        Type   Model   Address         Status
10-00A7503    9332    400    0010-0100FFFF   Non-configured
10-00A7503    9332    400    0010-0101FFFF   Non-configured
10-00A3651    9332    400    0010-0400FFFF   Non-configured
10-00A3651    9332    400    0010-0401FFFF   Non-configured


 Press Enter to continue.

 F3=Exit   F5=Refresh  F11=Display disk configuration Status  12=Cancel
```

5. Write down the serial number and address of the unit that you are going to use.
6. Return to the Work with Disk Configuration display by pressing F12 (Cancel) three times.

```
                          Work with Disk Configuration

 Select one of the following:

      1. Display disk configuration
      2. Work with ASP threshold
      3. Work with ASP configuration
      4. Work with checksum configuration
      5. Work with mirrored protection


 Selection
   3

 F3=Exit        F12=Cancel
```

Task 3. Add Units to the ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                        Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information

Selection
     1

F3=Exit         F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

```
                   Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshhold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection

Selection
     3

F3=Exit         F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display.

```
                          Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration








Selection
     4

F3=Exit          F12=Cancel
```

4. Select option 4 (Add units to existing ASP) on the Work with Disk Configuration display.

```
                          Select ASP to Add Units to

Type option, press Enter.
  1=Select

                                      --Protected--  --Unprotected--
Option  ASP  Threshold  Overflow       Size  %Used      Size  %Used
  _      1       90%     No              0   0.00%      600   77.84%
  _      2       90%     No              0   0.00%      200    0.53%
```

5. Type a 1 in the *Option* column to select the ASP.

   After you select an ASP number, the Select Units to Add to ASP display appears.

```
                       Select Units to Add to ASP

Type options, press Enter.
  1=Add unit to ASP

        Serial
Option  Number        Type  Model  Address
  _     10-00A7503    9332   400    0010-0100FFFF
  _     10-00A7503    9332   400    0010-0101FFFF
  _     10-00A3651    9332   400    0010-0400FFFF
  _     10-00A3651    9332   400    0010-0401FFFF

F3=Exit                    F12=Cancel
```

6. Type a **1** in the *Option* column and to select the disk units to place in the specified ASP and pressing the Enter key.

   The Confirm Add Units display shows what the the entire system configuration will be when you add the units. Use the serial number of the disk unit to verify that you are selecting the correct disk units to add to the ASP.

   Note: It is important that you verify that you want the disk units in the specified ASP. If you add a unit to the wrong ASP and want to move it, the source ASP is cleared after the move operation and will require a restore operation to recover the source ASP.

```
                          Confirm Add Units

Units will be added to ASP number  . . . :   _

Add will take several minutes for each unit.  The system will
have the displayed protection after the unit(s) are added.

Press Enter to confirm your choice for 1=Add units.
Press F12=Cancel to return and change your choice.

             Serial
ASP  Unit  Number       Type  Model  Address         Protection       CSS
 1                                                   Unprotected
        1  10-00A5703   9332   200   0010-0100FFFF   Unprotected
        2  10-00A4293   9335   B01   0110-0100FFFF   Unprotected
 2                                                   Unprotected
        5  10-00A3651   9332   400   0010-0400FFFF   Unprotected
        6  10-00A3651   9332   400   0010-0401FFFF   Unprotected

F12=Cancel
```

7. If you are satisfied with the configuration, press the Enter key to add the disk units to the ASP.

Mirrored Protection

Adding units will take several minutes. The time it takes depends on the size of each unit being added and the ability of the system to do multiple adds at the same time.

Task 4. Start Mirrored Protection

1. Return to the Work with Disk Configuration display.

```
                      Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection









Selection
     5

F3=Exit          F12=Cancel
```

2. Select option 5 (Work with mirrored protection) on the Work with Disk Configuration display.

```
                  Work with Mirrored Protection

Select one of the following:

     1. Display disk configuration
     2. Start mirrored protection
     3. Stop mirrored protection
     4. Calculated mirrored capacity










Selection
     2

F3=Exit          F12=Cancel
```

3. Select option 2 (Start mirrored protection) on the Work with Mirror Protection display.

```
                    Select ASP to Start Mirrored Protection

 Type options, press Enter
   1=Select

Option   ASP   Protection
  _       1    Unprotected
  _       2    Unprotected





 F3=Exit         F12=Cancel
```

4. Select the ASP or ASPs to be mirrored on the Select ASP to Start Mirrored Protection display and press the Enter key.

   The system shows a confirmation display of the new mirrored protection configuration, including the levels of protection. Notice that half of the previous unit numbers for the ASP no longer exist. The storage units for those unit numbers have been paired with the storage units for the remaining unit numbers to make mirrored pairs.

```
                     Confirm Start Mirrored Protection

 Press Enter to confirm your choice to start mirrored
   protection.  During this process the system performs an IPL.
   You will return to the DST main menu after the IPL is
   complete.  The system will have the displayed protection.
 Press F12 to return to change your choice.


            Serial
 ASP  Unit  Number      Type  Model  Address        Protection     CSS
  1                                                 Mirrored
         1  10-00A5736  9332   400   0020-0010FFFF  Controller
         1  10-00A5729  9332   400   0010-0000FFFF  Controller
         2  10-00A5736  9332   400   0010-0101FFFF  Controller
         2  10-00A5729  9332   400   0010-0001FFFF  Controller
  2                                                 Unprotected
         3  10-00A5703  9332   400   0010-0201FFFF  Configured
         4  10-00A7498  9332   400   0020-0101FFFF  Configured

                                                                 More...
 F12=Cancel
```

Figure 21-1. Confirm Start Mirrored Protection

5. If the configuration is what you had planned and you do not have other configuration changes to make, turn the keylock switch to the **Normal** position and press the Enter key to accept the configuration.

   The system continues the start mirrored protection process in "Start Mirrored Protection Processing" without further operator intervention.

If the configuration is not what you had planned, for example, the level of protection is less, you have the following options:

- Verify that the correct ASP was selected. Verify that any new storage units have been added to the correct ASP. If a storage unit was added to a wrong ASP, complete start mirroring, then refer to "Removing Units from a Mirrored ASP without Clearing the ASP" on page 22-35, for a method to get the storage unit into the correct ASP without clearing the ASP.
- Determine if additional hardware is required to achieve the planned level of protection. Review "Step 5: Determining the Extra Hardware You Need for Mirroring" on page 20-6. Contact your IBM marketing representative for assistance.
- Determine if the existing hardware needs to be connected differently to achieve the planned level of protection. Review "Step 8. Planning Your Installation" on page 20-8 and refer to "Reconfiguring Your System" on page 24-2 for information on how to achieve your planned level of protection. Contact your IBM marketing representative for assistance.
- Consider continuing the start mirrored protection process which will provide better availability than non-mirrored protection, rather than waiting until additional hardware arrives so that you can achieve your planned level of protection. After you receive and install the additional hardware, do a Stop Mirrored Protection procedure followed by a Start Mirrored Protection procedure to achieve the best level of protection for the new hardware configuration. Even on very large systems, the tasks to stop mirroring, add units, and start mirrored protection can be done in several hours.

6. After the system reaches the Command Entry display, create the QSYSMSG message queue to receive messages. See "Message Handling" on page 22-56 for more information.

Start Mirrored Protection Processing

The following steps are performed by the system when mirrored protection is started.

1. Data is moved off half of the storage units in the selected ASPs. This can take from a few minutes to a few hours, depending on the amount of data that must be moved.

   Objects created on a preferred unit may be moved to another unit. The preferred unit number may no longer exist when mirrored protection is started.

2. New control information is written to disk, describing the new mirrored system configuration.
3. After the data is moved and the control information is written, the system performs an IPL.
4. When the system reaches DST, the previously selected ASPs are mirrored, although the two storage units in the mirrored pairs are not yet synchronized.

   If the keylock switch is in the Manual position, you have the option to perform other configuration changes or perform an IPL. If you do not have configuration changes to make, select the option to perform an IPL and press the Enter key.

If the keylock switch is in the Normal position, the system automatically continues the IPL.

5. When the system continues the IPL past DST, the mirrored pairs are synchronized during storage management recovery. This can take a few hours, although this long recovery time only occurs when mirrored protection is first started, and not during every IPL on a mirrored system. On very large systems, the entire start mirrored protection process can take up to approximately eight to ten hours.
6. After storage management recovery is completed, the selected ASPs have mirrored protection. If the keylock switch is in the Normal position, the command entry display is shown. If the keylock switch is in the Manual position, the Sign On display is shown.

Mirrored Protection Configuration Errors

Starting mirrored protection requires that the storage management directories are good and there are no missing or suspended mirrored units in the configuration. You must do the following before mirrored protection can start:

- If the directories are not good, you need to recover them through storage management recovery by performing an IPL of the system.
- Units with a status of missing, must be powered on, repaired, or replaced.
- Units with a status of suspended in a mirrored ASP need to be repaired or replaced, and resumed before mirrored protection can start in another ASP.
- If you have stopped checksum protection on an ASP, and you want to start mirrored protection on the same ASP, you must IPL the system through storage management recovery. You cannot start mirrored protection on an ASP that has checksum protection.

Start mirrored protection can fail if there is insufficient storage available in the ASP to contain the current data in the ASP. The percentage used in the ASP must be less than half of the ASP threshold.

There must be sufficient storage units in the ASP for the system to create mirrored pairs. If you receive a message indicating that the system cannot pair unit 1 or other units, review "Mirrored Protection Configuration Rules" on page 21-1.

Mirrored Protection

Chapter 22. Managing the Mirrored Environment

This chapter provides information about managing and recovering the mirrored environment on your system. SST and DST are the tools which provide menu options for you to display, change, or recover the mirrored environment. For more information about these options, see "Overview of SST and DST Options" on page 14-16.

Management Options

The following configuration changes are only allowed when all currently configured disk units are present and are not suspended. However, units that are being made nonconfigured may be suspended. All remaining units must be present and not suspended.

- Add units
- Start mirrored protection
- Start checksum protection
- Stop checksum protection
- Move units
- Replace units

The following configuration changes are allowed when the units that are being made nonconfigured units are missing or suspended.

- Stop mirrored protection
- Remove units

Adding Disk Units to a Mirrored ASP

You can add storage units to an ASP while mirrored protection is in effect to increase the storage capacity of that ASP. When mirrored protection is in effect, you must add an even number of each type of storage unit. The number of mirrored pairs added are half the number of storage units added.

It is not necessary to perform a save and restore operation when adding units to a mirrored ASP. You should have a current backup copy of your system.

The following apply to adding units:

- Adding units is only allowed using DST, before an IPL.
- All configured storage units must be operational.
- The new units are automatically paired with one another or with existing units as part of the add operation to provide the best level of protection possible.
- Adding units is not allowed if the unit 1 has been saved by the DST Save Disk Unit Data function. This is because a restore of the saved information would clear the configuration change.
- An even number of storage units of each type are required.

Adding a mirrored unit takes at least twice as long as adding a unit that is not mirrored because each unit is a pair of mirrored units. Later, during the IPL, the added units are also synchronized. This can take up to 30 minutes for each mirrored unit.

Adding a mirrored unit can result in a lower level of protection. The Confirm Add Units display shows the level of protection that results from the add operation.

The following assumes that the service representative has attached the new disk units and they are in nonconfigured status.

Task 1. Calculate Mirrored Capacity Using SST

Use the SST Calculate Mirrored Capacity displays to determine if the new disk units can provide the required capacity.

1. Type the following on the command line and press the Enter key.

   STRSST

   The System Service Tools (SST) menu is shown.

2. Select option 3 (Work with disk units) on the System Service Tools (SST) menu and press the Enter key.

```
                         Work with Disk Units

 Select one of the following:

      1. Display disk configuration
      2. Display checksum configuration
      3. Calculate checksum configuration
      4. Work with ASP threshold
      5. Work with disk unit recovery
      6. Work with disk unit information
      7. Calculate mirrored capacity





 Selection
     7

 F3=Exit          F12=Cancel
```

3. Select option 7 (Calculate mirrored capacity) on the Work with Disk Units display.

   The Specify ASP to Calculate Mirrored Capacity display is shown.

```
                Specify ASP to Calculate Mirrored Capacity

 Type choices, press Enter.

   ASP . . . . . . . . . . . . . .   _       1-16

 F3=Exit   F11=Display configuration capacity    F12=Cancel
```

4. Type the number of the ASP in the *ASP* prompt and press the Enter key.

   The Calculate Mirrored Capacity display is shown.

```
                         Calculate Mirrored Capacity

 ASP . . . . . . . . . . . . . . :            2

 Type choices, press Enter.

 Specify required size . . . . . . . . .   1000       Megabytes

  -------- Number of Units ------
    Planned              Current       Type      Model        Size
        0                    0         6100       015          315
        0                    0         6105       010          320
        8                    0         6105       020          320
        0                    0         6105       030          320
        0                    0         6107       040          320
        0                    0         6107       010          400
        0                    0         6107       020          400
        0                    0         6107       030          400
        0                    0         6107       040          400

 F3=Exit   F11=Display configuration capacity    F12=Cancel
```

5. Type the required size of the ASP in the *Specify required size* field if the ASP does not exist.
6. Type the number of planned storage units in the *Planned* prompt in the *Number of Units* column and press the Enter key.

   The Capacity Provided by Mirrored Protection display is shown.

Mirrored Protection

```
                    Capacity Provided By Mirrored Protection

ASP . . . . . . . . . . . . . . . :          2
Required size . . . . . . . . . . :       1000   Megabytes
Actual size  . . . . . . . . . . :       1280   Megabytes
Planned Amount used. . . . . . . .:         78   %
 Specify required size . . . . . . . . . .      ______           Megabytes

  -------- Number of Units ------
    Planned              Current       Type     Model         Size
        0                    0         6100      015
        0                    0         6105      010
        8                    0         6105      020
        0                    0         6105      030
        0                    0         6107      040
        0                    0         6107      010
        0                    0         6107      020
        0                    0         6107      030
        0                    0         6107      040


 F3=Exit   F11=Display configuration capacity    F12=Cancel
```

7. Exit SST by pressing F3 (Exit) three times.
8. On the Exit System Service Tools display, press the Enter key to end SST.

Task 2. Access DST Options

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   `CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)`

3. End all subsystems:

   `ENDSBS SBS(*ALL) OPTION(*IMMED)`

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.

4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   `PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)`

7. When the system has powered down, the IPL or Install the System display appears.

```
                    IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system









Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                              © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

Protection

```
                              Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage





Selection
    4

F3=Exit            F12=Cancel
```

Task 3. Adding New Units

After the disk units are attached, they are shown as nonconfigured units on the DST Display Nonconfigured Units display.

When selecting disk units to place in an ASP that has mirrored protection, you need to select an even number of units of each type.

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                                Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information








Selection
     1

F3=Exit         F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units display.

```
                    Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP thresholds
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection

Selection
     3

F3=Exit        F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display.

```
                         Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration

Selection
     4

F3=Exit        F12=Cancel
```

4. Select option 4 (Add units to existing ASP) on the Work with Disk Configuration display.

```
                         Select ASP to Add Units to

 Type option, press Enter.
   1=Select

                                        --Protected--  --Unprotected--
 Option  ASP  Threshold  Overflow       Size  %Used      Size  %Used
  1       1       90%      No            0   0.00%       600   77.84%
  _       2       90%      No            0   0.00%       200    0.53%
```

5. Type a 1 in the *Option* column to select the ASP.

   After you select an ASP number, the Select Units to Add to ASP display appears.

```
                         Select Units to Add to ASP

 Type options, press Enter.
   1=Add unit to ASP

         Serial
 Option  Number        Type  Model  Address
   _     10-00A7503    9332   400    0010-0100FFFF
   1     10-00A7503    9332   400    0010-0101FFFF
   _     10-00A3651    9332   400    0010-0400FFFF
   1     10-00A3651    9332   400    0010-0401FFFF









 F3=Exit                       F12=Cancel
```

6. Type a **1** in the *Option* column and to select the disk units to place in the specified ASP and pressing the Enter key.

   The Confirm Add Units display shows what the the entire system configuration will be when you add the units. Use the serial number of the disk unit to verify that you are selecting the correct disk units to add to the ASP.

   Note: It is important that you verify that you want the disk units in the specified ASP. If you add a unit to the wrong ASP and want to move it, the source ASP is cleared after the move operation and will require a restore operation to recover the source ASP.

```
                         Confirm Add Units

 Units will be added to ASP number . . . :    1

 Add will take several minutes for each unit.  The system will
 have the displayed protection after the unit(s) are added.

 Press Enter to confirm your choice for 1=Add units.
 Press F12=Cancel to return and change your choice.


            Serial
 ASP  Unit  Number       Type  Model  Address          Protection      CSS
  1                                                    Unprotected
         1  10-00A5703   9332   200   0010-0100FFFF    Unprotected
         2  10-00A4293   9335   B01   0110-0100FFFF    Unprotected
  2                                                    Unprotected
         5  10-00A7503   9332   400   0010-0101FFFF    Unprotected
         6  10-00A3651   9332   400   0010-0401FFFF    Unprotected


 F12=Cancel
```

7. If you are satisfied with the configuration, press the Enter key to add the disk units to the ASP.

   Adding units will take several minutes. The time it takes depends on the size of each unit being added and the ability of the system to do multiple adds at the same time.

Moving a Disk Unit from a Mirrored ASP to Another ASP

Units cannot be moved to, or from, a mirrored ASP using the DST move function. When there are mirrored ASPs, units must first be removed from one ASP and then added to another ASP, using the remove function and then the add function. The ASP from which units are removed is cleared. All data in that ASP is lost. Therefore, you should backup your system before removing any units.

Removing Units from the System ASP that Has Mirrored Protection

When a unit is removed from an ASP that has mirrored protection, both mirrored units of that mirrored pair are removed. When a unit is removed from an ASP, the ASP is cleared.

Task Overview

You will perform the following steps during this procedure:

1. Save the System
2. Access DST
3. Remove the unit from the system ASP
4. Restore the licensed internal code using function code 23.
5. Restore the operating system
6. Restore the remaining objects to the system

Warning: Use caution when removing a unit from an ASP. When the system removes a unit from an ASP, all objects are destroyed unless the unit that does not have checksum protection is being removed from an ASP that has checksum protection

Before you begin...

The following conditions apply when you remove a unit:

- When removing a disk unit from the system ASP, the licensed internal code must be restored using function code 23 (restore).
- You cannot remove unit 1 from the system ASP. (Unit 1 is reserved for the system licensed internal code.)
- Any overflowed use ASP will be cleared if a unit is removed from the system ASP.
- All data in the ASP from which you are moving the unit is destroyed except when you remove a unit that is not checksum protected from an ASP that is checksum protected.

In the following example, a unit currently in the system ASP is being removed.

Do the following tasks to remove a disk unit:

Task 1. Save the Entire System

If you are removing a unit from the system ASP, save the entire system with the appropriate save commands.

Notes:

1. When saving the entire system, it is important for you to remember to display the system log (QHST) and move or delete all but the current log. These steps will improve the performance of the save system operation.
2. To significantly reduce the time it takes to restore the system, consider specifying SAVLIB LIB(*NONSYS) ACCPTH(*YES).
3. If your application programs are dependent on spooled output files, you can save them. The system does not directly support saving spooled output files. However, you can save the files by copying them (CPYSPLF command) before you perform a save system operation. For an example of how to save the spooled ouput files, see the topic "Example of Saving Spooled Output Files Using the Copy Spooled File (CPYSPLF) Command" on page 9-22.

There are two methods you can use to save the entire system:

1. Using option 21 (Entire system) on the Save menu allows you to save the entire system without entering the commands.
2. Using the Save commands allows you to save the entire system by entering the commands from the command line.

Method 1. Using Option 21 (Entire system) on the Save Menu

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:

   `WRKACTJOB`

3. Display the system log QHST to verify it is up to date:

   `DSPLOG LOG(QHST)`

   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:

   `WRKF FILE(QSYS/QHST*)`

   Look at the list to verify that you saved all copies of the log that will be needed later.
5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.
6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   `DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)`

7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   `DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)`

   Keep this list with your backup log or your save system tapes for future reference.
8. Go to the Save menu:

   `GO SAVE`

   The Save menu is shown.

```
 SAVE                                  Save
                                                          System:   RCHAS791
 Select one of the following:

      1. Files
      2. Libraries
      3. Documents and folders
      4. Programs
      5. Other objects
      6. Changed objects only
      7. Licensed programs
      8. Security data
      9. Storage

     20. All libraries other than system library
     21. Entire system
     22. All IBM libraries other than system library
                                                                 More...
 Selection or command
 ===>

 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
 F16=AS/400 Main menu
     (C) COPYRIGHT IBM CORP. 1980, 1991.
```

Mirrored Protection

9. Select option 21 (Entire system) from the Save menu and press the Enter key.

```
                         Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .   TAP01        Names
                                   ______
                                   ______
                                   ______

  Prompt for commands  . . . . .   Y            Y=Yes, N=No

  Check for active files . . . .   Y            Y=Yes, N=No
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device when the first tape is full.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting (for unattended save operations) and uses the default values.

Check for active files

Allows you to specify whether or not you want to check for active files. If you specify Y=Yes, the system sends a message when active files are encountered. You can end the checking process or clear the existing files and continue. If N=No is specified, all active files encountered during the save are cleared.

Option 21 will guide you through the following if you selected Y on the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS SBS(*ALL) OPTION(*IMMED)
b. SAVSYS
c. SAVLIB LIB(*NONSYS) ACCPTH(*YES)
d. SAVDLO DLO(*ALL) FLR(*ANY)
e. STRSBS SBSD(controlling-subsystem)

If you want to be notified when the subsystems are ended, change the QSYSOPR message queue by typing the following and pressing the Enter key.

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
```

Messages are sent to the QSYSOPR message queue indicating when the subsystems have ended and the system is in a restricted state.

10. Continue loading tapes when the system sends a message asking you to load the next volume.

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be restored beyond the failure by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
           OUTPUT(*PRINT)
   ```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.
3. Load a new tape and type the following:

   ```
   SAVLIB LIB(*NONSYS) STRLIB(library-name)
   ```

   Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

Method 2. Using the Save commands

If you do not want to use option 21, you can do the following steps from the command line of a menu:

1. Sign on the system as QSECOFR.
2. Verify that no users are on the system and that no batch jobs are running:

   ```
   WRKACTJOB
   ```
3. Display the system log QHST to verify it is up to date:

   ```
   DSPLOG  LOG(QHST)
   ```

   Displaying the QHST log automatically brings it up to date.
4. Display all copies of the system log:

   ```
   WRKF FILE(QSYS/QHST*)
   ```

   Look at the list to verify that you saved all copies of the log that will be needed later.
5. Select the Delete option on the display to delete all but the current copies of the system log to prevent confusion about the date of the log. This step improves the performance of the SAVSYS command.
6. Print a list of all the libraries on the system. You can use this list later if you need to restore a single library.

   ```
   DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)
   ```
7. Print a list of all the licensed internal code fixes currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(*ALL) OUTPUT(*PRINT)
   ```

Keep this list with your backup log or your save system tapes for future reference.

8. Change the QSYSOPR message queue:

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK)  SEV(60)
```

9. End all subsystems:

```
ENDSBS SBS(*ALL) OPTION(*IMMED)
```

   Messages are sent to the QSYSOPR message queue indicating that the subsystems have ended and the system is in a restricted state.

10. Load the first tape and make the tape device ready.

11. Save the system:

```
SAVSYS DEV(TAP01) ENDOPT(*LEAVE)
```

12. When a message similar to the following appears, load the next tape or make the device ready, and then enter R.

```
Device was not ready or next volume was not loaded (C R)
```

13. Save all user and IBM libraries:

```
SAVLIB LIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
       ACCPTH(*YES)
```

If a media error occurs...

If an unrecoverable media error occurs during the SAVLIB procedure, you can restart the procedure using the STRLIB parameter on the SAVLIB command.

1. Check the job log to determine the library where the previous RSTLIB SAVLIB(*NONSYS) failed. Find the last library saved (successful completion message).
2. Determine the next library to be restored beyond the failure by by looking at the listing you created before you started the save operation using the DSPOBJD command, or enter the following to create a list.

```
DSPOBJD OBJ(QSYS/*ALL) OBJTYPE(*LIB)
        OUTPUT(*PRINT)
```

   The libraries are listed in the order that they are saved. The correct starting library is normally the one listed after the last library saved successfully.
3. Load a new tape and type the following:

```
SAVLIB LIB(*NONSYS) STRLIB(library-name)
```

   Note: A restore of the system using this set of media will require two RSTLIB SAVLIB(*NONSYS) commands to restore all libraries.

14. Save the documents, folders, and distribution documents:

```
SAVDLO DLO(*ALL) FLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
```

15. Start the subsystems:

```
STRSBS SBSD(controlling-subsystem)
```

Task 2. Access DST Options

Sign on to DST using the following steps:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   `CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)`

3. End all subsystems:

   `ENDSBS SBS(*ALL) OPTION(*IMMED)`

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.

4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   `PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)`

7. When the system has powered down, the IPL or Install the System display appears.

```
                          IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system









Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                               © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

Mirrored Protection

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                      Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage

Selection
     4

F3=Exit            F12=Cancel
```

Task 3. Remove the Unit From the System ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                          Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information









Selection
     1

F3=Exit        F12=Cancel
```

2. Select option 1 (Work with disk configuration) on the Work with Disk Units Display and press the Enter key.

```
                     Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection








Selection
     3

F3=Exit        F12=Cancel
```

3. Select option 3 (Work with ASP configuration) on the Work with Disk Units display; the following display is shown.

Protection

```
                          Work with ASP Configuration

 Select one of the following:

      1. Display disk configuration capacity
      2. Create User ASP
      3. Delete user ASP
      4. Add units to existing ASP
      5. Reset overflowed user ASPs
      6. Change ASP storage threshold
      7. Move units from one ASP to another
      8. Remove unit from configuration








 Selection
      8

 F3=Exit          F12=Cancel
```

4. Select option 8 (Remove unit from configuration) to remove the desired unit. The following display is shown.

   Note: If any user ASP has overflowed, reset the ASP before removing the unit from the system ASP. For more information on how to reset the ASP, see "Resetting an Overflowed User ASP" on page 15-67.

```
                         Remove Units from Configuration

 Type options, press Enter.
    4=Remove unit from the configuration

                     Serial
 OPT  Unit  ASP  Number  Type   Model  Address        Status
        1    1           9332   400    0010-0000FFFF  Mirrored
        1    1           9332   400    0010-0100FFFF  Mirrored
        2    1           9332   400    0010-0300FFFF  Mirrored
        2    1           9332   400    0010-0001FFFF  Mirrored
        3    1           9332   400    0010-0600FFFF  Mirrored
        3    1           9332   400    0010-0101FFFF  Mirrored
        7    2           9332   200    0010-0201FFFF  Configured
        8    3           9332   200    0020-0701FFFF  Configured
        9    4           9332   200    0010-0700FFFF  Configured

                                                                    More...
 Press Enter to continue.

 F3=Exit   F5=Refresh    F11=Display non-configured units    F12=Cancel
```

5. Select the unit you want removed from the configuration by typing a **4** in the *OPT* column and pressing the Enter key.

   Note: More than one unit can be removed at the same time. Removing a unit from a mirrored or unprotected ASP will clear the ASP.

   The Confirm Remove of Units display is shown.

```
                         Confirm Remove of Units

Warning: Removing a unit from an ASP will destroy all the data
from that ASP.

Press F10 to confirm your choice to remove the unit.
Press F12=Cancel to return and change your choice.
                 Serial
 Opt  Unit  ASP  Number  Type  Model  Address        Status
         3    1          9332   400   0010-0101FFFF  Configured
         3    1          9332   400   0010-0600FFFF  Configured













                                                              Bottom


 F10=Confirm   F12=Cancel
```

6. To confirm the remove of the unit, press F10 (Confirm). Press F12 (Cancel) to return and change your choice.

   All data in the ASP is destroyed unless you are removing a unit that does not have checksum protection from an ASP that has checksum protection.
7. Press F3 (Exit) until you return to the Use Dedicated Service Tools menu.
8. Select option 7 (Start a service tool) on the Use Dedicated Service Tools menu and press the Enter key.
9. Select option 8 (Power off the system) on the Start a Service Tool menu.

Task 4. Restore the Licensed Internal Code

1. Restore the licensed internal code to recover the PTF index if the unit was removed from the system ASP by doing the following:
2. Ensure the key is in the keylock switch on the control panel.
3. Turn the key to the Manual position.
4. Press the Function Select switch to display **02** in the Function display on the control panel.
5. Press the Enter button on the control panel.
6. Select IPL type D (this specifies that the IPL source comes from tape) by pressing the Function Select switch on the control panel until **D** is shown on the Data display.
7. Press the Enter button on the control panel.
8. For the 9406 system unit, ensure that the power switches for the IPL tape unit and all disk units are in the On position.
9. Find the licensed internal code tape, which is the first volume of the most recent set of save tapes.
10. Place the tape in the tape unit used for the IPL. See the setup manual for the device for more information on loading the tape.

Note: If your tape unit cannot be loaded when the power is off, continue with the next step. You will be prompted later by an SRC code for the tape.

11. Turn on the power to the system by pushing the Power switch up. The switch returns to center after you push it. The 9402 system unit has a green button labeled Power On.
12. Wait as explained below for the tape unit to power on. See the following explanations:

    Notes:

    a. SRC A100 1938 (Tape not found) may be displayed along with the System Attention light until the tape unit is turned on. Continue with the next step.
    b. SRC A100 1933 or 1934 (Tape not ready) may also be displayed along with the System Attention light. When the tape is ready, continue with the next step. If this SRC is displayed for more than 1 or 2 minutes, call your service representative.
    c. SRC 2507 XXXX appears if manual select IPL intervention is required. If intervention is required, press the Function Select switch until 03 appears. Press the Enter button on the control panel.
13. Ensure that the tape is online or ready. No action is required for tape units that perform this step automatically (such as the tape cartridge unit).
14. Ensure that the console display is turned on.
15. Wait for the yellow System Attention light on the control panel to light up.

    There is a delay while the system loads information from the tape. SRCs showing status are continuously updated on the control panel while processing occurs. This can take from 5 to 20 minutes; the time varies depending on the speed of the tape unit and the processor speed for the specific system model.

    When SRC A6xx 6001 is displayed, the system is prepared to start installing or restoring the licensed internal code on the disk unit containing unit 1. Continue with the next step.
16. Select the correct function code by pressing the Function Select switch on the control panel until the correct function code is displayed. (See the "Before You Begin" section in this topic to determine the correct function code.)

    Warning: Option 24 (Install) is used only to recover from the loss of unit 1 in the system ASP, or to recovery from a disaster. Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

    If you selected function code 24, the System Attention light light may appear in one or two minutes and SRC `A6xx 6002` is shown. If you are sure you want to install the licensed internal code, select function code 24 again.

Notes:

a. When SRC A6xx 6002 is displayed, option 23 can also be selected. If option 23 is selected, a restore of the licensed internal code is performed (not an install). Data on the disk units will not be lost.

b. If another SRC appears after A6xx 6001 that is not in the A6xx xxxx format, then the system needs additional attention. Call your service representative.

   If the following SRCs appear after SRC A6xx 6001 is displayed, see Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" for an explanation of these SRCs and the steps to follow.

   - A6xx 6002 Disk unit may contain a valid system
   - A6xx 6003 Disk unit not currently a load source
   - A6xx 6004 Disk unit not currently a load source
   - A6xx 6005 Disk unit not found

17. Press the Enter button on the control panel.

    Note: The system will start displaying status SRCs again, which will be continuously updated to show the status of installing the licensed internal code. An example of a status SRC is D6xx 6201 (stand-alone install operations is running).

18. If the yellow System Attention light is on again, and SRC A6xx 6048 (New tape volume needs to be loaded) appears, the system needs the next tape. The xx tells which volume needs to be loaded. Load the correct tape and make the device ready. The install operation will automatically continue.

    If SRC A6xx 6051 appears, the stand-alone function is requesting the model-unique licensed internal code tape found in the service kit. Unload the current tape from the tape device and load the model-unique licensed internal code tape.

    - A6xx 6051 Model-unique licensed internal code tape needs to be loaded

    - A6xx 6052 Tape loaded was not the model-unique licensed internal code,

    If another SRC A6xx xxxx is displayed, look up the displayed SRC in Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" on page A-1 and follow the instructions. For all other SRCs call your service representative.

19. After the install or restore operation of licensed internal code or the model-unique licensed internal code is complete, the system will automatically perform an IPL.

20. Remove the model-unique licensed internal code tape from the tape unit (if it is loaded) when the IPL or Install the System menu is shown.

21. If you have other restore steps to perform and the model-unique licensed internal code tape was loaded, load the previous save tape.

Task 5. Restore the Operating System

Load the SAVSYS media in the tape unit and continue with the following:

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                          IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

   Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

```
                    Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.










 F12=Cancel
```

3. Press the Enter key.

4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                        Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .  2924









F3=Exit         F12=Cancel
```

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                    Confirm Language Feature Selection

Language feature  . . . . . . . . . . . . . . . . . :   2924

Press Enter to confirm your choice for language feature.
  Installing the system will continue.
Press F12 to return to change your choice for
  language feature.








F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. Status messages are displayed.

The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                         Running IPL Step

Current IPL step . . . . : Storage Management Recovery
```

After the IPL steps complete, the Install the Operating System menu appears.

```
                        Install the Operating System

Type choices, press Enter.

    Default
      option . . . .  _               1=Take defaults, show no
                                        other installing displays
                                      2=Change installing options

    Date:
      Year . . . . .  _               00-99
      Month  . . . .  _               01-12
      Day  . . . . .  _               01-31

    Time  . . . . . . _       00-23   HH is hours
                      _       00-59   MM is minutes
                      _       00-59   SS is seconds
```

8. When the Install the Operating System display is shown, use the following information to respond to the prompts.

Default Option

Value	**Description**
1	Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation. If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.
2	If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system.

If you select 2 for *Default option*, the Installing Options display appears.

Date

The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

Time

The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

9. Press the Enter key.

   If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages are for your information only. Continue loading tapes in sequence when messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

   If you selected option 2 (Change installing options), the following display is shown.

```
                         Specify Install Options

Type choices, press Enter.

    Restore option . . . . _          1=Restore programs and language
                                        objects from current tape
                                      2=Do not restore programs or
                                        language objects
                                      3=Restore only language objects
                                        from current tape
                                      4=Restore only language objects
                                        from a different tape




    Job and output
      queue options. . . . 1          1=Clear,  2=Keep
```

10. When the Installing Options display appears, use the following information to respond to the prompts.

    Restore Option

Value	**Description**
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

Mirrored Protection

Notes:

a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.

b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.

c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.

d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.

e. The system entry-point table is created again.

f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	Description
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                    Specify Restore Options

Type choices, press Enter.

    Restore from tape:

      System values . . . . . 2      1=Restore,  2=Do not restore
      Edit descriptions . . . 2      1=Restore,  2=Do not restore
      Message reply list. . . 2      1=Restore,  2=Do not restore
```

11. Using the following information, respond to the prompts on the Restore Options display.

System Values

Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID Default system code page and character set

QCURSYM Currency symbol

QDATFMT Date editing format

QDATSEP Date separator character

QDECFMT Decimal data editing format

QKBDTYPE Default work station keyboard type

QLEAPADJ Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Value	Description
1=Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

12. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

13. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    `Load the next volume on the tape device (1 2).`

14. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    `Operating system has been installed.  IPL in progress.`

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

15. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).

16. Press the Enter key. Informational messages are displayed.

17. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.

18. When the IPL Options display is shown, respond to the prompts using the following information.

```
                              IPL Options

Type choices, press the Enter key.

  System date . . . . . . . . . . . . . .  07 / 26 / 88   MM / DD / YY
  System time . . . . . . . . . . . . . .  12 : 00 : 00   HH : MM : SS
  Clear job queues . . . . . . . . . . . . N              Y=Yes,  N=No
  Clear output queues . . . . . . . . . .  N              Y=Yes,  N=No
  Clear incomplete job logs . . . . . . .  N              Y=Yes,  N=No
  Start print writers . . . . . . . . . .  Y              Y=Yes,  N=No
  Start this device only . . . . . . . . . N              Y=Yes,  N=No


  Set major system options . . . . . . . . N              Y=Yes,  N=No
  Define or change system at IPL . . . . . N              Y=Yes,  N=No







Last power-down operation was normal
                                   (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure 22-1. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a Y to start the console device only, or type an N to start all devices.

d. Enter the value for set major system options.

   Type an N for each option to leave them unchanged, or type Y to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to N. If you installed the licensed internal code using option 24 (Install), the default value is set to Y.

19. Press the Enter key.

20. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

```
                  Edit Rebuild of Access Paths                RCHAS331
                                                      05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                  ------------Access Paths----------  Unique  Rebuild
Seq   Status      File        Library     Member      Keyed   Time
25__  IPL         FILE234512  LIBRARY111  MBR1234567  No      00:00:56
25__  IPL         FILE234513  LIBRARY111  MBR1234567  No      00:00:56
75__  IPL         FILE234514  LIBRARY111  MBR1234567  Yes     00:00:56
75__  IPL         FILE234515  LIBRARY111  MBR1234567  Yes     00:00:56
88__  IPL         FILE234516  LIBRARY111  MBR1234567  No      00:00:56
99__  AFTIPL      FILE234517  LIBRARY111  MBR1234567  Yes     00:00:56
*OPN  OPEN        FILE126789  L123456789  MBR4567890  Yes     12:34:56
*OPN  OPEN        FILE346789  L123456789  MBR4567890  No      12:34:56
*HLD  HELD        F123336789  L123456789  MBR4567890  No      10:30:06
*HLD  HELD        F123456789  L123456789  MBR4567890  Yes     99:56:01
                                                             More...


F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                        Display Access Path Status

IPL Threshold . . . . . . : 88

            ----------Access Paths----------   Rebuild     Current
Status      File         Library    Member     Build Time  Run Time
RUN         F123456789   L123456789 MBR4567890             00:00:01
JRN         F123456789   L123456789 MBR4567890
JRN         F123456789   L123456789 MBR4567890
JRN         F123456789   L123456789 MBR4567890
JRN         F123456789   L123456789 MBR4567890
JRN         F123456789   L123456789 MBR4567890
JRN         F123456789   L123456789 MBR4567890
SYS         F123456789   L123456789 MBR4567890  12:34:56
SYS         F123456789   L123456789 MBR4567890  12:34:56
IPL         F123456789   L123456789 MBR4567890  12:34:56
                                                              More...




F3=Exit and continue IPL   F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

Protection

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN – indicates that the access path is being rebuilt.
- SYS – indicates the access path is a system access path and is waiting to be rebuilt.
- JRN – indicates that the access path is being recovered from its associated journal.
- IPL – indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

21. Press the Enter key to continue.
22. Ensure the keylock switch is in the Normal position.
23. This completes the restore operation for the operating system if you have no other restore steps to perform.
24. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

After the restore process is complete, the continue with the following:

Task 6. Restore the Remaining Objects to the System

There are three methods to restore the remaining objects to the system:

- Method 1 is used to restore objects if no user ASPs exist or if you want to avoid a more complicated set of restore steps.
- Method 2 is used to restore the system ASP and recover journals, journal receivers, or save files in user ASPs.
- Method 3 is used to restore the system and recover libraries and their associated objects in user ASPs.

Select the method to use and continue with the following steps.

1. Type the following to reclaim storage:

   ```
   RCLSTG
   ```

   If you had journals, journal receivers, or save files (whose library is in the system ASP) stored in user ASPs when you began this procedure, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile.

   If you had libraries and associated objects stored in a user ASP, this command will restore their addressability and transfers ownership of the objects to QDFTOWN user profile.

If objects are damaged in the ASP, this command places the objects in library QRCL and transfers ownership of the objects to QDFTOWN user profile. This happens whether you have isolated objects or an entire library in an ASP.

2. Restore user profiles from the correct save tape file (label QFILEUPR).

   Note: Use the tapes from the most recent complete save operation (SAVSYS). If a SAVSECDTA command has been run since the last complete save operation, use the SAVSECDTA tapes.

   If the SAVSYS tape is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
   ```

   If the save security data tape (SAVSECDTA) is used, type the following:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

3. Restore the device configuration objects from your most recent SAVSYS tapes. Type the following:

   ```
   RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

4. It is recommended that you save library QRCL and its contents to create a backup copy of the user ASPs containing isolated objects for use in case another failure should occur. This copy may be needed in the following restore steps, depending how you chose to do the restore operation.

   Load a scratch tape, type the following and press the Enter Key.

   ```
   SAVLIB LIB(QRCL) DEV(TAP01) ENDOPT(*UNLOAD)
   ```

5. Restore the objects to the system in **one** of the following ways:

Method 1. Recovery When No Objects Exist in User ASPs or Old Type ASPs Exist

If you had no objects stored in user ASPs or had journals, journal receivers and save files in user ASPs (where the library for the objects are in the system ASP) at the start of this procedure, or if you want to simplify the restore process at the expense of going through restoring objects that are still intact in the user ASPs on your system, perform the following steps:

1. Delete library QRCL by typing the following and pressing the Enter key.

   ```
   DLTLIB LIB(QRCL)
   ```

2. Restore the IBM and user libraries:

   Type the following and press the Enter key,

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01)  ENDOPT(*UNLOAD)
   ```

3. If you have document library objects to restore, load the SAVDLO tape, and then type the following:

   ```
   RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
             SEQNBR(beginning-number ending-number)
   ```

4. Grant all private object authorities that existed when the system was saved by typing the following and pressing the Enter key. No media is required.

   ```
   RSTAUT
   ```

Method 2. Recovery of Objects and Libraries Existing in User ASPs

After the RCLSTG command is run, the addressability of libraries and objects in the user ASP is restored.

1. Restore the individual libraries to the system ASP from your save tapes.

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) OMITLIB(user-ASP-library-name)
   ```

   Do not restore libraries from the SAVLIB tapes that are in user ASPs. Specify the OMITLIB parameter to exclude up to 50 libraries that exist in the user ASPs. Journals should be restored before the journaled files, or journaling is not resumed for the restored files.

2. Display library QRCL to determine if damaged objects were found.

   ```
   DSPLIB LIB(QRCL)
   ```

   If objects are found in QRCL, do the following:

   a. Delete the damaged objects in the user ASP. If the object is a physical file, you must delete the associated logical files first, and then delete the physical file. Do the following:

      1) Display the library in the user ASP:

         ```
         WRKLIB LIB(library-name)
         ```

      2) Type a **12** (Work with objects) in the *Opt* column and press the Enter key.

      3) Find the objects to be deleted in the *Object* column.

      4) Type a **4** (Delete) in the *Opt* column for each object you want to delete.

      5) Press the Enter key.

   b. After the objects in the ASP are deleted, move the objects in library QRCL back to their original library.

      ```
      MOVOBJ OBJ(QRCL/object-name) OBJTYPE(*XXXX) TOLIB(library-name)
      ```

3. The RCLSTG command changed ownership of objects existing in user ASPs to QDFTOWN user profile. Transfer ownership of the objects in the user ASP library from QDFTOWN user profile to the correct user profile.

   a. Type the following and press the Enter key:

      ```
      WRKOBJOWN USRPRF(QDFTOWN)
      ```

      The Work with Objects by Owner display is shown.

   b. On the Work with Objects by Owner display, type a **9** in the *Opt* column for each object in the ASP library that you want to change ownership for.

   c. If all the objects will have the same owner, type the following on the command line of the Work with Objects by Owner display. Otherwise, continue with step 3d.

      ```
      NEWOWN(owner-name)
      ```

      Note: If you enter NEWOWN(owner-name) on the command line of the Work with Object by Owner display, you will not have to enter an owner name in the *New owner* prompt on the Change Object Owner display for each object.

   d. Press the Enter key.

e. On the Change Object Owner (CHGOBJOWN) display, type the name of the new owner in the *New owner* prompt and press the Enter key. Repeat this step for all the objects that need the ownership changed.

4. Load the the SAVDLO tape, and then type the following to restore document library objects:

```
RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
       SEQNBR(beginning-number ending-number)
```

5. Type the following to grant all private object authorities that existed when the system was saved:

```
RSTAUT
```

Removing Units from a Mirrored ASP without Clearing the ASP

Warning: For 9406 model D70 or model D80 that has mirrored protection for the system ASP (ASP 1) and 224MB or more of main storage, must have four 2800-001 storage units configured to ASP 1.

If a mirrored ASP has excess capacity, units can be removed from it without clearing the ASP if the following steps are followed:

- Suspend those storage units you want to remove from the mirrored ASP.
- Stop mirrored protection on that ASP.

The mirrored units are unpaired and one storage unit of each mirrored pair becomes nonconfigured. Mirrored units with one suspended storage unit have the suspended storage unit made nonconfigured.

Although the DST remove function is not used, the nonconfigured storage units are effectively removed from the ASP. At this point, storage units can be added back to the ASP if required. Mirrored protection can be started again with fewer units.

Removing Units from a User ASP that Has Mirrored Protection

When a unit is removed from an ASP that has mirrored protection, both mirrored units of that mirrored pair are removed. When a unit is removed from an ASP, the ASP is cleared unless the unit is not in a checksum set in an ASP that has checksum protection.

Task Overview

You will perform the following steps during this task

1. Save the security data
2. Save the objects in the user ASP
3. Access DST
4. Remove the unit from the user ASP
5. Restore the objects to the source user ASP

Warning: Use caution when removing a unit from the configuration. Removing a unit from an ASP, destroys all objects in the ASP.

> **Before you begin . . .**
>
> The following conditions apply when you remove a unit:
>
> - All data in the ASP from which you are removing the unit is destroyed except when you remove a unit that is not checksum protected from an ASP that is checksum protected.

In the following example, a unit currently in a user ASP is being removed.

Do the following steps to remove a disk unit:

Task 1. Save the Security Data

Use the Save Security Data (SAVSECDTA) command to save the security data.

By entering this command now, you can save all private object authorities. If you do not do this now, you must later manually restore private authorities to every object (with the EDTOBJAUT command) that resides on the user ASP containing the units to be reassigned.

To save the security data, do the following:

1. To change the system operator message queue so all messages will appear on the display, type the following and press the Enter key.

   ```
   CHGMSGQ QSYSOPR *BREAK SEV(60)
   ```

2. Load the first tape, and make the tape device ready.
3. Save the security data:

   If you are saving distribution (mail) objects, type the following and press the Enter key.

   ```
   SAVSECDTA DEV(TAP01) MAIL(*YES)
   ```

   If one or more tapes have not been initialized, a message similar to the following may appear:

   ```
   Volume on device TAP01 wrong type (C INZ R)
   ```

   Enter INZ and press the Enter key to initialize the tape.

4. When the following message appears, load the next tape, make the device ready, and then enter R.

   ```
   Device was not ready or next volume not loaded (C R)
   ```

Task 2. Save the Objects in the User ASP

1. Save all objects contained in the user ASP with the appropriate command:

   ```
   SAVLIB LIB(library-name) DEV(TAP01) LABEL(label-name)
   ```

   Or

   ```
   SAVOBJ OBJ(*ALL) LIB(library-name) DEV(TAP01) OBJTYPE(*ALL)
          VOL(*MOUNTED) ENDOPT(*LEAVE)
   ```

   If the objects you save include one or more journals, you may want to save all database files associated with those journals. This step makes it possible to reestablish journaling by deleting the old files and restoring the saved files.

Task 3. Access DST Options

Sign on to DST using the following steps:

1. Notify the users to sign off the system by sending a break message.
2. Change the QSYSOPR message queue to break mode:

   `CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)`

3. End all subsystems:

   `ENDSBS SBS(*ALL) OPTION(*IMMED)`

   Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.

4. Insert the key into the keylock switch on the control panel.
5. Turn the key to the Manual position.
6. Power down the system:

   `PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)`

7. When the system has powered down, the IPL or Install the System display appears.

```
                          IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system










Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                               © COPYRIGHT IBM CORP. 1980, 1991.
```

8. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

Mirrored Protection

```
                    Dedicated Service Tools (DST) Sign On

 Type choice, press Enter.

   DST password . . . . . . . .   _____
```

9. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

   The Use Dedicated Service Tools (DST) menu is shown.

```
                    Use Dedicated Service Tools (DST)

 Select one of the following:

           1. Perform an IPL
           2. Install the operating system
           3. Work with licensed internal code
           4. Work with disk units
           5. Work with DST environment
           6. Select DST console mode
           7. Start a service tool
           8. Perform automatic installation of the operating system
           9. Work with save storage and restore storage

 Selection
     4

 F3=Exit            F12=Cancel
```

Task 4. Remove the Unit From the User ASP

1. Select option 4 (Work with disk units) on the Use Dedicated Service Tools menu.

```
                         Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information

Selection
     1

F3=Exit        F12=Cancel
```

2. On the Work with Disk Units display, select option 3 (Work with ASP configuration). The following display is shown.

```
                         Work with ASP Configuration

Select one of the following:

     1. Display disk configuration capacity
     2. Create User ASP
     3. Delete user ASP
     4. Add units to existing ASP
     5. Reset overflowed user ASPs
     6. Change ASP storage threshold
     7. Move units from one ASP to another
     8. Remove unit from configuration

Selection
     8

F3=Exit        F12=Cancel
```

3. Select option 8 (Remove unit from configuration) to remove the desired unit.

   Note: If any user ASP has overflowed, reset the ASP before rmoving the unit from the ASP. For more information on how to reset the ASP, see "Resetting an Overflowed User ASP" on page 15-67.

```
                    Remove Unit from Configuration

Type choices, press Enter.

                                               --Protected--  --Unprotected--
 ASP  Unit  Type  Model  Threshold  Overflow    Size  %Used      Size  %Used
  1                         90%     No             0  0.00%      1802 59.74%
        1   9332   400                             0  0.00%       200 97.84%
        2   9332   400                             0  0.00%       200 56.62%
        3   9332   400                             0  0.00%       200 57.70%
        4   9332   400                             0  0.00%       200 57.15%
        5   9332   400                             0  0.00%       200 58.34%
        6   9332   400                             0  0.00%       200 56.12%
        7   9332   200                             0  0.00%       200 58.95%
        8   9332   200                             0  0.00%       200 47.36%
        9   9332   200                             0  0.00%       200 47.59%

 Press Enter to continue.

 F3=Exit   F5=Refresh    F11=Display non-configured units    F12=Cancel
```

4. Select the unit you want removed from the configuration.

   Note: All objects in the source ASP will be destroyed as a result of the remove operation.

   The Confirm Remove of Units display is shown.

```
                         Confirm Remove of Units

Warning: Removing this unit will destroy all the data in the ASP
from which the unit is being removed.

Press Enter to confirm your choice for 1=Move units.
Press F12=Cancel to return and change your choice.

ASP being removed from . . . . . . :    __
Unit to remove . . . . . . . . . . :   ___






F10=Confirm      F12=Cancel
```

5. To confirm the remove of the unit, press F10 (Confirm). Press F12 (Cancel) to return and change your choice.

   All data in the ASP is destroyed unless you are removing a unit that does not have checksum protection from an ASP that has checksum protection.

6. Exit from the Work with Disk Units function.

Task 5. Restore the Objects to the Source User ASP

1. Load the correct volume of the SAVSECDTA tapes.
2. Type the following to restore the user profiles:

```
RSTUSRPRF USRPRF(*ALL) DEV(TAP01)
```

3. Load the correct volume of the RSTLIB or RSTOBJ tapes.
4. Use the appropriate restore command to restore all objects that were in the source ASP:

```
RSTLIB SAVLIB(library-name) DEV(TAP01)
```

   Or:

```
RSTOBJ OBJ(*ALL) SAVLIB(library-name) OBJTYPE(*ALL)
       ENDOPT(*REWIND) MBROPT(*ALL)
```

   If an object resided in a user ASP that no longer exists after the object is saved, the object is not restored unless you specify an existing ASP or the default value of *SAVASP for the RSTASP parameter on the appropriate Restore command.

5. Restore private authorities:

```
RSTAUT
```

   If any objects were restored into a different library than the one from which they were saved, you must manually grant private authority for the restored objects with the GRTOBJAUT command.

   Notes:

   a. If the journals are in the system ASP, create a new receiver in the user ASP, and run the CHGJRN command to attach the new journal.
   b. If the user ASP contained journals and you did not delete the old files and then restore them from the media, create a new journal receiver and the journal in the user ASP. Use the Start Journal Access Paths (STRJRNAP) and the Start Journal Physical Files (STRJRNPF) commands to start journaling again.
   c. If the user ASP contained journals and you did not delete the old files and then restore them from the save media, you must delete and restore the files that were journaled which will start journaling again.

6. This completes the steps to remove a unit.

Stopping Mirrored Protection

When mirrored protection is ended, mirrored units are unpaired and half of the storage units become nonconfigured units. All units in the the ASPs that do not have mirrored protection ended must be present and not suspended before mirrored protection can be ended. To control which mirrored unit of each pair becomes nonconfigured, you may suspend the storage units that you want to become nonconfigured. For units that are not suspended, the selection is automatic. An attempt is made to keep entire disk enclosures either configured or nonconfigured. Mirrored protection can be ended using the DST option before the IPL to the OS/400 licensed program.

To end mirrored protection, do the following:

1. Access DST options.

a. Notify the users to sign off the system by sending a break message.

b. Change the QSYSOPR message queue to break mode:

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
```

c. End all subsystems:

```
ENDSBS SBS(*ALL) OPTION(*IMMED)
```

Wait until a message is sent to the QSYSOPR message queue indicating that all subsystems have ended and the system is in a restricted state.

d. Insert the key into the keylock switch on the control panel.

e. Turn the key to the Manual position.

f. Power down the system:

```
PWRDWNSYS OPTION(*IMMED) RESTART(*YES) IPLSRC(B)
```

g. When the system has powered down, the IPL or Install the System display appears.

```
                          IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools (DST)
          4. Perform automatic installation of the operating system









Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                  © COPYRIGHT IBM CORP. 1980, 1991.
```

h. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key. The Dedicated Service Tools (DST) Sign On display is shown.

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . . . .   ______
```

i. Sign on DST with the DST *security* or *full* level password. *Security Concepts and Planning*, SC41-8083, has more information about DST passwords.

The Use Dedicated Service Tools (DST) menu is shown.

```
                         Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic installation of the operating system
          9. Work with save storage and restore storage




Selection
    4

F3=Exit            F12=Cancel
```

2. Select option 4 (Work with disk units) on the Use dedicated Service Tools menu and press the Enter key.

```
                             Work with Disk Units

Select one of the following:

     1. Work with disk configuration
     2. Analyze disk problem
     3. Work with disk recovery
     4. Work with disk unit information




Selection
    1

F3=Exit          F12=Cancel
```

3. Select option 1 (Work with disk configuration) on the Work with Disk Units display and press the Entry key.

Mirrored Protection

```
                         Work with Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Work with ASP threshold
     3. Work with ASP configuration
     4. Work with checksum protection
     5. Work with mirrored protection

Selection
     5

F3=Exit         F12=Cancel
```

4. Select option 5 (Work with mirrored protection) on the Work with Disk Configuration display and press the Enter key.

```
                      Work with Mirrored Protection

Select one of the following:

     1. Display disk configuration
     2. Start mirrored protection
     3. Stop mirrored protection
     4. Calculated mirrored capacity

Selection
     3

F3=Exit         F12=Cancel
```

5. Select option 3 (Stop mirrored protection) on the Work with Mirror Protection display.

```
                    Select ASP to Stop Mirrored Protection

 Type options, press Enter
   1=Select

Option   ASP   Protection
          1    Mirrored
          2    Mirrored










 F3=Exit         F12=Cancel
```

6. Select the ASP or ASPs for which mirrored protection is to be stopped on the Select ASP to Stop Mirrored Protection display and press the Enter key.

7. The Confirm Stop Mirrored Protection display is shown. Press the Enter key to confirm your choice.

```
                      Confirm Stop Mirrored Protection

 Press Enter to confirm your choice to stop mirrored
   protection.  During this process the system will be IPLed.
   You will return to the DST main menu after the IPL is
   complete.  The system will have the displayed protection.
 Press F12 to return to change your choice.


             Serial
 ASP  Unit   Number      Type  Model  Address         Protection      CSS
  1                                                   Unprotected
         1   57-1253342  9335   B01   0010-0101FFFF   Unprotected
         3   57-1225682  9335   B01   0020-0201FFFF   Unprotected
         4   57-1107344  9335   B01   0010-0401FFFF   Unprotected
  2                                                   Mirrored
         5   10-00A5703  9332   400   0010-0101FFFF   Active
         5   10-00A4293  9332   400   0020-0201FFFF   Active


                                                                   More...
 F12=Cancel
```

Figure 22-2. Confirm Stop Mirrored Protection

Mirrored Protection Recovery Actions

In considering aspects of recovery, you need to distinguish between *errors* and *failures* in the disk subsystem.

A disk *error* refers to an unexpected event during an I/O operation which can cause the loss or corruption of the data being transferred. Most disk errors are hardware related, caused by a failure in some part of the component chain from the I/O processor to the disk surface. Environmental effects such as power abnormalities or severe electrostatic discharges can also cause disk errors. Included in the definition of disk errors is a failure of the licensed internal code that controls the disk subsystem.

When the system detects an error, generally the occurrence is logged and the operation is attempted again. Temporary errors are those from which the system can recover and complete the I/O operation successfully. When the error is so severe that the I/O operation cannot succeed, it is a permanent error.

When the system detects a permanent error, it classifies it as a *failure* in that hardware subsystem. In an ASP that does not have mirrored protection, a failure causes the system to become unusable.

On a system with mirrored protection, errors and failures have different effects.

In "Level of Protection" on page 19-3, the concept of level of protection was introduced. When a failure occurs on a system with mirrored protection, the recovery procedure is affected by the level of protection configured. Because of the hardware differences between the 9406 and the other system units, they will be discussed separately. There are variations between the 9402 or 9404 system unit and the 9406 system unit in the handling of the IPL process.

9402 and 9404 System Units

The following considerations apply to 9402 and 9404 system units:

- Disk units cannot be repaired when the system is powered on. If spare units are available, the SST Replace function may be used to replace a failed disk unit with a spare while the system is running.
- There is a special consideration for unit 1 if it fails. Rule 6 on page 21-1 has more information about restrictions for unit 1.

  Note: This restriction does not apply for the type 2615 service processor on the 9404 system unit.

9406 System Units

The following considerations apply to 9406 system units:

- Bus-level protection is possible only on AS/400 models with two or three busses. Because system hardware requires that unit 1 be on bus 0, the two storage units of the mirrored pair for unit 1 must be on the same bus. Therefore, bus-level protection is not possible for unit 1. A failure of bus 0 always results in a system failure.
- When a bus fails, a disk unit that is mirrored on another bus continues to operate. However, it must be understood that a bus failure can cause other subsystems to fail. A failure of active work stations or communications lines limits the availability provided by mirrored protection. The system must be powered down to repair a bus.
- When an I/O processor fails, there are two possibilities. If there are any configured units attached to the I/O processor that have mirrored protection under the same I/O processor or that are not mirrored, then the system becomes unusable. If the remaining half of each mirrored pair is active, the system suspends all mirrored units under the failed I/O processor and continues operation. A failed I/O processor can be replaced only when the

system power is off. After the failed I/O processor is replaced, the system synchronizes the suspended mirrored units during the next IPL.

- It is possible that disk unit temporary errors will be logged with increasing frequency well before the time of a failure. When a disk unit shows these symptoms, you may decide to repair it. If the disk unit contains multiple storage units, all storage units in the disk unit must be suspended before the disk unit can be repaired.
- The 9332 model 400, 9332 model 600, and the 9335 model B01 disk units on the 9406 system unit have two storage units in a single disk unit. It is possible for only one of the storage units in a disk unit to fail. It is recommended that the other storage unit be suspended before the disk unit is powered down.
- During some parts of the IPL process, the system cannot use the mirrored unit for unit 1. If unit 1 fails during this time, the IPL is tried again from the mirrored unit for unit 1. If this is unsuccessful, one of the units in the mirrored pair may be manually powered down or physically removed, and an IPL may be tried again to circumvent the problem.

  Note: The 2800 disk units do not have power off switches. You may be able to switch the book adapter cards.

Suspending or Resuming Mirrored Units

If you have to suspend or resume a mirrored unit, you can do so using the Suspend/Resume Mirrored Protection option on the Work With Disk Unit Recovery display using SST or DST.

There can be more than one unit with the status of *suspended* or *resuming status* on the system at the same time. However, only one unit at a time can have its status changed. If you suspended both units in a 9332 model 400 disk unit, both storage units must be resumed before full protection is started. For a unit to be resumed, it must be selected individually. However, you do not need to wait until the status of the unit is *active* before selecting the next unit.

1. Type the following to suspend or resume mirrored protection using SST options:

   ```
   STRSST
   ```

2. Select option 3 (Work with disk units) on the System Service Tools (SST) menu and press the Enter key.

Mirrored Protection

```
                         Work with Disk Units

Select one of the following:

     1. Display disk configuration
     2. Display checksum configuration
     3. Calculate checksum configuration
     4. Work with ASP threshold
     5. Work with disk unit recovery
     6. Work with disk unit information
     7. Calculate mirrored capacity






Selection
     5

F3=Exit         F12=Cancel
```

3. Select option 5 (Work with disk unit recovery) on the Work with Disk Units display and press the Enter key.

```
               Work with Disk Unit Recovery

Select one of the following:

     1. Replace configured unit
     2. Disk problem recovery procedures
     3. Suspend/resume mirrored protection
     4. Delete disk unit data












Selection
      3

F3=Exit      F11=Display configuration status    F12=Cancel
```

4. Select option 3 (Suspend/resume mirrored protection) on the Work with Disk Unit Recovery display and press the Enter key.

```
                    Suspend/Resume Mirrored Protection

Type option, press Enter.
  1=Suspend Protection        2=Resume Protection

                      Serial
Option  Unit  ASP     Number       Type  Model  Address          Status
  _       1    1      57-00B57BE   9335   B01   0020-0001FFFF    Active
  _       1    1      57-00B5CF7   9335   B01   0020-0100FFFF    Active
  _       2    1      57-00B8BB7   9335   B01   0120-0002FFFF    Active
  _       2    1      57-00B6F1D   9335   B01   0110-0100FFFF    Active









F3=Exit        F5=Refresh        F12=Cancel
```

5. Type a *1* (Suspend Protection) or a *2* (Resume Protection) in the *Option* column for each unit that you want to suspend or resume mirrored protection. You can suspend protection only on units that have both units in an *Active* or *Resuming* status. If one of the units is in *Resuming* status, then it is the only unit that can be selected to suspend. You can select only a unit in *Suspending* status to resume.

Replacing a Mirrored Unit

A unit selected to replace the failed mirrored unit must satisfy all of the mirrored protection configuration rules and restrictions when it is paired with the remaining unit in the mirrored pair. (See "Mirrored Protection Configuration Rules" on page 21-1.)

You can replace mirrored units using the Replace Disk Unit option in either DST or SST. To do this, you need to have a spare storage unit that can be paired with the mirrored unit of the storage unit being replaced. The storage unit being replaced can have a status of active or suspended. However, one of the storage units in the pair must be suspended. The results of the replace operation is different for each status. Replacing a suspended storage unit causes that storage unit to go to a status of resuming after the replace operation. Replacing an active unit causes the ASP to be cleared. The storage unit being replaced can also be missing or not missing. To replace a unit with a status of resuming, you must suspend it. If the status of unit 1 is unknown, replace operations are not allowed until the status of the mirrored units for unit 1 is known. The unit selected to replace another mirrored unit must satisfy all of the mirrored protection configuration rules and restrictions when it is paired with the remaining unit in the mirrored pair. (See "Mirrored Protection Configuration Rules" on page 21-1.)

If a storage unit fails, and if the same storage unit that failed has been repaired, it is not necessary to replace it. The failed disk should have a status of suspended.

If the storage unit being replaced is active, it can only be replaced at DST before the IPL to the OS/400 licensed program. It should never be necessary to replace an active unit unless both units of the mirrored pair have failed. If this situation does occur, the service representative should first try to recover the data from the failed units using the Save Disk Unit Data option on the Work with Disk Unit Recovery display. When an active unit is replaced, the last good copy of the data has been lost and a warning display is shown. If the replacement continues, the ASP is cleared as if the replace option was used on an unprotected ASP.

Replacing unit 1 requires special handling. If the system ASP has mirrored protection, one of the units in the mirrored pair for unit 1 is selected as the IPL device. It is the only unit that is used until the system performs an IPL to the OS/400 licensed program. Until then, it cannot be replaced or even suspended. However, its mirrored unit can be both suspended and replaced. After the IPL to the OS/400 licensed program, the IPL device can be suspended and then replaced.

Replacing a unit may cause the level of protection for a mirrored pair to change. If a lower level of protection results from a replacement operation, a warning screen is displayed. At certain times, especially when missing units are involved in the replace operation, the system may not be able to accurately calculate the level of protection and the same warning display is shown.

To replace a disk unit using SST, do the following:

1. Type the following to access SST:

   STRSST

2. Select option 3 (Work with disk units) on the System Service Tools (SST) menu and press the Enter key.

```
                         Work with Disk Units

 Select one of the following:

      1. Display disk configuration
      2. Display checksum configuration
      3. Calculate checksum configuration
      4. Work with ASP threshold
      5. Work with disk unit recovery
      6. Work with disk unit information
      7. Calculate mirrored capacity





 Selection
      5

 F3=Exit         F12=Cancel
```

3. Select option 5 (Work with disk unit recovery) on the Work with Disk Units display and press the Enter key.

```
                Work with Disk Unit Recovery

Select one of the following:

     1. Replace configured unit
     2. Disk problem recovery procedures
     3. Suspend/Resume mirrored protection
     4. Delete disk unit data

Selection
     1

F3=Exit      F11=Display configuration status   F12=Cancel
```

4. Select option 1 (Replace configured unit) on the Work with Disk Unit Recovery display and press the Enter key.

   The Select Configured Unit to Replace display is shown.

```
                   Select Configured Unit to Replace

Type option, press Enter.
  1=Select

                     Serial
Option  Unit  ASP  Number      Type  Model  Address           Status
  1       3    2   10-0122875  6100   015   0110-0300FFFF     Active
          3    2   10-0123972  6100   015   0110-0200FFFF     Suspended
```

5. Type a 1 in the *Option* column on the Select Configured Unit to Replace display and press the Enter key.

Mirrored Protection

```
                     Select Configured Unit to Replace

           Serial
Unit  ASP  Number      Type  Model  Address          Status
 3     2   10-0122875  6100   015   0110-0300FFFF    Suspended
 Type option, press Enter.
   1=Select

       Serial
OPT  Number      Type  Model  Address         Status
  1  10-0124597  6100   015   0110-0301FFFF   Non-configured
     10-0126894  6100   015   0110-0303FFFF   Non-configured
```

6. Type a 1 in the *Option* column on the Select Configured Unit to Replace display and press the Enter key.

```
                     Confirm Replace of Configured Unit

Warning: Replacing a unit will destroy all the data from the
ASP that contained the unit being replaced.

ASP that will have its data destroyed . . . . . . .       2

Press F10 to confirm your choice for 1-Replace
Press F12 to return and change your choice.

                     Serial
 OPT  Unit  ASP  Number      Type  Model  Address         Status
   1   3     2   10-0124597  6100   015   0110-0301FFFF   Active










 F10=Confirm       F12=Cancel
```

7. Press F10 (Confirm) or F12 (Cancel).

8. The replacement function runs for several minutes. Wait until the replacement function completes.

Using Spare Nonconfigured Units for Replacement

If mirrored units become suspended as a result of a hardware failure, the system continues to run. However, one or more storage units will be suspended and therefore unprotected until your service representative can repair or replace the failed hardware. You may be able to resume mirrored protection before the repair actions are done if you have spare nonconfigured units.

Call your service representative. You will be directed to run problem analysis on those units that are suspended. You can determine what units are suspended by using the Display Disk Configuration Status option using SST or the Work with Disk Status (WRKDSKSTS) command. If all disk units under an I/O processor are suspended, the I/O processor probably has failed. If you have enough spare units of the right type and model, and if the spare units are not on the I/O processor that has failed, you may be able to use the spare nonconfigured units to resume mirrored protection.

After your service representative repairs a failed storage unit, you may want to use it instead of the spare to restore the previous level of protection. To use the repaired unit, do the following:

1. Suspend the active storage unit that was previously used as the spare by typing the following on a command line and pressing the Enter key.

   ```
   STRSST
   ```

   The System Service Tools (SST) menu is shown.

2. Select option 3 (Work with disk units) on the System Service Tools menu and press the Enter key.

```
                          Work with Disk Units

Select one of the following:

     1. Display disk configuration
     2. Display checksum configuration
     3. Calculate checksum configuration
     4. Work with ASP threshold
     5. Work with disk unit recovery
     6. Work with disk unit information
     7. Calculate mirrored capacity






Selection
     5

F3=Exit         F12=Cancel
```

3. Select option 5 (Work with disk unit recovery).

Mirrored Protection

```
                  Work with Disk Unit Recovery

 Select one of the following:

      1. Replace configured unit
      2. Disk problem recovery procedures
      3. Suspend/resume mirrored protection
      4. Delete disk unit data











 Selection
      3

 F3=Exit      F11=Display configuration status   F12=Cancel
```

4. Select the option 3 (Suspend/resume mirrored protection).

```
                     Suspend/Resume Mirrored Protection

 Type option, press Enter.
   1=Suspend Protection        2=Resume Protection

                      Serial
 Option  Unit  ASP    Number      Type  Model  Address        Status
   _       1    1     57-00B57BE  9335   B01   0020-0001FFFF  Active
   _       1    1     57-00B5CF7  9335   B01   0020-0100FFFF  Active
   1       2    1     57-00B8BB7  9335   B01   0120-0002FFFF  Active
   _       2    1     57-00B6F1D  9335   B01   0110-0100FFFF  Active









 F3=Exit        F5=Refresh        F12=Cancel
```

5. Type a *1* (Suspend Protection) in the *Option* column. The original spare unit is the same disk type and model as the repaired disk unit.
6. Return to the Work with Disk Unit Recovery display by pressing F12 (Cancel)

```
                    Work with Disk Unit Recovery

 Select one of the following:

      1. Replace configured unit
      2. Disk problem recovery procedures
      3. Suspend/Resume mirrored protection
      4. Delete disk unit data











 Selection
      1

 F3=Exit      F11=Display configuration status    F12=Cancel
```

7. Select option 1 (Replace configured unit).

```
                         Select Configured Unit to Replace

 Type options, press Enter.
   1=Select

                    Serial
 Option  Unit  ASP  Number       Type  Model  Address            Status
           3     2  10-0122875   6100   015   0110-0300FFFF      Suspended
           3     2  10-0123972   6100   015   0110-0200FFFF      Active
```

8. Type a 1 in the *Option* column on the Select Configured Unit to Replace display and press the Enter key.

```
                         Select Configured Unit to Replace

           Serial
 Unit  ASP  Number      Type  Model  Address           Status
  3     2   10-0122875  6100   015   0110-0300FFFF     Suspended
  Type option, press Enter.
    1=Select

                  Serial
 Option  Unit  ASP  Number      Type  Model  Address           Status
   1      10-0124597  6100   015    0110-0301FFFF     Non-configured
          10-0126894  6100   015    0110-0303FFFF     Non-configured
```

9. The replacement function runs for several minutes. Wait until the replacement function completes.

Mirrored Protection Recovery Actions Performed by the Service Representative

The procedures described here are overviews of the steps and considerations involved in disk unit repair in the mirrored environment. Although these steps are performed by your service representative, they are included here for your information.

The problem analysis procedure suspends mirrored protection on any storage units required to perform service on the reported failure. After the repair action, the problem analysis procedure resumes the suspended storage units.

9402 and 9404 System Units:

1. The system must be powered down to repair the failing storage unit.
2. If unit 1 at the IPL address has failed, see 6 on page 21-1 for restrictions that apply.
3. An attended IPL of the system is performed to bring up the DST menu.
4. If the Replace configured unit option is necessary, the new storage unit is formatted and initialized.
5. If the disk configuration status is displayed, the failed disk unit is in suspended status.
6. The IPL completes to normal operations.

9406 System Units:

1. If there is a working storage unit in the disk unit to be repaired, it is suspended by the PAR procedure.
2. Power down the disk unit.
3. Repair the failing unit.

Other Recovery Considerations for Mirrored Protection

Message Handling: When a system with mirrored protection experiences a disk failure, the only external indication of the failure is a message sent to the system operator message queue (QSYSOPR). If there is a message queue named QSYSMSG in the QSYS library, the messages are sent there also.

When suspended units exist, the system sends a message every hour to the QSYSOPR message queue as a reminder.

You should have a method of bringing these messages to the attention of the system administrator. If the interactive job at the console allocates the QSYSMSG message queue and places it in break mode, you are notified of any problems. For more information on QSYSMSG, see the *CL Programmer's Guide*.

Synchronization: When the system is synchronizing (resuming) a disk unit, the system response time becomes longer.

When mirrored protection is resumed on a suspended disk unit at DST, the synchronization is done during the IPL to the OS/400 licensed program.

Mirrored Protection Disk-Error Handling

Mirrored protection handles disk errors as follows:

Unrecoverable device error:

1. The system suspends the failing storage unit and mirrored protection is suspended for the mirrored pair.
2. The system continues operations using the operating storage unit of the mirrored pair.
3. A message is sent to the QSYSOPR message queue identifying the failing storage unit and informing you that mirrored protection is suspended for the mirrored pair.

Permanent read error:

1. The system reads from the other storage unit of the mirrored pair. If the permanent read error occurs on the other storage unit as well, the original read request completes with a permanent read error.
2. If the read operation from the other storage unit is successful, the data is written back to the first unit of the mirrored pair, assigning an alternate sector. Only then does the system signal that the original read request is complete.

Not operational storage unit:

1. The system attempts recovery. If it is successful, normal system operations continue with mirrored protection and without suspending or synchronizing the unit. The 0244 attention SRC is displayed in the control panel.
2. If recovery has not succeeded within a time limit the unit is considered to have an unrecoverable device error which is processed as described previously.

Timeout

1. The system attempts to recover from the timeout. If it is successful, normal system operations continue with mirrored protection and without suspending or synchronizing this unit.
2. If recovery is unsuccessful, the storage unit is considered to have an unrecoverable device error which is processed as described previously.

I/O processor or bus failure

1. The system suspends each disk unit attached to the failing I/O processor or bus in the same way it is done for an unrecoverable error.
2. The system saves a copy of the failing I/O processor's storage so the problem can be diagnosed. The system continues without the failing I/O processor.

Disk-related failure of unit 1 before the IPL to the Operating System/400: See 6 on page 21-1 for restrictions that apply.

Missing Disk Units

If a disk unit, a controller, or an I/O processor fails during an IPL, the system detects the failure and does one of the following:

- Displays an SRC on the control panel if the keylock switch is not in the Manual position.
- Displays the Missing Disk Unit display on the console if the keylock switch is in the Manual position.

If the failing unit has mirrored protection and its mirrored unit is active, the following display is shown.

```
                          Suspend Missing Disk Units

 The following disk units are missing from the system disk
 configuration:

                                  Serial
 ASP   Unit  Type  Model  Number       Address         Reference Code
 _2_   _3_   9332  400_   10-234233    0020-0201FFFF      1713

                                                                  More . . .

 Press Enter to suspend missing disk units and continue the IPL.

 F6=Use Dedicated Service Tools (DST)
 F11=Display Unit Status
```

You can suspend mirrored protection on the affected units and continue the IPL. An entry is written in the problem log. You can run on-line problem analysis on the failing unit at a later time. The *type* and *reference code* fields can be used with the unit reference code guide to determine the cause of the problem. If the keylock switch is not in the Manual position, a system reference code is displayed on the control panel. The system automatically suspends mirrored protection on the affected units and continues the IPL.

Saving a Unit

The system allows you to save data from storage units using the DST Save Disk Unit Data option.

The following rules apply to saving units on a system with mirrored protection:

- Only configured units can be saved.
- The save operation is not allowed when both mirrored units of a mirrored pair are active. Only one of the mirrored units can be saved. Therefore, one of them must be suspended.
- Only the active unit of a mirrored pair can be saved because the active unit contains the current data.

- If multiple failures cause the state of unit 1 to be unknown, saving of any storage unit is not allowed.

Restoring a Unit

In the mirrored environment, the system allows you to restore data to storage units.

The following rules apply to restoring units on a system with mirrored protection:

- The restore is possible only for an active device.
- This option can restore either a configured or nonconfigured disk unit.
- The restore operation requires that the unit restored to is the same type as the unit saved.
- The restore operation is not permitted if the state of a unit is unknown. You can restore unit 1 only to the IPL device.
- After restoring unit 1, the system performs an IPL to DST.
- The unit being restored must satisfy all mirrored protection configuration rules and restrictions.

Mirrored Protection for Unit 1 On the 9404 and 9402 System Units

If unit 1 on the 9402 or 9404 System Unit fails during the IPL, and the service processor is not type 2615, the system does not recover automatically. If a system reference code indicates that unit 1 at address 0010-0001FFFF has failed, the service representative may exchange the disk units in the mirrored pair for unit 1. No data recovery actions are required if the exchange is successful.

Note: For systems without service processor card 2615, you should not suspend mirrored protection on unit 1 at location 1 (address 0010-0001FFFF) for system units 9404 and 9402. The system will not be able to perform an IPL past DST. If unit 1 at location 1 is still on, the system will IPL from it and will give a *Back level load source* error. The service representative must then exchange the units in the mirrored pair for unit 1.

Active Mirrored Load Source Failure

If unit 1 has mirrored protection and one of the mirrored units fails, the system determines if the remaining mirrored unit has the correct system records and licensed internal code. If not, the following display is shown when the keylock switch is in the Manual position.

```
                    Display Load Source Failure

The system could not use the load source disk unit that
contains correct data.  The following disk unit contains the
correct data:

Disk unit:
  Type . . . . . . . . . . . . . . . . . . :   9335
  Model  . . . . . . . . . . . . . . . . . :   B01
  Serial number  . . . . . . . . . . . . . :   57-1020449
  Address  . . . . . . . . . . . . . . . . :   0010-0000FFFF

Press Enter to use Dedicated Service Tools (DST).




F11=Display reference codes
```

If the keylock switch is not in the Manual position, a system reference code is displayed on the control panel. The failing unit must be repaired before the system can be used.

If the system can locate a nonconfigured disk unit that matches the missing disk unit, the following display is shown.

```
                    Assign Missing Load Source Unit

The system could not locate one disk unit of the load source
mirrored pair.  The following disk unit is the nonconfigured
disk unit located at that address:

Disk unit:
    Type . . . . . . . . . . . . . . . . . :   9335
    Model  . . . . . . . . . . . . . . . . :   B01
    Serial number  . . . . . . . . . . . . :   57-1020449
    Address  . . . . . . . . . . . . . . . :   0010-0000FFFF

Press Enter to continue the IPL and configure the displayed
  disk unit as the mirrored pair of the load source disk unit.




F6=Use Dedicated Service Tools (DST)
F11=Display reference codes
```

You can accept the nonconfigured unit as the missing unit and continue the IPL.

Unknown Unit 1 Status

If both the service processor and one unit of the mirrored pair for unit 1 have failed, the following display is shown.

```
                    Display Unknown Mirrored Load Source Status

The system can not determine which disk unit of the load
source mirrored pair contains the correct level of data.
The following disk unit is not available:

Disk unit:
   Type . . . . . . . . . . . . . . . . . . :   9335
   Model  . . . . . . . . . . . . . . . . . :   B01
   Serial number  . . . . . . . . . . . . . :   57-1020449
   Address  . . . . . . . . . . . . . . . . :   0010-0000FFFF

Press Enter to use Dedicated Service Tools (DST).






F11=Display reference codes
```

If the keylock switch is not in the Manual position, a system reference code is displayed on the control panel. The missing unit must be repaired. If the missing unit cannot be repaired, you must install the licensed internal code and restore the entire system.

If the system can locate a nonconfigured disk unit that matches the missing disk unit, the following display is shown.

```
                    Assign Missing Load Source Unit

The system could not locate one disk unit of the load source
mirrored pair.  The following disk unit is the nonconfigured
disk unit located at that address:

Disk unit:
  Type . . . . . . . . . . . . . . . . :  9335
  Model  . . . . . . . . . . . . . . . :  B01_
  Serial Number  . . . . . . . . . . . :  57-1020449
  Address  . . . . . . . . . . . . . . :  0010-0100FFFF

Press Enter to continue the IPL and configure the displayed
  disk unit as the mirrored pair of the load source disk unit.





F6=Use Dedicated Service Tools (DST)
F11=Display reference codes
```

Mirrored Protection

You can accept the nonconfigured unit as the missing unit and continue the IPL.

Display Incorrect Licensed Internal Code Install

When the licensed internal code is restored on a mirrored unit for unit 1, one of the mirrored units may have the incorrect level of data stored on it. If this condition occurs, and the disk unit containing the correct data is not available, the licensed internal code is restored to the disk unit with the incorrect data. When an IPL is performed from disk and the correct disk unit is available, the following display is shown. If the keylock switch is not in the Manual position, system reference code (SRC) is shown on the control panel.

```
                  Display Incorrect Licensed Internal Code Install

Licensed internal code has been installed on the incorrect
disk unit of the load source mirrored pair.
If you continue the IPL, the previously installed licensed internal
code installed on the incorrect disk unit of the mirrored
load source pair will be deleted.  The licensed internal code
will be replaced by the licensed internal code from the correct
disk unit.  The following disk unit is the correct disk unit.

Disk unit:
  Type . . . . . . . . . . . . . . . . :     ____
  Model  . . . . . . . . . . . . . . . :     ____
  Serial number. . . . . . . . . . . . :     __-_______
  Address  . . . . . . . . . . . . . . :     ____-_______




Press Enter to continue the IPL and replace the licensed internal code.

F6=Use Dedicated Service Tools (DST)
```

Chapter 23. Performance Considerations for Mirrored Protection

This chapter discusses the effects of mirrored protection on system performance. The normal run-time performance is looked at, as well as the effect of synchronization on system operation. We also look at some ways to improve performance in the mirrored environment.

Run-Time Performance for Mirrored Protection

A general performance expectation for normal operation on a system with mirroring is:

- ***Processing unit***: Mirrored protection places a minimal demand on the processing unit. Mirrored protection uses the processing unit mainly for flagging and look-up operations, and some calculations, such as sector address compares, stripe computations, and page allocation. Processing unit use can be expected to increase by about 1%.
- ***Disk***:
  - Read performance:

    When storage units are busy, the ability to read from either mirrored unit leads to a performance improvement in read-intensive jobs. If storage units are less than 20% busy, then there is no benefit from the balancing effect of mirrored-read operations.
  - Write performance:

    When storage units are busy, write performance decreases in the mirrored environment because every write operation causes two mirrored-write operations. Jobs that characteristically perform large numbers of database I/O operations experience this decrease in performance. Jobs in this category are:
    - batch update jobs
    - jobs using journaling and commitment control
    - large restore operations
    - save operations to save files
    - rebuild of large access paths

Synchronization Effects

When a mirrored pair is synchronized, the synchronization process affects other jobs running on the system. Synchronization is a licensed internal code task that runs in the machine pool. Some of the major factors that affect job performance during synchronization are:

- Use of the disk subsystem being synchronized
- I/O activity to the disk being synchronized; that is mainly write operations
- Available main storage
- The number of disks being synchronized simultaneously

Some suggestions to minimize the effect of synchronization are:

- If you choose to resume mirrored protection using DST, the synchronization process occurs during the IPL to the OS/400 licensed program. The system must complete the IPL before you can use it. Mirroring can be resumed while the system is operational.
- There are two approaches to multiple disk synchronization. You can resume mirrored protection on all mirrored pairs at the same time, or you can resume mirrored protection to one mirrored pair at a time.

  The first approach returns the system to a fully protected state faster, but can significantly impact other jobs while the synchronization takes place. High-disk utilization also slows the synchronization process.

  The second approach affects other jobs on the system to a lesser extent, while extending the exposure window (the time during which there are mirrored pairs on the system that are unprotected).

  It is not possible to recommend one method over the other, as priorities vary from system to system. An awareness of the factors involved will help you make a decision.
- The length of time a mirrored unit is suspended has no impact to the length of the synchronization process. The moment a storage unit is suspended (either manually

or by the system) in response to a disk failure, it must be fully synchronized.

- An uninterruptible power supply enables the system to copy main storage during a power failure, and reduces the chance of a long recovery on the following IPL. If there is an incomplete copy of main storage, the system synchronizes parts of the mirrored pairs. This makes the recovery process longer.

Additional IPL Time after an Abnormal System End

In some cases, if the system ends abnormally, the system cannot be sure that the last changes were written to both units of each mirrored pair. In this case, the system synchronizes parts of the mirrored pairs. The synchronization occurs during the IPL following an abnormal system end. If the system can save a copy of main storage before it ends, this process takes just a few minutes. If not, the synchronization process can take much longer.

Chapter 24. Mirrored Protection Considerations

This chapter provides information about the following:

- Abbreviations
- Using DST and SST
- Capacity planning tools
- Reconfiguring your system
- Examples of mirrored protection configuration

Abbreviations

ABBREVIATION	MEANING
ASP	Auxiliary storage pool
DST	Dedicated service tools
GSP400	General system predictor
I/O	Input/output
IOP	Input/output processor
IPL	Initial program load
KB	Kilobytes
MB	Megabyte
SST	System service tools
2800	Disk unit for the AS/400 9406 system unit. It has two actuators and a storage capacity of 640MB.
6100	Disk unit for the AS/400 9404 system unit. It has one actuator and a storage capacity of 315MB.
6102	Disk unit for the AS/400 9402 system unit. It has one actuator and a storage capacity of 320MB.
6103 G102	Disk unit for the AS/400 9402 system unit. It has one actuator and a storage capacity of 400MB.
6105	Disk unit for the AS/400 9404 system unit. It has one actuator and a storage capacity of 320MB.
6107	Disk unit for the AS/400 9404 system unit. It has one or two actuators and a storage capacity of 400MB or 800MB.
6110	Magnetic storage device controller IOP
6111	Magnetic storage device controller IOP
9332-200	Disk unit for the System/3X systems. It has one actuator and a storage capacity of 200MB. It can be migrated to the AS/400 9406 system unit.
9332-400	Disk unit for the AS/400 9406 system unit. It has two actuators and a storage capacity of 400MB.
9335-A01	Device function controller for the 9335-B01 disk units
9335-B01	Disk unit for the AS/400 9406 system unit. It has two actuators and a storage capacity of 856MB.
9336	Disk unit for the AS/400 9406 system unit. It has two to four actuators and a storage capacity between 942 and 3428MB.

Using DST and SST for Mirrored Protection Management

Mirrored protection management is done by means of System Service Tools (SST) and Dedicated Service Tools (DST). The SST is used while the system is up and running; the DST implies a dedicated system.

Table 24-1. Disk Management

Operation	DST	SST
Display disk configuration	X	X
Change disk configuration	X	
Display ASP configuration	X	X
Configure ASP	X	
Display mirrored protection	X	X

Table 24-2. Mirrored Protection Management

Operation	DST	SST
Calculate mirrored configuration	X	X
Starting mirrored protection	X	
Stopping mirrored protection	X	
Suspend mirrored protection	X	X
Resume mirrored protection	X	X
Add disk units	X	
Move disk units	X	
Replace disk units	X	X
Assign disk units	X	
Save disk units	X	
Restore disk units	X	

Protection

Capacity Planning Tools

- AS/400 Performance Tools

  Given normal growth within a business and the ever-increasing application demands being put on the data processing area of a business, users need to monitor changing system work loads and plan for future system requirements. The AS/400 Performance Tools licensed program provides a set of easy-to-use tools that can assist with these tasks.

  A capacity planning function is available to help determine future system requirements. The capacity planner incorporates a performance model, an evaluator, and a high-level configurator that assists the user in configuring a balanced system that meets resource use guidelines as well as optional response time and throughput objectives. Users may override the evaluator recommendations if they desire to analyze any system configuration. The ability to base projections on measured data is another feature of the planner.

  The capacity planning function is integrated into the data collection and reporting support. Electronic input of the data needed by the capacity planner allows the user to more easily do capacity planning and also allows for increased accuracy by projecting future requirements based on actual measured data. The capacity planner also includes a set of predefined work loads which the user can use to represent applications which are not currently running on the system. The measured data can be combined with the predefined work loads to show the effect of new applications on the current workload.

- General Systems Predictor (GSP400)

  A General System Predictor (GSP400) program is available on A/FE HONE to assist marketing representatives and systems engineers in evaluating customer performance requirements. GSP400 contains the same support that the capacity planner portion of the IBM AS/400 Performance Tools has. Although there is no automatic measurement interface, measurement data can be manually entered.

  GSP400 allows you to define a variety of work loads such as:

  - RAMP-C
  - AS/400 Office
  - Various batch work loads
  - Spool
  - Measured data

  GSP400 allows you to analyze the following system considerations:

  - PURGE(*YES) versus PURGE(*NO)
  - Checksum protection
  - Mirrored protection
  - Local versus remote
  - Changes in the configuration

  GSP400 gives a detailed report of the analysis which includes:

  - Response times
  - Throughputs
  - Device use
  - Pool size recommendations
  - Activity level recommendations
  - Growth analysis
  - Evaluation of the configuration

- AS/400 Quicksizer

  AS/400 model selection can be controlled for sizing purposes, and checksum protection, mirrored protection, and journaling. DDM can be included in sizing estimates. Quicksizer takes your objectives (numbers of user, throughput, growth rate, disk space required) and predicts the process, or main storage unit, and disk units required, and allows a quick analysis of your requirements.

  Quicksizer is a simple analytic model that is used by marketing representatives and support people to predict processor unit and disk use, but not response time. A response time estimate is given, but only to give you an feeling for the response time you could expect.

  AS/400 Quicksizer is available in both VM and DOS versions.

Reconfiguring Your System

To get your desired level of mirrored protection as determined in "Step 4: Determining the Level of Protection" on page 20-5, you may need to reconfigure your existing hardware or install additional hardware. Your IBM marketing representative will help you plan for any hardware configuration changes necessary to configure your system according to standard system configuration rules.

Determining Your Current Hardware Configuration

For an existing system, you can use the Work With Hardware Resources command to determine your current hardware configuration. From the displays, you can see the following:

- What disk units are attached to each controller (disk storage controller).
- What I/O processor (storage controller) each disk unit and controller is attached to.
- What bus each storage controller is attached to.
- Where each device or card is located - rack ID, EIA location, and card slot. (This information should have been entered into the system when the hardware was installed or changed.)

You need to consider separately each ASP that is to be mirrored.

Look at the Display Disk Configuration Status display using SST to determine what disk units are assigned to each ASP.

1. Enter the STRSST command to start the system service tools.
2. Select the Work with disk units option.
3. Select the Display disk configuration option.
4. Select the Display disk configuration status option. Notice which disk units are assigned to each ASP. If a printer is available, it is helpful to print the display of your disk configuration.
5. Exit SST.
6. Enter WRKHDWRSC TYPE(*STG) on the command line and press the Enter key. A list of storage resources is displayed. Beside each resource of type 6110 or 6111, enter a 9 (Work with resource).
7. A list of storage controller resources is displayed. Look at the entries that have a description of Disk Storage Control. Use F11 (Display resource addresses/statuses) to display the resource address of each resource.
8. Look at the *resource address* field of each disk unit and disk storage controller and determine its bus number, storage controller number, and disk storage controller number. Refer to Figure 19-2 on page 19-6.
9. It may be helpful for you to draw a diagram to represent the data collected from the Disk Configuration Status display and the storage controller resources list, and to show how each disk unit and controller is attached. (See Figure 24-1 on page 24-4.)

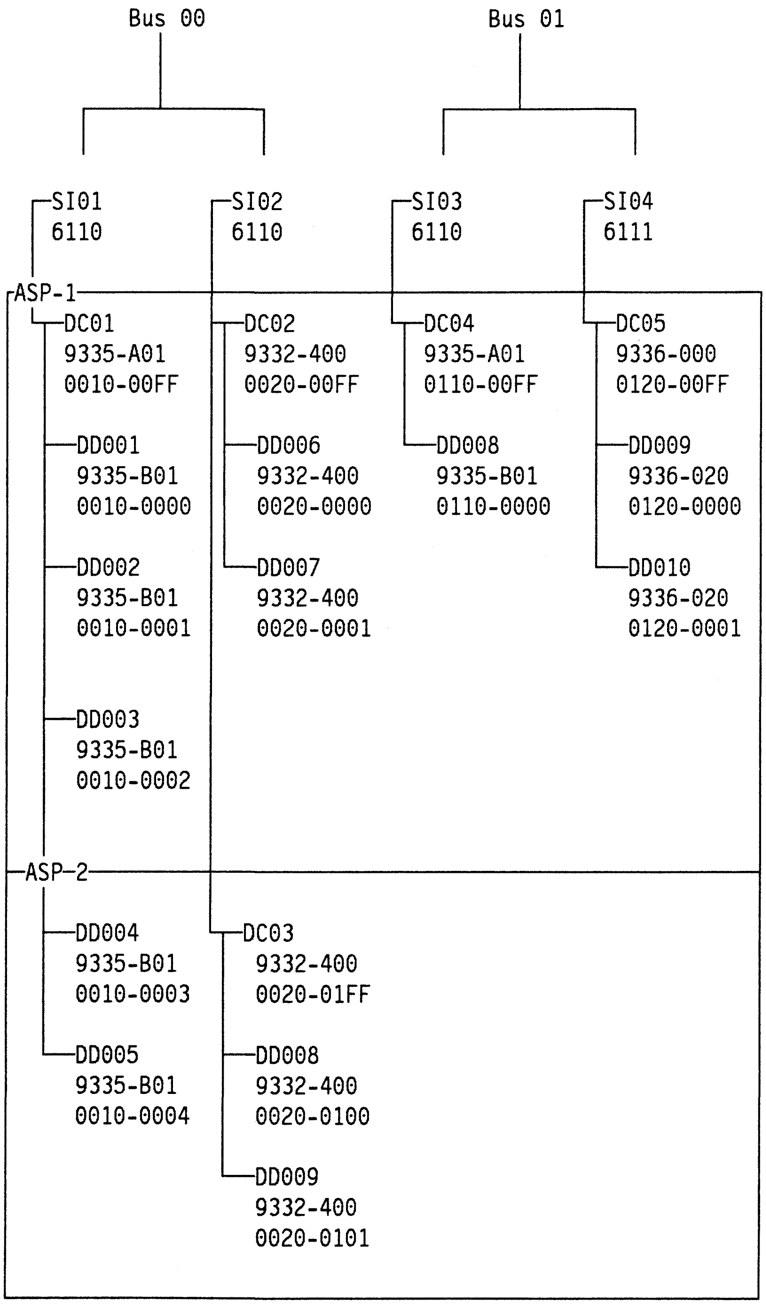


Figure 24-1. Hardware Configuration Diagram

Balancing Your Configuration

With the help of your IBM marketing representative, follow the standard configuration rules in the *9406 System Installation and Upgrade Guide*, and plan how to reconfigure your system. If the system is configured according to the standard rules, it is balanced to provide the best possible mirrored protection with the hardware available. You may find it helpful to use diagrams to note the new configuration.

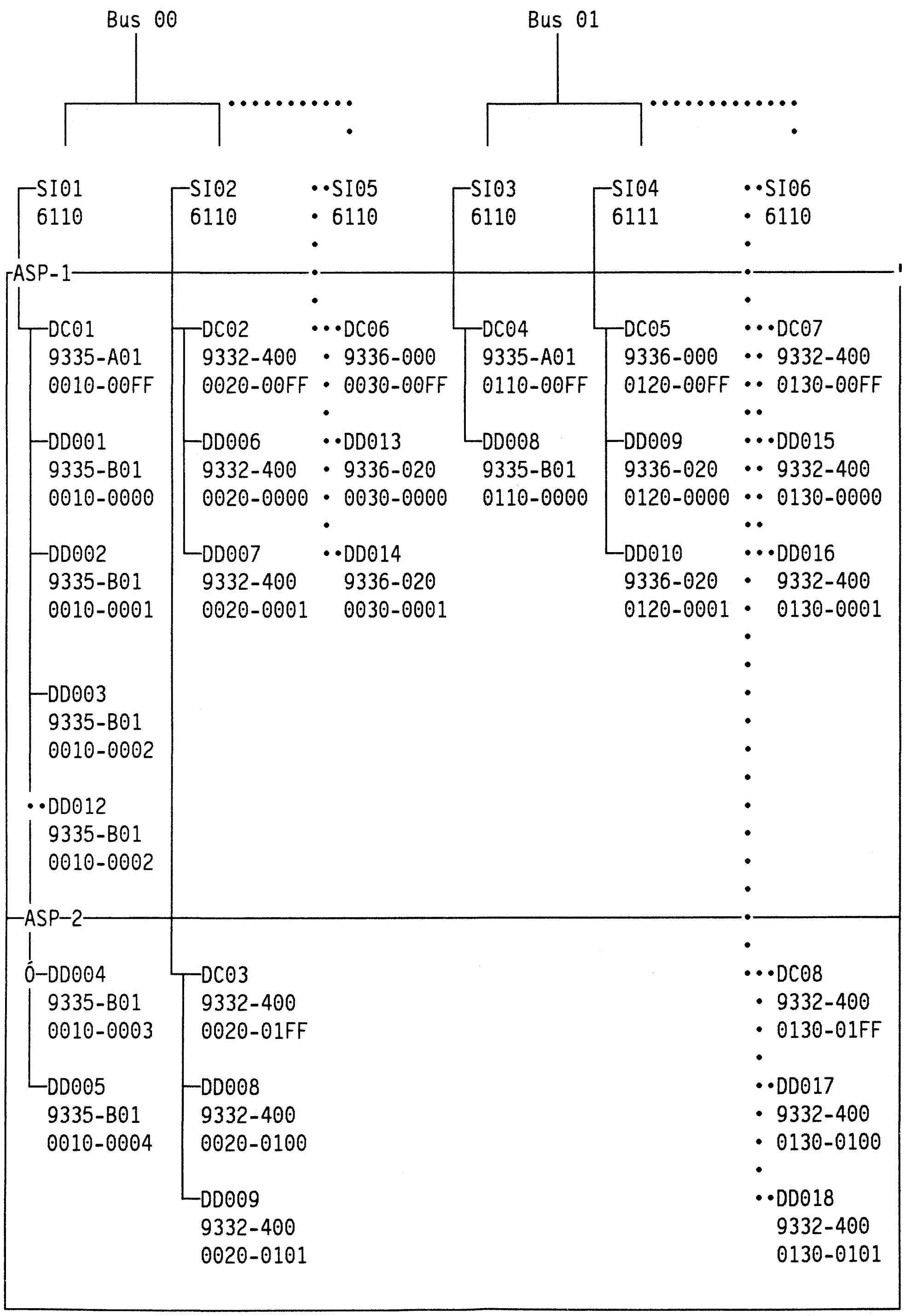


Figure 24-2. Augmented Hardware Configuration

Figure 24-2 shows how the configuration in Figure 24-1 on page 24-4 looks after you add hardware to allow for mirrored pairs and maintain the same capacity for growth. Notice that all disk units have bus-level protection except for unit 1, which has controller-level protection.

Examples of Mirrored Protection Configurations

Example of a Mirrored Configuration with Disk Unit-Level Protection

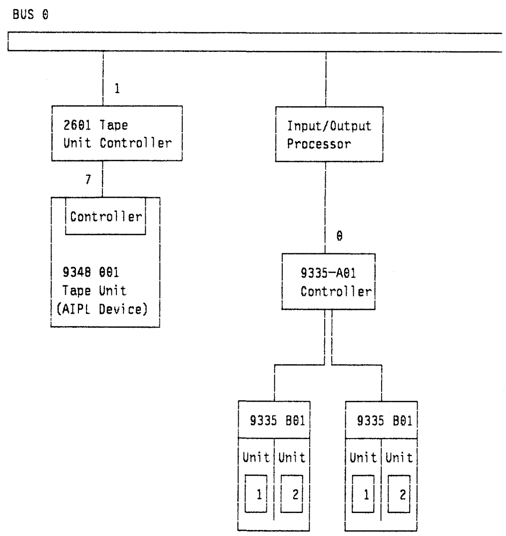


Figure 24-3. Example of a Mirrored Configuration with Disk Unit-Level Protection

Figure 24-3 shows a small system with disk units of type 9335, one controller, and one I/O processor. The system has disk unit-level protection. This configuration is not the best; a failure of the bus, the I/O processor, or the 9335-A01 controller causes all disk units to be unusable, and the system to end abnormally.

Concurrent maintenance may not be possible because of the requirement of the test procedure to have the controller dedicated to the repair action.

To increase the effectiveness of mirrored protection for this set of disk units, an additional 9335-A01 disk storage controller is required. If the controller is added, the failure of any one of the disk units or a 9335-A01 can be tolerated. Because they must be at I/O processor address 1 or 2, there is no way for unit 1 to be under separate I/O processors.

Concurrent maintenance is not possible if the I/O processor must be reset as part of the problem analysis procedures.

Example of a Mirrored Configuration with Controller-Level Protection

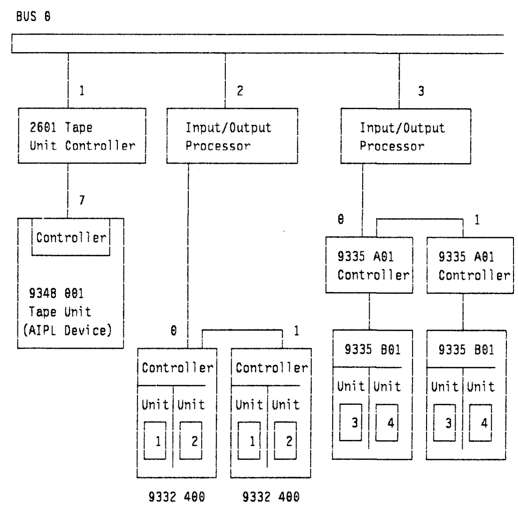


Figure 24-4. Example of Mirrored Configuration with Controller-Level Protection

Figure 24-4 shows a small-to-medium sized system with two different disk unit types (9332s and 9335s), two I/O processors, and controller-level protection for all disk units. I/O processor 2 controls the 9332 disk units, while I/O processor 3 controls the 9335 disk units. All disk units have controller-level protection because each disk unit and its mirrored unit are under separate controllers. This is the best protection possible with this hardware configuration.

Any disk unit or controller can be powered off while the system is running. However, if the problem analysis procedures need to reset the I/O processor controlling the disk unit or controller that has a failure, then concurrent maintenance is not possible.

Example of a Mirrored Configuration with I/O Processor-Level Protection on Units Other Than Unit 1

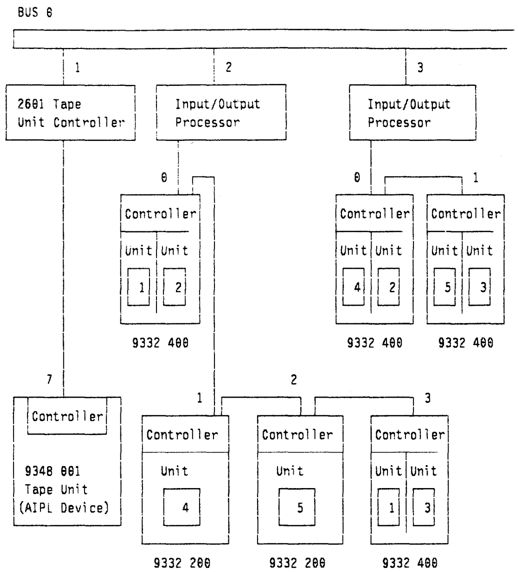


Figure 24-5. Example of a Mirrored Configuration With I/O Processor-Level Protection on Units Other Than the Unit 1

Figure 24-5 shows a system with all disk units of the same type (9332) and two disk I/O processors. All units other than unit 1 have I/O processor-level protection. This is the maximum protection possible on a single-bus system.

If I/O processor 2 has to be reset during the problem analysis procedures, then concurrent maintenance is not possible because both unit 1 and its mirrored unit are under this I/O processor. For units under I/O processor 3, concurrent maintenance is usually possible.

Example of a Mirrored Protection Configuration with Bus-Level Protection on Units Other Than the Unit 1

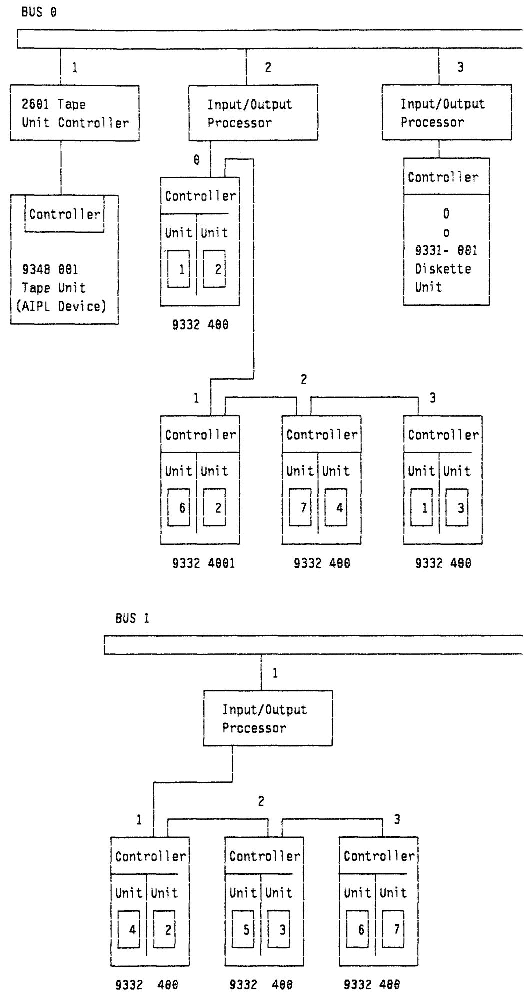


Figure 24-6. Example of a Mirrored Configuration with Bus-Level Protection

Figure 24-6 shows a two-bus system with 9332 disk unit types and bus-level protection for all disk units except unit 1. This means that bus 1 can fail while the system continues to run. Concurrent maintenance is usually possible for all units. Unit 1 may not be serviced during system operation if I/O processor 2 has to be reset as part of the problem analysis procedures.

If I/O processor 1 on bus 1 has to be reset as part of the problem analysis procedure, then all disk units in the system except unit 1 and its mirrored units are suspended and therefore unprotected for the duration of the maintenance.

To improve protection on this set of disk units, additional I/O processors on bus 1 are required. Storage units 3 and 5 under I/O processor 1 can be moved to an additional I/O processor. This reduces the number of storage units that are unprotected during the maintenance of any disk units on bus 1.

Part 7. Uninterruptible Power Supply

Chapter 25. Description of Power Loss Recovery

One way to help prevent the system from ending abnormally because of a power failure is to provide an uninterruptible power supply. An uninterruptible power supply provides auxiliary power supply to the system during a power failure. Two types of power supplies are supported.

The most frequently used type of uninterruptible power supply provides *complete* support by supplying power to the processing unit and all disk units long enough to last through a temporary power outage or to power the system down normally.

Uninterruptible power supplies vary, but Figure 25-1 shows a logical view of a typical uninterruptible power supply:

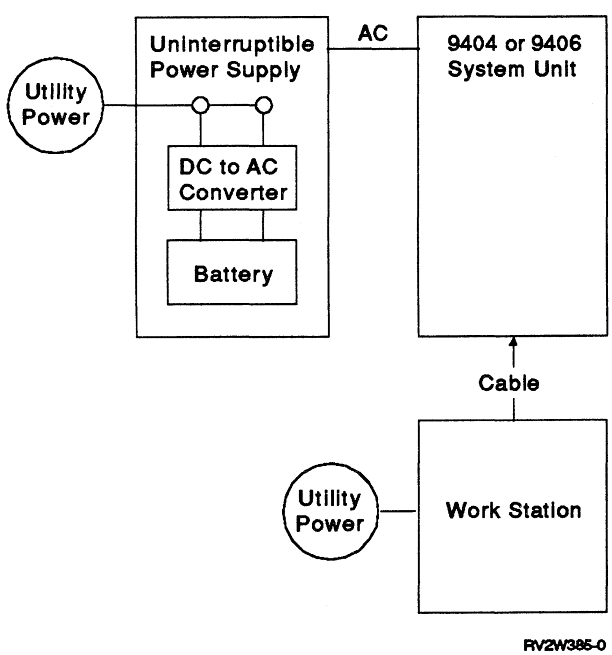


Figure 25-1. Logical View of a Typical Uninterruptible Power Supply

Another type of uninterruptible power supply provides *limited* support by supplying power to a limited number of devices: the processing unit, unit 1[1], and all storage device controllers. The system continues to run for up to 30 seconds. If power does not return in 30 seconds, the system saves main storage and a power down of the system occurs.

Unit 1 contains the licensed internal code and resides on the disk unit from which the system performs an IPL. Storage device controllers are cards in the card enclosure of the system that disk units attach to.

AS/400 Uninterruptible Power Supply Support

There are two approaches to uninterruptible power supply support, depending on the type and model of the AS/400 system:

- Type 9402 and 9404. A battery feature exists on the system. This is the normal approach if you use an uninterruptible power supply. However, you can use a vendor-supplied uninterruptible power supply with or without the battery feature.

  Normally, if you use a vendor-supplied uninterruptible power supply, you will not have the battery feature. If you do, the battery feature is not activated unless the vendor-supplied power supply stops providing power.

  The uninterruptible power supply that provides limited support requires that power be supplied to only the processing unit, unit 1[1], and all storage device controllers. Therefore, a system with two system units and with disk units in the second system unit would not benefit from the use of an uninterruptible power supply that provides limited support because both system units would need power. In addition, a single 9404 System Unit would not benefit from the limited support power supply because the entire system would be powered.

- Type 9406. For stage 2 hardware, the system comes with a battery that allows the system time to copy main storage in the event a power failure occurs. If you have frequent power outages, consider using an uninterruptible power supply.

[1] If the system ASP has mirrored protection, the mirrored unit for unit 1 is also supplied power.

UPS

- Type 9406. For stage 1 hardware, you need a vendor-supplied uninterruptible power supply.

Uninterruptible Power Supply Attachment

The uninterruptible power supply attachment allows you to attach a vendor-supplied uninterruptible power supply. The uninterruptible power supply attachment can be used on all models. The vendor-supplied uninterruptible power supply must be able to supply power to the processing unit and all disk units. The vendor-supplied uninterruptible power supply optionally supplies power to the other devices (work stations).

A vendor-supplied uninterruptible power supply that provides limited support need only provide power to the processing unit, unit 1, and the storage device controllers.

You control whether a power down of the system occurs or operations continue when the utility power is interrupted. See Figure 25-2 for an example of the attachment of an uninterruptible power supply supplied by a vendor.

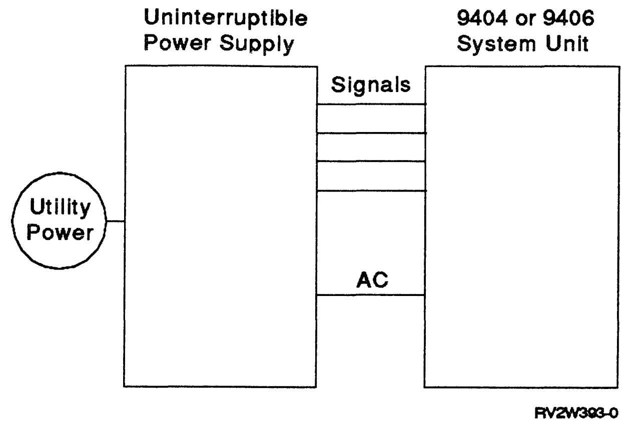


Figure 25-2. Attachment of a Vendor-Supplied Uninterruptible Power Supply

Power Loss Recovery

This topic discusses the system values that affect the uninterruptible power supply. Three system values define the action the system takes in response to a change in one of the power supply signals, as shown in Figure 25-3.

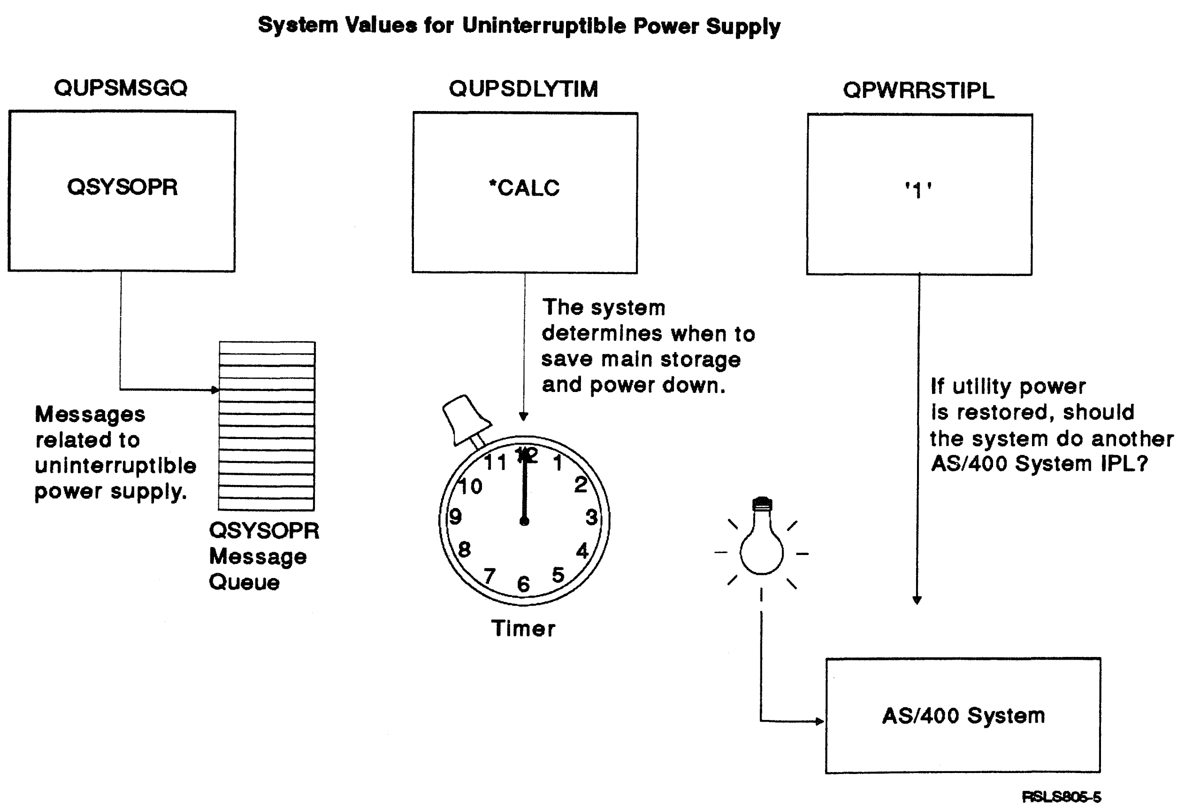


Figure 25-3. System Values That Define Uninterruptible Power Supply Action

The system values shown previously do the following:

- QUPSMSGQ - Uninterruptible Power Supply Message Queue

  Messages sent by the system about the power supply are sent to the system operator (QSYSOPR) message queue regardless of the value specified in the system value. If you specify a different message queue, that message queue also receives the same power supply messages. Specify a different message queue if you have one of the following:

  - Another message queue you want to receive the power supply messages (for example, the data processing manager's message queue).
  - A program that handles events that are related to the uninterruptible power supply. For more information about writing a program to handle the power, see the topic "Using a Program to Handle Uninterruptible Power Supply Conditions" on page 25-7.

  See the topic "Uninterruptible Power Supply Messages" on page 25-11 for information on power supply messages and a description of the messages and message IDs.

- QUPSDLYTIM - Uninterruptible Power Supply Delay Time

  The default value, *CALC, causes the system to calculate the wait time in seconds for the Battery Power Unit. This value controls the amount of time the system waits before saving main storage and powering down the system.

  If you have the uninterruptible power supply attachment, do not use *CALC. The value *CALC is designed for the 9402 and 9404 Battery Power Unit feature. Instead, specify a value based on the following.

  When the system senses that it is on backup power (either the uninterruptible power supply or the Battery Power Unit) a timer is set based on the QUPSDLYTIM system value. If utility power is restored before the time ends, the system ends the timer. If the timer ends first, the system saves main storage and powers down.

  You can also:

  - Specify *BASIC if the uninterruptible power supply attached to the system provides limited support. This value must be specified if the uninterruptible power supply that is attached provides power for only the processing unit, unit 1[1], and the storage device controllers.

    The system sets a timer based on the system configuration. If utility power is restored before the time ends, the system ends the timer and normal operations continue. If the timer ends first, the system saves main storage and powers down the system.
  - Specify a value of 0 to ask the system to immediately save main storage and then power down. Normally, you only specify this value when you are controlling the uninterruptible power supply conditions with a program and suspect that the battery is not properly recharged.
  - Specify a different value based on the duration of your battery's charge and the time it normally takes the system to save main storage and power down. See the topic "Setting the QUPSDLYTIM System Value" on page 25-4.
  - Specify a value of *NOMAX to tell the system that you do not want this function to save main storage and then power down. Normally, specify *NOMAX only when you are controlling the uninterruptible power supply conditions with a program. See "Using a Program to Handle Uninterruptible Power Supply Conditions" on page 25-7.

- QPWRRSTIPL - Power Restore IPL option following a power down

  This Yes or No value controls what happens if the system is ended when utility power is off and then power is later restored. The default is '0', meaning not allowed, which causes the system to not perform an IPL when utility power is restored. The value '1' (Allowed) is normally appropriate if:

  - You do not have a Battery Power Unit or an interruptible power supply attachment, and the utility power is lost. The system immediately ends. When utility power is restored, the system value is used to determine if the system should automatically do an IPL.

UPS

- You have a Battery Power Unit or an uninterruptible power supply attachment, and you use either the Power Down System (PWRDWNSYS) command or the power down function for the limited power support while utility power is off and the uninterruptible power supply is providing power.

  The system is powered down when either the PWRDWNSYS command or the limited support power-down sequence starts. If utility power is restored during the sequence, the system is powered down and an IPL occurs if the system value QPWRRSTIPL is set to '1'. If utility power is restored after the system is powered down, the system uses the value to see if an automatic IPL should occur.

The keylock switch must be in the Normal or Automatic position for the automatic IPL function to work.

The QPWRRSTIPL system value has no effect if the system is powered down normally while running on utility power. If utility power goes off after the system is powered down normally, the system does not automatically do an IPL when utility power is restored.

Normally, you would only set this value to '0' if:

- You prefer to manually start the system again.
- You have a power handling program that determines whether the batteries have not been recharged enough to allow another IPL.

The system may override a '1' value to try to protect the battery from being completely discharged. There are two cases:

- During an IPL, if utility power fails and a weak battery is signaled before the OS/400 licensed program is given control, the system will save main storage, power down, and override the power restore option to Not allowed.
- When performing an IPL due to the power restore IPL value and a utility power failure occurs, if the OS/400 licensed program is not given control before the power supply delay time ends, the system will save main storage, power down, and override the power restore option to Not allowed.

Setting the QUPSDLYTIM System Value

With the Battery Power Unit on the 9404 System Unit, you would normally use the default value of *CALC. This should allow sufficient time for the system to save main storage and power down, based on the battery rating and the maximum size of main storage.

With the uninterruptible power supply attachment, the value *BASIC must be specified if an uninterruptible power supply with limited power support is providing power only to the processing unit, unit 1[1], and the storage device controllers (this means that not all system units are powered by the uninterruptible power supply). This approach normally requires a less expensive uninterruptible power supply but provides less function. Tables Table 25-1 and Table 25-2 on page 25-5 show the time it takes for the basic uninterruptible power supply to shut down.

Table 25-1. Basic Uninterruptible Power Supply. Save Main Storage Times for the 9402, 9404, and 9406 System Units, Model B.

Main Storage Size	Time (seconds)	Time (minutes)
8	45	0 minutes 45 seconds
16	60	1 minute 0 seconds
32	95	1 minute 35 seconds
64	160	2 minutes 40 seconds
96	225	3 minutes 45 seconds
128	300	5 minutes 0 seconds
160	350	5 minutes 50 seconds
192	420	7 minutes 0 seconds

Table 25-2. Basic Uninterruptible Power Supply. Save Main Storage Times for the 9406 System Unit, Model (Internal Battery)		
Main Storage Size	**Time (seconds)**	**Time (minutes)**
8	35	0 minutes 35 seconds
16	40	0 minutes 40 seconds
32	60	1 minute 0 seconds
64	75	1 minute 15 seconds
96	95	1 minute 35 seconds
128	115	1 minute 55 seconds
160	135	2 minutes 15 seconds
192	160	2 minutes 40 seconds
224	180	3 minutes 0 seconds
256	200	3 minutes 20 seconds
288	220	3 minutes 40 seconds
320	240	4 minutes 0 seconds
352	265	4 minutes 25 seconds
384	280	4 minutes 40 seconds

The basic uninterruptible power supply attached to the system must provide power to the system at least as long as the time in Table 25-2. This time includes the shutdown time in addition to a 30-second delay time used when waiting for utility power to return.

The internal battery of a 9406 Model D system provides power to the system for 5 minutes. This battery provides the uninterruptible power supply with limited support by default.

For more information, see "Power Down for Limited Power Supply Support" on page 25-7 for the function your system must perform.

With the uninterruptible power supply attachment, do not use the value *CALC. Consider how long your battery stays charged and how long the system takes to save main storage and then power down. Specify the difference between these two as the time for the uninterruptible power supply delay time (QUPSDLYTIM), that is, how long the system should wait before it saves main storage and powers down. The time line of the function is shown in Figure 25-4.

If you have a main storage size of 16MB or less, consider a 5-minute value for the length of time to save main storage and power down. For a larger system, add 1 minute for each 16MB. For example, for a 64MB system you would use a value of 8 minutes (5 + 3).

This minutes-per-megabyte number is not an exact number. It is an estimate based on the number of jobs that normally are run in different main storage sizes and the average amount of time it takes for the system to end them. For more information, see the topic "Power Down When Using a Power Supply on 9402 or 9404 Battery Power Unit" on page 25-6 for the function your system must perform. Add a buffer of time and make a conservative estimate to account for unusual circumstances. Also consider that if power is fluctuating, your battery may be discharged to the point where it may not have enough time to save main storage and power down.

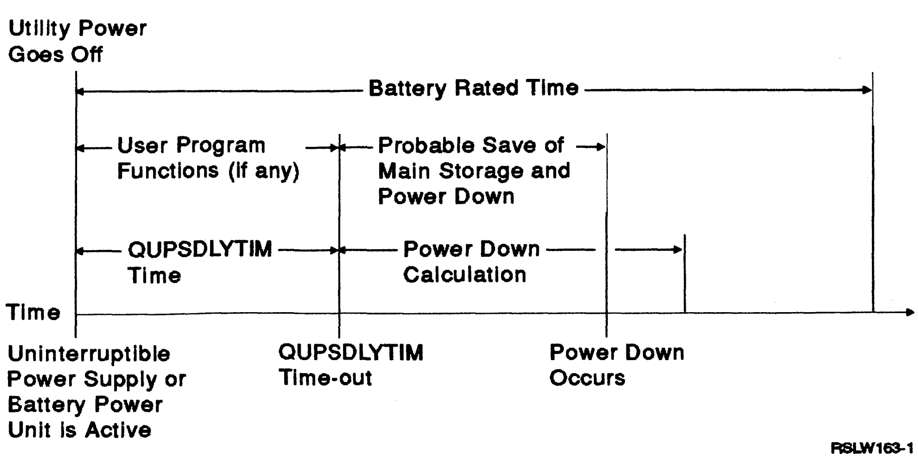


Figure 25-4. Time Line of QUPSDLYTIM Function

When you know your battery time duration and the time it takes to save main storage and power down, subtract to determine the value to specify for QUPSDLYTIM. For example, if you have a 32MB system and a battery rating of 15 minutes, calculate as follows:

```
Battery rating           15  Minutes
Save-main-storage and   - 6          (5 Standard + 1)
power-down time         ----
                          9
Buffer                  - 2
                        ----
                          7  Minutes (420 seconds)
```

To change the system QUPSDLYTIM value according to the above calculation, specify:

```
CHGSYSVAL SYSVAL(QUPSDLYTIM) VALUE('420')
```

If you change the main storage size, consider changing the QUPSDLYTIM value.

Normal Power Down: To properly power down the system, end all other work on the system. Then, type the Power Down System (PWRDWNSYS) command. The PWRDWNSYS command can be run with OPTION(*IMMED) or with a time delay, which causes any active jobs to be abnormally ended. In either case, successfully completing the PWRDWNSYS command causes the next IPL of the system programs to be considered normal by both the licensed internal code and the OS/400 licensed program.

Power Down When Using a Power Supply on 9402 or 9404 Battery Power Unit: The power-down function when using power supply support operates from the licensed internal code and does not involve the OS/400 licensed program. It cannot be started from a command (the PWRDWNSYS command is not run). There are multiple phases and considerations for the power-down function associated with power supply support:

- The licensed internal code signals each job to end at the next internal instruction boundary. (This same function, known as forced licensed internal code completion, can be started from the control panel. For more information about licensed internal code refer to the topic "Licensed Internal Code Completion" on page D-9.) Each job attempts to come to a point where an orderly end can occur.

  Usually, a job is at the next internal instruction boundary or will be shortly. However, some internal instructions are long running, such as those that build access paths or create programs. A job performing one of these instructions may not be able to reach an orderly end in the time allowed.

  This first phase is shorter than that performed by the forced licensed internal code completion function. The time allowed for completion varies, depending on the main storage size of the system.

- If all jobs have reached a point where they can properly end, the system performs additional operations and then powers down. For example, the system must write all of the changed pages from main storage to auxiliary storage. The next IPL is normal from a licensed internal code viewpoint, but is considered abnormal from the operating system viewpoint. There is no directory recovery or access path recovery required.

- The licensed internal code ends jobs that cannot reach a point (in a reasonable amount of time) where they can end properly, and continues to power down. The next IPL does not involve directory recovery, but may require some access path recovery, depending on what was happening to the job when it ended.

- In some cases the power-down function cannot complete successfully. The next IPL will be abnormal from a licensed internal code viewpoint.

The power-down sequence sets the system to do an IPL automatically when utility power is restored if the system value QPWRRSTIPL is set to '1'.

The system completes the power down if utility power is restored while the system is in the power-down sequence. If the QPWRRSTIPL system value is set to '1', the system automatically performs an IPL after the power down is complete.

The power-down function for power supply support always causes an abnormal IPL from an OS/400 point of view. This allows certain cleanup functions to occur, but normally will not significantly affect the IPL time.

Power Down for Limited Power Supply Support: The power-down function for the limited power supply support (*BASIC) operates from the licensed internal code and does not involve the OS/400 licensed program. It cannot be started from a command, but is selected by specifying the value *BASIC for the QUPSDLYTIM system value. The following is performed by specifying the *BASIC value:

- Contents of main storage are copied (saved) to the main storage dump area in unit 1.
- The system is powered down.
- The next IPL is abnormal from the viewpoint of the licensed internal code.

  The system must write all of the changed pages from the saved copy of main storage to auxiliary storage. The system performs a directory recovery that is shortened and you may need to recover certain access paths. Typically, the recovery time should be significantly reduced.

The power-down sequence using *BASIC sets the system to do an automatic IPL when utility power is restored if the system value QPWRRSTIPL is set to '1'.

The system completes the power-down if utility power is restored while the system is in the *BASIC power-down sequence. If the QPWRRSTIPL system value is '1', the system automatically does an IPL after the power down is complete.

The power-down function using *BASIC always causes an abnormal IPL from an OS/400 point of view. This allows certain cleanup functions to occur, but normally does not significantly affect the IPL time.

Using a Program to Handle Uninterruptible Power Supply Conditions

System software support is essentially the same for both the Battery feature and the uninterruptible power supply attachment. You control the type of processing you want to perform when utility power is interrupted through system values and an optional user-written program.

When No User Power-Handling Program Exists

You can specify that you do not have a power handling program by using the default for the QUPSMSGQ system value, which is QSYSOPR. This causes all power-related messages to be sent to QSYSOPR. The QUPSDLYTIM system value should be set to something other than *NOMAX.

Normally, you supply power to the racks and not to all of the work station devices. When utility power is interrupted, the system remains active, but the work station jobs usually end abnormally. If utility power is restored while operating on the uninterruptible power supply, the system remains active and the work station jobs can be restarted.

If the QUPSDLYTIM timer ends or the weak-battery signal occurs, the system saves main storage and powers down. Change the QUPSDLYTIM value according to your uninterruptible power supply and your system size. This is particularly important if you do not have a weak-battery connection or if the time is fixed by the uninterruptible power supply vendor and is not appropriate for your AS/400 configuration. For more information, see "Weak-Battery Signal" on page 25-11 and "Setting the QUPSDLYTIM System Value" on page 25-4.

If the system is powered down while on uninterruptible power supply, the QPWRRSTIPL system value determines whether an IPL is performed when utility power is restored. The default is to not perform the IPL.

When a Power-Handling Program Does Exist

In some environments you may want to perform different actions when you begin operating on uninterruptible power supply or when power is fluctuating. A user program can handle these situations by sending specific messages to interactive users, ending batch jobs in preparation for powering down, or dynamically changing the system values that control uninterruptible power supply processing.

To specify that you have power handling programs, change the QUPSMSGQ system value to the name of a queue you have created. The same messages will be sent to both QSYSOPR and the queue you specified. Change the QUPSDLYTIM system value to *NOMAX.

The program you use to handle the message queue must be active and must allocate the queue (normally done by using the ALCOBJ command). If a program has not allocated the queue specified in QUPSMSGQ, the system will assume no power handling program exists.

Writing the Program

A power handling program should be activated at each IPL and remain active at all times. It should be accounted for in the activity level available in work management subsystem specifications.

The message queue specified in QUPSMSGQ is used for uninterruptible power supply message processing. The program normally allocates the queue by specifying the command:

```
ALCOBJ OBJ(xxx/yyy *MSGQ *EXCL)
```

When a message arrives, the critical messages to process are:

- **CPF1816: System utility power failed at &1.** (this message applies to the battery feature and full power supply)
- **CPF1817: System power restored at &1.** (this message applies to the battery feature and full power supply)
- **CPI0994: System power is restored** (this message applies to the limited uninterruptible power supply)

You can choose to ignore the other messages.

Your program can handle a brief power interruption without doing any unique processing. For example, when the CPF1816 message arrives, you can set a switch in your program indicating that the message occurred. The program could then perform a RCVMSG with WAIT(10) to cause a time-out in 10 seconds. If the CPF1817 message is received before the time-out occurs, you can reset the switch and perform no other action.

Your program can prepare for a normal power down if power is not restored after a brief time period. For example, if you have remote work stations that are still active, you may want to send them a message requesting they sign off quickly. You may want to issue ENDSBS OPTION(*CNTRLD) to prevent new work stations from signing on or new batch work from beginning. If you have batch jobs running, you may want to end them with:

```
ENDJOB   OPTION(*CNTRLD)
```

This sets an indicator to end the job. Some higher level languages and the control language allow you to test within a program to see if a controlled ENDJOB was specified. If the program does not end itself, the default on ENDJOB (30 seconds) is used.

You can set a second timer in your program, such as RCVMSG WAIT(120). If utility power has not been restored, you can issue the PWRDWNSYS OPTION(*IMMED) command. The wait time should be specified based on your battery time and the time required for a power-down.

If you name a message queue (other than QSYSOPR) for the QUPSMSGQ system value and *NOMAX for QUPSDLYTIM, this message queue must be allocated by a program when the CPF1816 message occurs, or be in a break or notify mode if it is a work station message queue. If not, the system assumes no power handling program exists and the system will be powered down.

Note: When the system has been placed in a restricted state (for example, ENDSBS *ALL), your uninterruptible power supply handling program will no longer be active. For this reason, it is necessary to prepare an alternate method of dealing with your uninterruptible power supply and any possible power interruptions that may occur while your system is in a restricted state.

For example, when performing a SAVSYS (Save System) or RCLSTG (Reclaim Storage), your uninterruptible power supply program will no longer be active once all subsystems have been terminated since only a single workstation job will be active. As an alternative:

1. Once all subsystems have been ended, from the command line change the message queue specified in system value QUPSMSGQ to

*BREAK. This will cause all uninterruptible power supply messages to be sent as break messages to the user signed on to that work station. With this method the user will manually decide what to do should a power failure occur.

2. Change the system value QUPSDLYTIM to some value other than *NOMAX (for example, the number of minutes you wish the uninterruptible power supply to ride out the power failure). This method will prevent the system from performing an immediate quick power down; however, if a power failure occurs, a quick power down will be performed should the power failure last longer than the value specified for the system value QUPSDLYTIM.

3. Modify your existing uninterruptible power supply handling program for use as a BREAK HANDLING program which may be used while the system is in a restricted state (all subsystems ended). This can be done by creating a second version of your uninterruptible power supply program that does not allocate the message queue specified in system value QUPSMSGQ (i.e, no ALCOBJ command). To utilize this program while in a restricted state, prior to starting a dedicated function such as SAVSYS, enter the command:

```
CHGMSGQ MSGQ(LIB/MSGQ) DLVRY(*BREAK)
        PGM(LIB/PGM)
```

where (LIB/MSGQ) is the name the message queue specified in system value QUPSMSGQ, and (PGM/LIB) is the name of your modified uninterruptible power supply handling program. Now, should a power failure occur, the power failure message will be handled by the break handling program, even while a function such as SAVSYS is running. To deactivate the break handling program either have the user sign off or enter:

```
CHGMSGQ MSGQ(LIB/MSGQ) DLVRY(*HOLD)
        PGM(*DSPMSG)
```

Once you have deactivated the break handling program you should immediately start your subsystems and your normal uninterruptible power supply handling program.

Weak Battery Conditions

Any indication of a weak-battery while on the uninterruptible power supply causes the system to save main storage and power down, no matter what your program is coded to do. If the system value QUPSDLYTIM is met, the system also automatically starts to save main storage and powers down. You can prevent this system action by not connecting the lead for the weak-battery condition.

If you are using the limited uninterruptible power supply, the weak battery signal causes the power-down sequence for limited power supply support (see the topic "Power Down for Limited Power Supply Support" on page 25-7).

If you do not have a weak-battery indication, you must consider the results if the utility power fluctuates. Normally, the uninterruptible power supply battery does not fully recharge unless the system has been on utility power for some time. Account for this in your program by adjusting the values for QUPSDLYTIM and QPWRRSTIPL.

For example, if power has gone off, comes back on, and then goes off again, you may want to adjust the QUPSDLYTIM to account for the fact that the battery is not fully recharged. If power has gone off and on several times in a 30-minute period, you may want to set QPWRRSTIPL to '0' (meaning No) and issue the PWRDWNSYS command. This prevents the system from doing another IPL. You could either wait for a manual IPL or set the QIPLTIME system value to cause a timed IPL at a later point. In general, it is desirable to prevent the battery from being totally used up.

Running the Program

Run the power handling program with a priority higher than your normal jobs. The system normally is very busy attempting to end and clean up interactive jobs that have lost power to the work stations. You may want to hold any batch work, even during a brief power interruption.

Program Flowchart

Figure 25-5 is a flowchart of a typical power handling program.

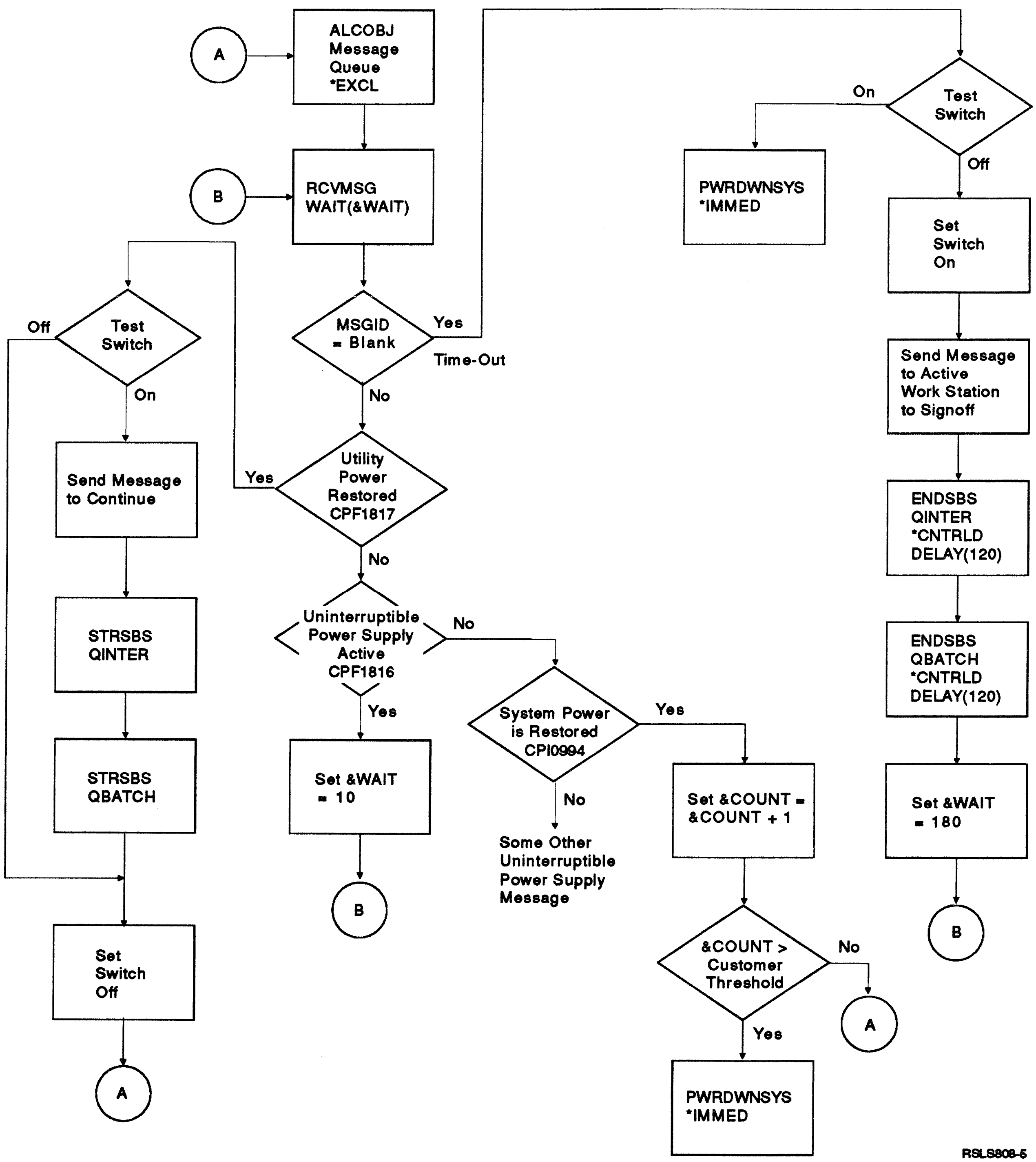


Figure 25-5. Flowchart of a Typical Power-Handling Program

Using an Uninterruptible Power Supply

This topic describes the 9404 System Unit Battery feature and user programs for handling uninterruptible power supply conditions.

Battery Feature

You should use the Battery feature with the system defaults for the 9404 System Unit. When you use the Battery feature, the system remains active if utility power is lost, although the work station jobs may end abnormally. You need to restart the work station jobs (sign on again) if utility power is restored while the system is operating on the battery. If utility power is not restored shortly, either the QUPSDLYTIM timer or the weak-battery signal causes the system to save main storage and power down.

If the system is powered down while on an uninterruptible power supply, the QPWRRSTIPL system value determines if an IPL will occur when utility power is restored. The default is to IPL the system programs.

Weak-Battery Signal

If you make the connection for a weak-battery signal and get a signal that the battery is weak, the system responds by saving main storage and powering down, regardless of the setting of the system values.

If a system is still running on utility power and a weak-battery signal is received, message CPI0964 is sent. The saving of main storage and power-down functions are performed only if the system is already running on an uninterruptible power supply or the Battery feature.

This is to provide the shortest possible IPL time in the event of a weak battery condition. This does not assure that sufficient time remains on the battery for the system to perform its power-down sequence.

On the Battery feature, the connection always exists. You cannot override the system action, which saves main storage and then powers down.

For the uninterruptible power supply attachment, this is an optional connection that can be made from certain uninterruptible power supply attachments. This connection allows the system to automatically save main storage and then power down. For most uninterruptible power supply, it is better not to completely use up the battery. The weak battery signal helps prevent this. If you do not want the weak-battery signal, do not make the connection.

Some vendor-supplied uninterruptible power supply attachments may have an adjustable setting to allow you to set the time for a weak battery condition. Other uninterruptible power supply attachments may be adjusted at the factory. Some may have a fixed value and others may not have any.

Do not assume that the time remaining on the battery is sufficient for the system to save main storage and power down if the uninterruptible power supply has a weak-battery condition. Calculate the amount of time necessary to save main storage and power down. For more information on setting the QUPSDLYTIM system value, see "Setting the QUPSDLYTIM System Value" on page 25-4.

Plan for an increase in the main storage size, the number of units in the existing racks, and the number of racks. The planning requirement is minimized by the QUPSDLYTIM value. That is, whichever occurs first (weak battery or QUPSDLYTIM) signals the system to save main storage and power down. However, the QUPSDLYTIM is a constant value and cannot change to respond to unusual occurrences involving the uninterruptible power supply. The weak battery signal can account for these with more reliability.

Uninterruptible Power Supply Messages

The uninterruptible power supply messages are always sent to QSYSOPR and then to QHST. Optionally, the same messages can be sent to a user-specified message queue.

The following describes the uninterruptible power supply messages that are sent to QSYSOPR and to the named message queue specified in the QUPSMSGQ system value (if it differs from QSYSOPR):

CPF1816: System utility power failed at &1.
The system power switched to the auxiliary source.

CPF1817: System utility power restored at &1.
The system power switched to the utility source.

CPI0961: Uninterruptible power supply (UPS) no longer attached.

CPI0962: The uninterruptible power supply (UPS) is now attached.

CPI0963: System power supplied by uninterruptible power supply.
System power is currently being supplied by the uninterruptible power supply (UPS).

CPI0964: External UPS indicates weak-battery condition.
If utility power fails during this condition, the system may begin an immediate power down. See the Programming: Backup and Recovery Guide and your uninterruptible power supply manual for more information.

CPI0965: Failure of battery backup feature in system unit.
There may be a failure of the battery or the battery charger for the battery backup feature in the system unit. Contact your service representative.

CPI0966: Failure of battery backup feature in expansion unit.
There may be a failure of the battery or the battery charger for the battery backup feature in the expansion unit. Contact your service representative.

CPI0973: External UPS no longer indicates battery weak.
Refer to your UPS manual.

CPI0974: UPS has been bypassed.
If a utility power failure occurs, the uninterruptible power supply cannot supply system power. The system will end abnormally.

CPI0975: UPS no longer bypassed.
The uninterruptible power supply is no longer bypassed.

CPI0976: Notification of message &1 failed.
Unable to send &1 message to message queue. &2 in library &3 specified in QUPSMSGQ system value.

CPI0994: System power is restored.
The system power switched to the utility source at &1. The utility power failed for &2 seconds. During this time the system was not doing any application processing. If the utility power continues to fail, power down the system (PWRDWNSYS command).

CPI0981: Automatic IPL disabled.

Automatic IPL after utility power restored, specified by system value QPWRRSTIPL, was disabled for one of the following reasons:

- Utility power failed and the battery weak condition was detected during the IPL.
- Utility power failed during the IPL and the uninterruptible power supply delay time specified in system value QUPSDLYTIM was exceeded before the IPL complete.

IPL Considerations

When the system performs an IPL, the licensed internal code verifies various internal switches to see if the system was correctly powered down. Only the successful completion of the Power Down System (PWRDWNSYS) command causes the AS/400 system to be correctly powered down. The licensed internal code considers the IPL to be normal if the saving of main storage and the power-down sequence complete successfully; however, the OS/400 program considers the next IPL to be abnormal. If neither power-down technique completes normally, the licensed internal code runs various recovery functions on the next IPL.

When an abnormal IPL occurs, the OS/400 program performs additional recovery functions. In an attended IPL, you can control some of these functions. In an unattended IPL (caused by the QPWRRSTIPL system value or a timed IPL), only the defaults can be taken.

If the Power Down System (PWRDWNSYS) command is run while power is being supplied by the Battery feature or uninterruptible power supply, the system delays writing any job logs until the next IPL. The system handles this type of PWRDWNSYS so that the amount of processing is minimized. The system will not do an IPL while operating on the 9404 System Unit Battery feature.

You can IPL the system if utility power is off and the system is operating on an uninterruptible power supply. This does not apply for a timed or remote IPL. Only a manual IPL is allowed when utility power is interrupted.

Uninterruptible Power Supply Flowchart

Figure 25-6 on page 25-13 is the logical flowchart of how the AS/400 system handles the signals from the uninterruptible power supply.

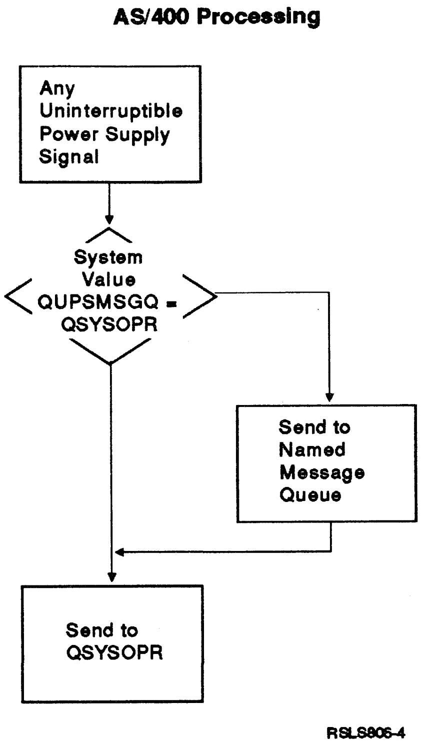


Figure 25-6. How the AS/400 System Handles Uninterruptible Power Supply Signals

Figure 25-7 on page 25-14 is the logical flowchart of how the licensed internal code handles the signals from the uninterruptible power supply.

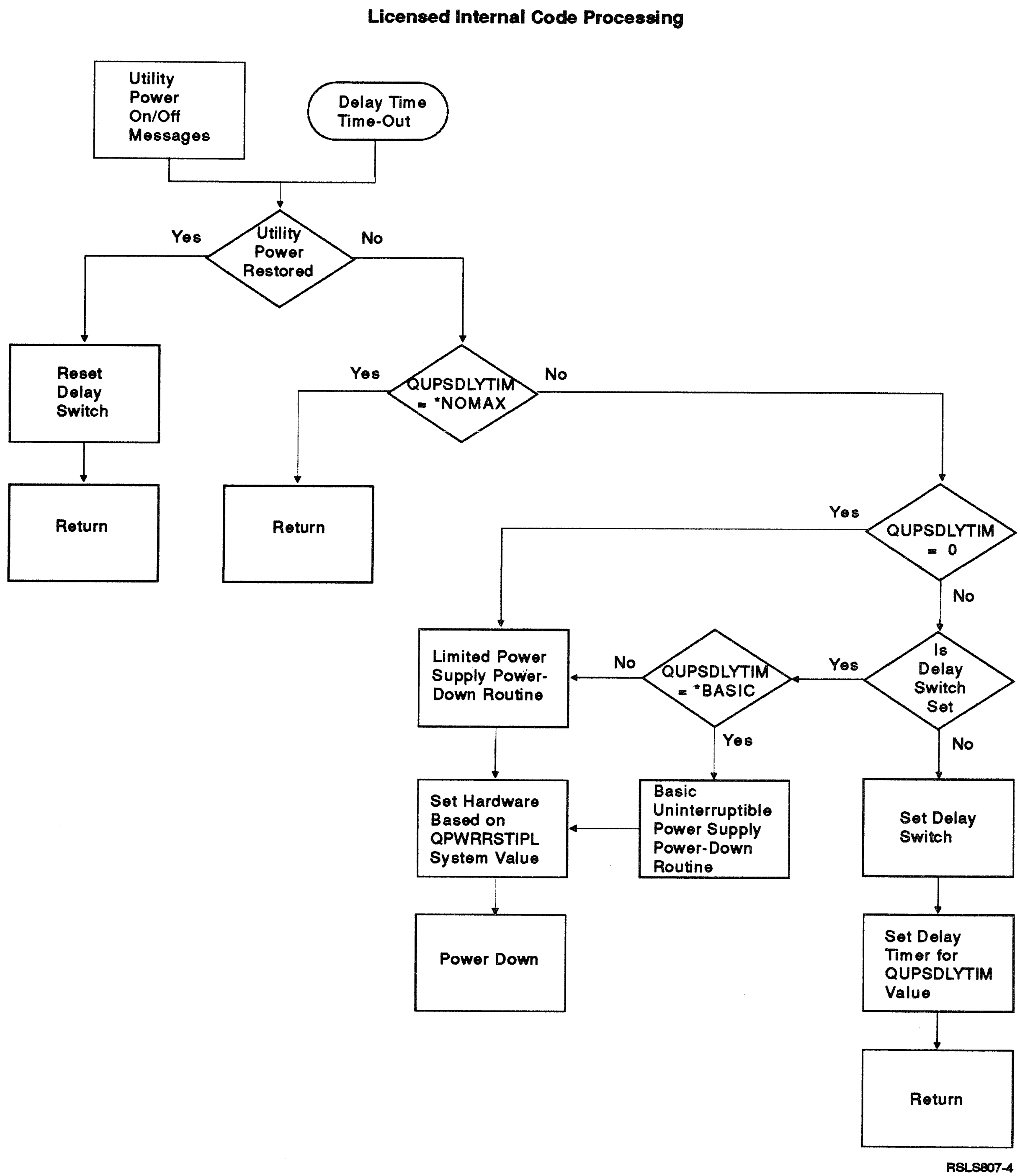


Figure 25-7. How the Licensed Internal Code Handles Uninterruptible Power Supply Signals

Part 8. Appendices

Appendix A. Licensed Internal Code SRCs That Require User Input (A6xx xxxx)

The following topics describe licensed internal code SRCs that require user input (A6xx xxxx).

Function 11, Data Code A6xx 6001

This topic describes the user input required by the licensed internal code SRC A6xx 6001.

Description Select the utility (Licensed Internal Code Install or Licensed Internal Code Restore you want to run.

Reply Using the Select key on the control panel, select the function code for the utility you want to run, then press the Enter key:

Function Code	*Utility Selected*
23	Stand-Alone Licensed Internal Code Restore The stand-alone licensed internal code restore utility copies all system licensed internal code from the tape and writes over the licensed internal code found on the disk. Select this utility to exchange or update an existing system's licensed internal code without losing customer data already on the system.
24	Stand-Alone Licensed Internal Code Install The stand-alone licensed internal code install utility deletes all information found on unit 1 (including customer data) and copies all system licensed internal code from tape to disk. The data on all the remainder of the disk units will not be erased but may not be accessible (because of the way data is spread over multiple units on the system). Select this utility when initially starting up a new system (which contains no customer data and no licensed internal code) or in cases where the primary disk was exchanged.
29	Load Model-Unique Licensed Internal Code The load model-unique licensed internal code utility copies only the licensed internal code from tape, and writes over any existing licensed internal code on disk. Select this utility when a hardware model upgrade is performed.
32	Download disk licensed internal code The download disk licensed internal code utility downloads the licensed internal code from tape to disk. Use this function when the disk licensed internal code should be changed.
(None)	Cancel this request. If you do not want to select any of these options, turn the system control panel off.

Function 11, Data Code A6xx 6002

This topic describes the user input required by the licensed internal code SRC A6xx 6002.

Description **Warning:** Do you want to destroy all data on all disk units?

The stand-alone licensed internal code install was requested; however, the disk unit that contains storage unit 1 already contains data. Continuing with licensed internal code install will delete all data on the disk unit that contains unit 1. The data on all the remainder of the disk units will not be deleted, but may not be accessible (because of the way data is spread over multiple units on the system).

Function codes may be used to display to which disk the system is attempting to install.

The function codes are different for systems with stage 2 hardware. The stage 2 hardware function codes are identified with a **-2** and are enclosed in parentheses (such as 15-2).

Function Code — *Function*

14 (15-2) Display type and model of the disk unit that contains unit 1. The first 4 characters displayed in the lights will show the type. The next 4 characters show the model.

An example of what would be shown after you select this option follows. (This shows what you would see if the disk was a 9332 Model 400.)

14 (15-2)	9332 0400

15 (16-2) Display address of the disk unit that contains unit 1. Eight characters will be shown on the display lights. The meaning of these characters is shown below.

Character Position	***Description***
1–2	Bus number (should be zero)
3	IOP card number
4	IOP board number
5–6	Facility address (DFCI)
7–8	Secondary address

The following is an example of what would be shown after you select this option:

15 (16-2)	0010 0700

16 (17-2) Display the serial numbers of the disk unit that contains unit 1.

The following is an example of what should be shown after you select this option:

16 (17-2)	0012 3456

Reply Using the Function Select key on the control panel, type one of the following codes and press the Enter key:

Function Code	*Function*
23	Restore the licensed internal code without destroying customer data. Select function code 23 if you want to restore the licensed internal code to the disk unit that contains storage unit 1 without deleting any other data from the system. Selecting this option will copy all system licensed internal code from tape and will exchange the licensed internal code found on unit 1. Select this option if you want to exchange or update an existing system's licensed internal code without losing customer data.
24	Destroy all system data and restore the licensed internal code. Select function code 24 if you want to restore the licensed internal code to the selected disk unit, and you want to *destroy all data* on the system. Selecting this option will first delete all information found on the disk unit that contains storage unit 1 (including customer data) and then will copy all system licensed internal code from tape to disk. The data on all the remainder of the disk units will not be deleted but may not be accessible (because of the way data is spread over multiple units on the system). Select this option if you are installing a new system (which contains no customer data and no licensed internal code), or if you have exchanged the disk unit that contains storage unit 1.
(None)	Cancel this request. If you do not want to continue with the licensed internal code install, turn the control panel off.

Function 11, Data Code A6xx 6003

This topic describes the user input required by the licensed internal code SRC A6xx 6003.

Description **Warning:** Do you want to destroy all data on all disk units?

The stand-alone licensed internal code install was requested. The system found a disk unit attached at the correct location for being the disk unit that contains storage unit 1. However, the disk unit found already contains data, and the data is not in the correct format to be unit 1. The wrong disk unit may be attached at the location where the disk unit that contains storage unit 1 should be, or the correct disk unit that contains storage unit 1 may not be turned on (in which case the system detected the wrong disk unit as unit 1).

Continuing with licensed internal code install will delete all data on the disk unit that contains storage unit 1. The data on all the remainder of the disk devices will not be deleted, but may not be accessible (because of the way data is spread over multiple units on the system). This information can be used with the rack configuration list printouts to ensure the system will install or restore to the correct disk.

Function codes may be used to display to which disk the system is attempting to install. The function codes are different for systems with stage 2 hardware. The stage 2 hardware function codes are identified with a **-2** and are enclosed in parentheses (such as 15-2).

Function Code — *Function*

14 (15-2) Display type and model of the disk unit that contains storage unit 1. The first 4 characters displayed in the lights will show the type. The next 4 characters show the model.

The following is an example of what would be shown after you select this option. (This shows what you would see if the disk was a 9332 Model 400.)

14 (15-2)	9332 0400

15 (16-2) Display address of the disk unit that contains storage unit 1. Eight characters will be shown on the display lights. The meaning of these characters is as follows:

Character Position	***Description***
1−2	Bus number (should be zero)
3	IOP card number
4	IOP board number
5−6	Facility address (DFCI)
7−8	Secondary address

The following is an example of what would be shown after you select this option:

15 (16-2)	0010 0700

16 (17) Display the serial numbers of the disk unit that contain storage unit 1.

The following is an example of what should be shown after you select this option.

16 (17-2)	0012 34561

Reply Using the Function Select key on the control panel, type one of the following codes and press the Enter key:

Function Code	*Function*
24	Destroy all system data and restore the licensed internal code. Select function code 24 if you want to do the following: • Restore the licensed internal code to the selected disk unit. • Destroy all data on the system. Selecting this option will first delete all information found on the disk unit that contains storage unit 1 (including customer data) and then will copy all system licensed internal code from tape to disk. The data on all the remainder of the disk units will not be deleted but may not be accessible (because of the way data is spread over multiple units on the system). Select this option if you are installing a new system (which contains no customer data and no licensed internal code), or if you have exchanged unit 1.
(None)	Cancel this request. If you do not want to continue with the licensed internal code install operation, turn the control panel off.

Function 11, Data Code A6xx 6004

This topic describes the user input required by the licensed internal code SRC A6xx 6004.

Description **Warning:** The disk unit that contains storage unit 1 does not contain licensed internal code.

The stand-alone licensed internal code restore was requested. The system found a disk unit attached at the correct location for being the disk unit that contains storage unit 1. However, the disk unit found already contains data and the data is not in the correct format to be unit 1. The wrong disk unit may be attached at the location where unit 1 should be, or the correct disk unit that contains storage unit 1 may not be turned on (in which case the system detected the wrong disk unit as unit 1).

To restore licensed internal code to this disk unit, the system must first destroy all data on the disk unit that contains storage unit 1 (to correctly format it to be a disk unit that contains storage unit 1). The data on all the remainder of the disk units will not be deleted, but may not be accessible (because of the way data is spread over multiple units on the system).

Function codes may be used to display which disk the system is attempting to restore. This information can be used with the rack configuration list printouts to ensure the system will restore to the correct disk.

The function codes are different for systems with stage 2 hardware. The stage 2 hardware function codes are identified with a **-2** and are enclosed in parentheses (such as 15-2).

Function Code	*Function*
14 (15-2)	Display type and model of the disk unit that contains storage unit 1. The first 4 characters displayed in the lights will show the type. The next 4 characters show the model.

The following is an example of what would be shown after you select this option. (This shows what you would see if the disk was a 9332 Model 400.)

14 (15-2)	9332 0400

15 (16-2) Display address of the disk unit that contains storage unit 1. Eight characters will be shown on the display lights. The meaning of these characters are as follows:

Character Position	***Description***
1–2	Bus number (should be zero)
3	IOP card number
4	IOP board number
5–6	Facility address (DFCI)
7–8	Secondary address

The following is an example of what would be shown after you select this option:

15 (16-2)	0010 0700

16 (17-2) Display the serial numbers of the disk unit that contains storage unit 1.

The following is an example of what should be shown after you select this option:

16 (17-2)	0012 3456

Reply

Function Code	*Function*
24	Destroy all system data and restore the licensed internal code.

Select function code 24 if you want to do the following:

- Destroy all data on the system.
- Restore the licensed internal code to the selected disk unit.

Selecting this option will first delete all information found on the disk unit that contains storage unit 1 (including customer data) and then will copy all system licensed internal code from tape to disk. The data on all the remainder of the disk units will not be deleted but may not be accessible (because of the way data is spread over multiple units on the system).

Select this option if you are installing a new system (which contains no customer data and no licensed internal code), or if you have exchanged the disk unit that contains unit 1.

(None) Cancel this request.

If you do not want to continue with the licensed internal code restore operation, turn off the control panel.

Function 11, Data Code A6xx 6005

This topic describes the user input required for the licensed internal code SRC A6xx 6005.

Description Disk unit that contains storage unit 1 not found.

The disk unit that contains unit 1 cannot be located. Try the following:

- For 9335 devices, ensure that the Enable/Disable switch is in the Enable position.
- Ensure that all disk units are turned on. If the devices are already turned on, turn them off, wait a minute, then turn them back on.

If the disk unit that contains storage unit 1 still cannot be located, call for service support.

Function 11, Data Code A6xx 6006

This topic describes the user input required for the licensed internal code SRC A6xx 6006.

Description Tape licensed internal code file is incompatible for restoring.

The *load licensed internal code* level found on tape is not compatible with the *licensed internal code* found on disk. Continuing to restore this code will result in system failure.

Reply Using the Select key on the control panel, select one of the following codes, and press the Enter key:

Function Code	*Function*
24	Destroy all system data and install the licensed internal code. Select this option if you are installing a new system (which contains no customer data) or if you have exchanged the disk unit that contains storage unit 1.
(None)	Cancel this request. If you do not want to continue with the licensed internal code restore operation, turn off the system control panel.

Function 11, Data Code A6xx 6007

This topic describes the user input required for the licensed internal code SRC A6xx 6007.

Description

Warning: Disk unit that contains storage unit 1 does not contain licensed internal code.

The stand-alone download disk licensed internal code, function code 32, was requested. The system found the disk unit attached at the correct location for being a disk unit that contains storage unit 1. However, the disk unit found is not in the correct format to be a unit 1. The wrong disk unit may be attached at the location where unit 1 should be, or the correct disk unit that contains storage unit 1 may not be turned on (in which case the system detected the wrong disk unit as the disk unit that contains storage unit 1).

Function codes may be used to display which disk the system is attempting to restore. This information can be used with the rack configuration list printouts to ensure the system will restore to the correct disk.

The function codes are different for systems with stage 2 hardware. The stage 2 hardware function codes are identified with a **-2** and are enclosed in parentheses (such as 15-2).

Function Code	*Function*
14 (15-2)	Display type and model of the disk unit that contains storage unit 1. The first 4 characters displayed in the lights will show the type. The next 4 characters show the model.

The following is an example of what would be shown after you select this option. (This shows what you would see if the disk was a 9332 Model 400.)

14 (15-2)	9332 0400

Function Code	*Function*
15 (16-2)	Display address of the disk unit that contains storage unit 1. Eight characters will be shown on the display lights. The meaning of these characters is as follows:

Character Position	***Description***
1−2	Bus number (should be zero)
3	IOP card number
4	IOP board number
5−6	Facility address (DFCI)
7−8	Secondary address

The following is an example of what would be shown after you select this option:

15 (16-2)	0010 0000

16 (17) Display the serial numbers of the disk unit that contains storage unit 1.

The following is an example of what should be shown after you select this option:

16 (17-2)	0012 3456

Reply

Function Code	*Function*
24	Destroy all system data and restore the licensed internal code. Select function code 24 if you want to do the following: • Restore the licensed internal code to the selected disk unit. • Destroy all data on the system. Selecting this option will first delete all information found on the disk unit that contains storage unit 1 (including customer data) and then will copy all system licensed internal code from tape to disk. The data on all the remainder of the disk units will not be deleted but may not be accessible (because of the way data is spread over multiple units on the system). Select this option if you are installing a new system (which contains no customer data and no licensed internal code), or if you have exchanged the disk unit that contains storage unit 1.
32	Forced download of disk licensed internal code. Select function code 32 to force down load of the disk licensed internal code to the disk unit selected.
(None)	Cancel this request. If you do not want to continue with the licensed internal code restore operation, turn off the control panel.

Function 11, Data Code A6xx 6008

This topic describes the user input required for the licensed internal code SRC A6xx 6008.

Description **Warning:** Load-source disk unit not found.

The vital product data (VPD) within the system indicates that unit 1 will be found at the address specified in the following function codes. Either the disk unit at this address did not report or the device at that address is not a disk unit.

This condition can occur because of one of the following:

1. Either the disk unit at that address is not powered on, or did not report in, or is missing. Verify the following and make any necessary corrections. The install automatically continues when the device is ready and reports in.

- Ensure that all 9335 and 9336 disk units (if there are any) have their switches set to the Enable position.
- Ensure that the disk unit at that address is powered on and the Ready light is on. If the disk unit is already powered on and its indicator shows a ready status, you might try powering the disk unit off, waiting a few minutes, and powering the disk unit back on. The install automatically continues when the disk unit becomes ready and reports in.
- Ensure that any separate disk unit controller attached to the requested disk unit is powered on and ready. Do not power off the controller.

2. The AS/400 service processor has been replaced.

   The service processor card contains some of the VPD information being used. If the card has been replaced, then the VPD information will be incorrect.

   Function 27 can be used to override the VPD information and locate the default load-source disk unit.

   Warning: Read the instructions for function 27 below before taking this option.

3. The disk units on this system has been removed or cabled again, or their address switches have been changed. Ensure that the system is cabled correctly and that the disk units have the correct addresses. Notice that some disk units also contain VPD information about the system. If the units are not in the same positions, the VPD information will be incorrect.

   To locate storage unit 1, power down the system and move and cable the disk units again so that storage unit 1 and its mirrored unit have the correct address. Then run the stand-alone utility again.

   Function 27 can be used to override the VPD information and locate the default unit 1.

   Warning: Read the instructions for function 27 below before taking this option.

The function codes are different for systems with stage 2 hardware. The stage 2 hardware function codes are identified with a **-2** and are in parentheses (such as 15-2).

Function Code	*Function*
14 (15-2)	Displays the disk unit address of the candidate for storage unit 1. The location of the candidate for storage unit 1 is displayed.

Character Position	***Description***
1−2	Bus number (should be zero)
3	IOP card number
4	IOP board number

Appendixes

5–6 Facility address (DFCI)
7–8 Secondary address

The following is an example of what would be shown after you select this option:

14 (15-2)	0010 0000

15 (16-2) Display the serial numbers of the disk unit that contains storage unit 1. If the serial number of the disk unit is available, it is displayed. If it is not available, zeros are displayed.

The following is an example of what should be shown after you select this option:

15 (16-2)	0012 3456

If the system has mirrored protection, 16 (17-2) and 17 (18-2) will contain the information for the other unit in the mirrored pair, if the information is available.

Reply Press the Function Select key on the control panel until the following code appears and press the Enter key:

Function Code	*Function*
27	Ignore the VPD information. Select the default disk unit.

Using this option may result in the licensed internal code being loaded to the wrong disk unit.

Note: Before selecting this option, you should unload the tape, and IPL the system from disk (either A or B side) with the keylock switch in the Manual position. This action should correct the VPD information or displays are shown to indicate the problem and recommend actions. If the IPL of the system cannot be started from disk, then select this function.

Select function 27 if you are unable to bring the requested disk unit online. This causes the SAU to select the default disk unit to be unit 1. On high-end systems, this is the disk unit whose address is 0010000 or 00200000. On low-end system, this is the disk unit at address 00100100.

Note: On low-end stage 1 systems, the disk unit must be in the bottom slot of the first tower. On low-end stage 2 systems, this restriction does not apply.

In some situations, the VPD information found on the default disk unit may be used to locate the correct unit 1.

(None) Cancel this request.

If you do not want to continue installing the licensed internal code, unload the tape and either power off the system or perform an IPL of the system from disk (either A or B side)

Function 11, Data Code A6xx 6009

This topic describes the user input required for the licensed internal code SRC A6xx 6009.

Description **Warning:** Mirrored load-source disk unit not found.

The vital product data (VPD) within the system indicates unit 1 will be found at the address specified in the following function codes. Either the disk unit at this address did not report or the device at that address is not a disk unit.

This condition can occur because of one of the following:

1. Either the disk unit at that address is not powered on, or did not report in, or is missing. Verify the following and make any necessary corrections. The install automatically continue when the device is ready and reports in.
   - Ensure all 9335 and 9336 disk units (if there are any) have their switches set to the Enable position.
   - Ensure that the disk unit at that address is powered on and the Ready light is on. If the disk unit is already powered on and its indicator shows a ready status, you might try powering the disk unit off, waiting a few minutes, and powering the disk unit back on. The install automatically continues when the disk unit becomes ready and reports in.
   - Ensure that any separate disk unit controller attached to the requested disk unit is powered on and ready. Do not power off the controller.
2. The AS/400 service processor has been replaced.

   The service processor card contains some of the VPD information being used. If the card has been replaced, then the VPD information will be incorrect.

   Function 27 can be used to override the VPD information and locate the default unit 1. **Caution: Read the instructions for function 27 below before taking this option.**
3. The disk units on this system has been removed or cabled again, or their address switches have been changed. Ensure that the system is cabled correctly and that the disk units have the correct addresses. Notice that some disk units also contain VPD information about the system. If the units are not in the same positions, this VPD information will be incorrect.

   To locate unit 1, power down the system and move and cable the disk units again so that the load-source disk unit and its mirrored unit have the correct address. The run the stand-alone utility again.

Function 27 can be used to override the VPD information and locate the default storage unit 1. **Caution: Read the instructions for function 27 below before taking this option.**

The function codes are different for systems with stage 2 hardware. The stage 2 hardware function codes are identified with a **-2** and are in parentheses (such as 15-2).

Function Code	*Function*
14 (15-2)	Displays the disk unit address of the mirrored storage unit 1.

The location of the mirrored disk unit 1 is displayed.

Character Position	***Description***
1−2	Bus number (should be zero)
3	IOP card number
4	IOP board number
5−6	Facility address (DFCI)
7−8	Secondary address

The following is an example of what would be shown after you select this option:

14 (15-2)	0010 0000

15 (16-2) Display the serial number of the disk unit that contains mirrored unit for storage unit 1. If the serial number of the disk unit is available, it is displayed. If it is not available, zeros are displayed.

The following is an example of what should be shown after you select this option:

15 (16-2)	0012 3456

If the system has mirrored protection, 16 (17-2) and 17 (18-2) will contain the information for the other unit in the mirrored pair, if the information is available.

Reply

Press the Function Select key on the control panel until the following code appears and press the Enter key:

Function Code	*Function*
27	Ignore the VPD information. Select the default disk unit.

Warning: Using this option may result in the licensed internal code being loaded to the wrong disk unit.

Note: Before selecting this option, you should unload the tape, and IPL the system from disk (either A or B side) with the keylock switch in the manual position. This action should correct the VPD information or displays are shown to indicate the problem and recommend actions. If the IPL of the system cannot be started from disk, then select this function.

Select function 27 if you are unable to bring the requested disk unit online. This causes the install to continue without the mirrored unit for storage unit 1.

(None) Cancel this request.

If you do not want to continue installing the licensed internal code, unload the tape and either power off the system or perform an IPL of the system from disk (either A or B side).

Function 11, Data Code A6xx 6010

This topic discusses the user input for licensed internal code SRC A6xx 6010.

Description **Warning:** Load-source disk unit not found.

The disk unit for storage unit 1, specified in the following function codes, indicates that it is a part of a mirrored pair. Because of the available information, the stand alone utility is unable to verify that the disk unit selected is in the correct mirrored state.

Warning: Continuing this procedure may cause the licensed internal code to be loaded to a suspended or resuming mirrored pair.

Either the vital product data (VPD) within the system indicates that the disk unit specified in the following function codes is unit 1 or a request was made to override the VPD by previously entering function 27 in response to SRCs A6xx 6008 or A6xx 6009.

This condition can occur because of one of the following:

1. The AS/400 service processor has been replaced.

   The service processor card contains some of the VPD information being used. If the card has been replaced, then the VPD information will be incorrect.

2. The disk units on this system has been removed or cabled again, or their address switches have been changed. Ensure that the system is cabled correctly and that the disk units have the correct addresses. Notice that some disk units also contain VPD information about the system. If the units are not in the same positions, this VPD information will be incorrect.

3. A request to override the VPD information was previously done and the default disk unit location was selected.

The function codes are different for systems with stage 2 hardware. The stage 2 hardware function codes are identified with a **-2** and are in parentheses (such as 15-2).

Function Code	*Function*
14 (15-2)	Displays the disk unit address of the candidate for storage unit 1. The location of the candidate for storage unit 1 is displayed.

Appendixes

Character Position	*Description*
1–2	Bus number (should be zero)
3	IOP card number
4	IOP board number
5–6	Facility address (DFCI)
7–8	Secondary address

The following is an example of what would be shown after you select this option:

14 (15-2)	0010 0000

15 (16-2) Display the serial number of the disk unit that contains storage unit 1. If the serial number of the disk unit is available, it is displayed. If it is not available, zeros are displayed.

The following is an example of what should be shown after you select this option:

15 (16-2)	0012 3456

If the system has mirrored protection, 16 (17-2) and 17 (18-2) will contain the information for the other unit in the mirrored pair, if the information is available.

Reply Press the Function Select key on the control panel until the following code appears and press the Enter key:

Note: Before selecting this option, you should unload the tape, and perform an IPL of the system from disk (either A or B side) with the keylock switch in the manual position. This action should correct the VPD information or displays are shown to indicate the problem and recommend actions. If the system cannot perform an IPL from disk, then select this function.

Function Code	*Function*
27	Ignore this warning and continue the install. To continue installing the licensed internal code, press the Function Select key until 27 appears. Use the function codes to display the disk unit where the licensed internal code will be installed.
(None)	Cancel this request. If you do not want to continue installing the licensed internal code, power off the control panel.

Function 11, Data Code A6xx 6011

This topic discusses the user input for licensed internal code SRC A6xx 6011.

Description Verify selection to load model-unique licensed internal code.

The Load Model-Unique Licensed Internal Code Utility was selected. Continuing this procedure will copy the model-unique licensed internal code from tape to disk. Any existing model-unique licensed internal code existing on disk will be exchanged.

Using the Function Selection switch on the control panel, select one of the following codes and press the Enter button on the control panel.

Function Code	*Function*
23	Load the model unique licensed internal code. Press the Function Select key until 23 appears to continue exchanging the model-unique licensed internal code disk with the model-unique licensed internal code on tape.
29	Cancel this request. Select option 29 if you do not want to load the model-unique licensed internal code. After selecting this option, SRC A6xx 6001 occurs again.

Function 11, Data Code A6xx 6030

This topic describes the user input required for the licensed internal code SRC A6xx 6030.

Description Primary disk unit is not ready or not operational.

Reply Prepare the disk unit. See the *System Operator's Guide*, SC41-8082, if you need more information on how to prepare the disk.

Function 11, Data Code A6xx 6041

This topic describes the user input required for the licensed internal code SRC A6xx 6041.

Description Tape device is not operational.

Reply See the *System Operator's Guide*, SC41-8082, for the specific type of tape unit and for an explanation of how to make the tape device operational.

Function 11, Data Code A6xx 6042

This topic describes the user input required for the licensed internal code SRC A6xx 6042.

Description Tape device is not ready.

The 3rd and 4th characters in the SRC give further information on why the tape unit is not ready.

Code	*Definition*
33	Load assistance
37	Cartridge length check

3B	Volume removed early
43	Tape not placed in tape unit
FE	Tape unit turned off
FF	Tape unit turned off

Reply Prepare the tape device.

See the *System Operator's Guide*, SC41-8082, for the specific type of tape device and for an explanation of how to prepare the tape device.

Function 11, Data Code A6xx 6043

This topic describes the user input required for the licensed internal code SRC A6xx 6043.

Description Tape device load failure occurred.

The 3rd and 4th characters in the SRC give further information on why a load failure occurred.

Code	*Definition*
11	Door open
12	Reel missing
13	Reel inverted
14	No BOT marker
16	Load failure
1B	Address switch changed
1C	Address switch failure

Reply Prepare the tape device.

For an explanation of how to make the tape device ready, see the device operator's guide that came with your tape unit.

Function 11, Data Code A6xx 6048

This topic describes the user input required for the licensed internal code SRC A6xx 6048.

Description New tape volume needs to be installed.

Reply Insert tape volume, and then prepare the tape device. After the tape is loaded and ready, the system will start reading this tape and continue with the install (or restore) process.

The 3rd and 4th characters in the SRC tell which volume should be loaded.

Code	*Definition*
01	Volume 1
02	Volume 2
03	Volume 3
.	
.	
.	
nn	Volume nn

For an explanation of how to insert a tape, see the device operator's guide that came with your tape unit.

Function 11, Data Code A6xx 6049

This topic describes the user input required for the licensed internal code SRC A6xx 6049.

Description Insert correct tape volume.

Reply You already inserted a new tape volume, but it was not the correct volume.

Code	*Definition*
01	Volume 1
02	Volume 2
03	Volume 3
.	
.	
.	
nn	Volume nn

For an explanation of how to install a tape, see the device operator's guide that came with your tape unit.

Function 11, Data Code A6xx 6051

This topic describes the user input required for the model-unique licensed internal code SRC A6xx 6051.

Description Load tape volume containing model-unique licensed internal code located in the system service kit.

Reply Insert the tape volume containing the model-unique licensed internal code into the alternative IPL tape device, and make the tape device ready. After the tape is loaded and the device is ready, the system will start reading this tape and load the model-unique licensed internal code.

Function 11, Data Code A6xx 6052

This topic describes the user input required for the model-unique licensed internal code SRC A6xx 6052.

Description The inserted tape did not contain the model-unique licensed internal code.

Reply Place the model-unique licensed internal code tape into the tape device used for the IPL, and prepare the tape device.

Appendix B. Tips and Techniques

Commands Available in QUSRTOOL Library

QUSRTOOL library contains CL source for commands you can create to help verify a backup and recovery design.

Save While Active (SAVWHLACT) Command in QUSRTOOL Library

The Save While Active (SAVWHLACT) command allow a data base file to be saved while it is active. Being active means that the file is open for update and one or more programs are making changes while the save occurs. The companion command Restore while Active (RSTWHLACT) does a restore operation from the copy made by SAVWHLACT command. Then, it applies changes which occurred during the copy. These commands are specifically designed for environments in which the file must be available for update around the clock. When this type of situation exists, the business typically requires a window of time in the early hours of the morning to be used for backup when users must signoff or be kept out of an application area. The SAVWHLACT command allows you to reduce the save window and make a file available all of the time.

The SAVWHLACT command does not save the file offline or to a save file. It makes a copy of the file that along with the journal receiver can be saved to produce an offline backup copy of the file.

Save All Change (SAVALLCHG) Command in QUSRTOOL Library

The Save All Change (SAVALLCHG) command does a SAVCHGOBJ against all libraries on the system to tape and prints a summary listing. This allows a good daily backup for many systems when used in conjunction with a periodic SAVLIB(*NONSYS or *IBM) and will help reduce your dependency on the SAVLIB(*NONSYS) command.

Any libraries which begin with the letter 'Q' are implicitly bypassed except for QGPL and QUSRSYS. The reason for this is that PTFs may be applied to IBM libraries which contain code and this would cause objects to be saved. However, the PTF index is only saved only by a SAVLIB command. Consequently, a restore of SAVCHGOBJ media would make the library and the index out of sync. You should retain PTFs until the next full library save. The IBM supplied '#' libraries (except for #LIB) which are part of the S/36 Environment are also bypassed for the same reason.

You may specify up to 100 libraries to be bypassed (not saved) by using the BYPASS parameter on the SAVALLCHG commands. This allows you to tailor your save strategy if you have some unique processing for certain libraries. For example, you may prefer to save the entire contents of some libraries and then use SAVALLCHG on the rest of the libraries.

A list is printed with one line for each library saved. The list describes the number of objects saved, not saved (some problem occurred such as an update lock exists on the object), excluded (based on the reference date/time), the create date of the library (if it is greater than the reference date) and the volume identification of where the objects were saved. If a library is bypassed, it is listed along with an indication that it was bypassed. Any bypassed libraries will not necessarily appear in alphabetical sequence.

Save All Changes 2 (SAVALLCHG2) Command in QUSRTOOL

This command is similar to SAVALLCHG command, except the library is saved to a save file.

Save All Save Files (SAVALLSAVF) Command in QUSRTOOL Library

The Save All Save Files (SAVALLSAVF) command is used to save to tape all SAVALLCHG2 created save files.

Save Spool Control (SAVSPLCTL) Command in QUSRTOOL Library

The Save Spool Control (SAVSPLCTL) command is part of the QUSRTOOL spool function, SPLCTL. The spool control function provides a method of backing up, retrieving, and reprinting spooled files.

In some environments, the end users expect the data processing department to be able to print a report again that was done several days ago. The data used for the report may have changed significantly and it may not be practical to re-create the data. The spool control function lets you save the spooled file contents in the event re-printing is anticipated. You can also use the technique for job logs or other spooled output which may only need to be printed on request.

You specify by output queue whether you want the spooled files saved. The spool control function then copies the spooled files to a physical file (using CPYSPLF), saves the physical file to an online save file and allows backup of the save file. A retrieval function is available which will reprint the file.

Commands to Verify a Backup Design

Verify what objects are being saved, and when, by doing the following:

- Locate objects not being saved
- Review the save and restore job logs
- Find out when and where libraries are saved

How to Locate Objects That are Not Being Saved

Your best chance of avoiding a recovery exposure is to know exactly what you are saving (or perhaps more importantly, what you are not saving). Although you may save objects weekly, if the objects are changed daily, you may not be in a position to recover the changes.

To prevent such an exposure, you should periodically verify your backup strategy by reviewing when objects and changes made to these object are being saved. This can be done by using the Check Save (CHKSAV) command in library QUSRTOOL.

The CHKSAV command allows you to determine whether one or more libraries have had any objects or members changed since the last save operation.

CHKSAV can be used on any or all libraries except library QSYS. When library QSYS is saved using the SAVSYS command, individual object descriptions are not updated. Instead, a SAVSYS updates the object description of two data areas named QSAVSYS and QSAVUSRPRF in library QSYS.

To determine when a SAVSYS command was run, use the DSPOBJD command to display the object description of these two data areas (SAVSECDTA updates the object description of the data area QSAVUSRPRF only). You do not display the data area itself. You display the object description of the data area using the DSPOBJD or WRKOBJD command.

Once you determine that all objects and changes are being saved to your satisfaction, consider using the CHKSAV command on a periodic basis, perhaps monthly, semi-annually, or yearly. As new applications are placed on the system, or as existing applications are changed, use the CHKSAV command to make sure these additions or changes have not created an exposure in your backup and recovery strategy.

How to Review the Save and Restore Joblogs

It is also important to make sure that the save command completes successfully. This can be done by reading through the joblog created during the save. However, this can be a tedious task for large save operations. To help automate this function, the Check Save Status (CHKSAVSTS) command in library QUSRTOOL can be used.

The CHKSAVSTS command reads the joblog that includes the save or restore commands and prints a list of completion messages for that save operation, as well as any diagnostic messages found. The CHKSAVSTS command basically summarizes the joblog used during the save or restore process and nets out what is important from a save or restore viewpoint. CHKSAVSTS command allows you to direct the output to either a printer or a database file.

How to Find Out When and Where Libraries Are Saved

Using the Display Object Description (DSPOBJD) command against a specific library can tell when and where the library was saved (date, time, and what tape volume). The DSPOBJD command works for single libraries. However, to help verify your backup and recovery strategy, you may want to look at multiple libraries. This can be accomplished by using the Print Save Status (PRTSAVSTS) command in library QUSRTOOL.

The PRTSAVSTS command is intended for use following a save of multiple libraries. The PRTSAVSTS command creates printed output that contains a description of one or more library names, save date, save command, and the volumes the libraries are saved to. This list should then be saved to tape in the event a restore is necessary. An external description exists for the libraries that were saved and what tape volumes contain the libraries.

Verify Integrity of the Save or Restore Tapes

The time spent saving an object is wasted if the tape used for the save operation is unusable. All tapes are subject to wear over a period of time. To help monitor the quality of your tapes, use the *Work with tape or diskette statistics* option accessed using the System Service Tools functions. This shows you the number of temporary and permanent read or write errors that have occurred for a particular tape volume as well as the total amount of data read or written to the tape.

Programming Examples

This section provides some example programs used for save and restore operations, for commitment control, and for controlling an uninterruptible power supply.

Retrieving the Device Name from Save Completion Messages

The CL program in Figure B-1 retrieves the device name from the CPC3701 message (found in positions 126 through 135 of the message data) and uses the information to determine which device is used by the next save command.

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7

               PGM
               DCL       &MSGDTA *CHAR LEN(150)
               DCL       &MSGID *CHAR LEN(7)
               DCL       &DEV *CHAR LEN(10)
               DCL       &DFV1 *CHAR LEN(10) VALUE(QTAPE1)
               DCL       &DEV2 *CHAR LEN(10) VALUE(QTAPE2)
               SAVLIB    LIB(LIB1) DEV(&DEV1 &DEV2) ENDOPT(*LEAVE)
     LOOP:     RCVMSG    RMV(*NO) MSGDTA(&MSGDTA) MSGID(&MSGID)
               IF        (&MSGID *NE CPC3701) GOTO LOOP /* Compltn */
               CHGVAR    &DEV %SST(&MSGDTA 126 10) /* Device name */
               IF        (&DEV *EQ 'QTAPE1') DO /* Last was QTAPE1 */
               CHGVAR    &DEV1 'QTAPE1' /* Set for first device */
               CHGVAR    &DEV2 'QTAPE2' /* Set for second device */
               ENDDO     /* Last was QTAPE1 */
               ELSE      DO /* Last was not QTAPE1 */
               CHGVAR    &DEV1 'QTAPE2' /* Set for first device */
               CHGVAR    &DEV2 'QTAPE1' /* Set for second device */
               ENDDO     /* Last was not QTAPE1 */
               SAVLIB    LIB(LIB2) DEV(&DEV1 &DEV2) /* Save Lib 2 */
               ENDPGM
```

RV2W354-1

Figure B-1. Example CL Program

If any objects cannot be saved, the operation attempts to save remaining objects and sends an escape message (CPF3771 for single libraries, CPF3751/CPF3778 for more than one library, and CPF3701 for save operations to save files) stating how many objects were saved and how many were not. To continue with the next library, the Monitor Message (MONMSG) command must be used to handle the escape condition. The format of the message data for the CPF3771 message is similar to the CPC3701 message and also identifies the last device used.

The SAVCHGOBJ command operates in a similar manner, but uses CPC3704 as a completion message, CPF3774 as an escape message for single libraries, and CPC3721 or CPF3751 for multiple libraries. For save operations to save files, these messages are CPC3723 as a completion message and CPF3702 as an escape message. These messages also contain the last device or save file used in the message data.

Using a Status Program for Saving Source Files

The following program sends a message to the external (*EXT) program message queue if all the source files cannot be saved. This same technique can be used for other objects.

```
PGM            /* SAVE SOURCE */
SAVLIB         LIB(SRCLIB) DEV(TAPE1) PRECHK(*YES)
MONMSG         MSGID(CPF0000) EXEC(DO)

SNDPGMMSG      MSG('Objects were not saved - Look at the job +
               log for messages') TOPGMQ(*EXT)
SNDPGMMSG      MSG('Objects were not saved - Look at the job +
               log for messages') TOPGMQ(*EXT)
SNDPGMMSG      MSG('SRCLIB library was not backed up') +
               TOPGMQ(xxxx)
RETURN
ENDDO
SNDPGMMSG
ENDPGM
```

Using the Retrieve Journal Entry (RTVJRNE) Command in a Program

Use the Retrieve Journal Entry (RTVJRNE) command in a control language program to retrieve a journal entry and place it in variables in the program. You can retrieve the following:

- Sequence number
- Journal code
- Entry type
- Journal receiver name
- Library name for the journal receiver
- Journal entry

For example, you can use this command to automate your recovery procedures or to change the journal receivers and then save them.

In the following example, the RTVJRNE command determines when job 000666/QPGMR/WORKST01 last opened file ORDENTP:

```
PGM
DCL &SEQ# TYPE(*DEC) LEN(10 0)
DCL &JRNE TYPE(*CHAR) LEN(200)
DCL &DATE TYPE(*CHAR) LEN(6)
DCL &TIME TYPE(*CHAR) LEN(6)
RTVJRNE JRN(DSTJRN/JRNLA) FILE(DSTPRODLIB/ORDENTP) +
        RCVRNG(DSTJRN/RCV30 DSTJRN/RCV27) FROMENT (*LAST) +
        TOENT(*FIRST) SEARCH(*DESCEND) +
        JRNCDE(F) ENTTYP(OP) JOB(000666/QPGMR/WORKST01) +
        RTNSEQNBR(&SEQ#) RTNJRNE(&JRNE)
CHGVAR &DATE (%SST(&JRNE 19 6))
CHGVAR &TIME (%SST(&JRNE 25 6))
ENDPGM
```

The *CL Reference* manual has a description of the received journal entry format.

CL Program to Handle Escape Conditions

The WRKJRN command offers recovery functions that can perform file recovery. Figure B-2 on page B-6 demonstrates how this escape condition can be handled in the CL program by prompting for restoration of the required receiver.

```
FILERECOV: PGM
           .
           .
           APYJRNCHG JRN(JRNLIB/JRNA) FILE((LIBA/FILEA)) +
                 RCVRNG(RCVLIB/RCV1 *CURRENT)
           MONMSG MSGID(CPF7053 CPF9801) +
              EXEC(CALL PGM(FIXLIB/RSTRCV) PARM(FILERECOV))
           .
           .
           ENDPGM
           .
           .
RSTRCV: PGM PARM(&PGMNM)
        /* Recover a nonexistent or unusable receiver       */
        /* in RCVRNG by prompting for a restore of          */
        /* receiver.
DCL &PGMNM TYPE(*CHAR) LEN(10)        /* name of program    */
                                      /* calling RSTRCV     */
                                      /* that received      */
                                      /* CPF7053 or         */
                                      /* CPF9801            */
DCL &MSGDTA TYPE(*CHAR) LEN(22) /* variable for   */
                                      /* CPF7053 or         */
                                      /* CPF9801            */
DCL &MSGDID TYPE(*CHAR) LEN(7) /* escape message  */
                                      /* ID                 */
DCL &RCVNAME TYPE(*CHAR) LEN(10) /* name of       */
                                      /* receiver to        */
                                      /* restore            */
DCL &RCVLIB TYPE(*CHAR) LEN(10) /* library name   */
                                 /* of receiver to          */
                                 /* restore                 */
DCL &RCODE TYPE(*CHAR) LEN(2) VALUE(x'0001')
                        /* reason code 1 of CPF7053         */
RCVMSG PGMQ(*SAME &PGMNM) MSGTYPE(*EXCP) WAIT(0) +
   RMV(*NO) MSGDTA(&MSGDTA) MSGID(&MSGID)
```

Figure B-2 (Part 1 of 2). Example Program Prompts for Restoring the Required Receiver

```
IF COND(&MSGID *EQ 'CPF9801') THEN(DO)    /* CPF9801 occurred */
   CHGVAR &RCVNAME (%SST(&MSGDTA 1 10))   /* get receiver     */
                                          /* from message     */
                                          /* data             */
   CHGVAR &RCVLIB (%SST(&MSGDTA 11 10))   /* get library      */
                                          /* name from        */
                                          /* message data     */
   ? RSTOBJ OBJ(&RCVNAME) SAVLIB(&RCVLIB) OBJTYPE(*JRNRCV)
                                    /* display RSTOBJ prompt */
  ENDDO
ELSE DO
IF COND((&MSGID *EQ 'CPF7053') & (%SST(&MSGDTA 1 2) +
   *EQ &RCODE)) THEN(DO)  /*CPF7053 RC(1) occurred*/
   CHGVAR &RCVNAME (%SST(&MSGDTA 3 10))   /* get receiver     */
                                          /* name from        */
                                          /* message data     */
   CHGVAR &RCVLIB (%SST(&MSGDTA 13 10))   /* get library      */
                                          /* name from        */
                                          /* message data     */
     ? RSTOBJ OBJ(&RCVNAME) SAVLIB(&RCVLIB) OBJTYPE(*JRNRCV)
                                   /* display restore prompt */
    ENDDO
  ELSE
.
.
 ENDDO
ENDPGM
```

Figure B-2 (Part 2 of 2). Example Program Prompts for Restoring the Required Receiver

Using an Application Program to Apply Journal Changes

The APYJRNCHG command has three restrictions:

- Journal entries can be applied only to the file and member being journaled.
- If a restore occurs, the object restored must have been journaling at the time of the save in order to apply journal entries.
- Journal entries can be applied from a journal receiver and not be converted (for example, by the DSPJRN or RCVJRNE commands).

In some recovery situations and applications you may need to bypass these restrictions by using the DSPJRN (or alternatively, RCVJRNE or RTVJRNE) command to extract entries from the journal and direct them to a database file. You then use an application program to apply the journaled changes to files.

The APYUSRCHG tool in library QUSRTOOL provides an example of how to apply journal entries from the output created by the Display Journal (DSPJRN) command.

Writing Output Using the Receive Journal Entry Command

The following example shows an RPG program that is being used as the exit program for the Receive Journal Entry (RCVJRNE) command. (See the discussion of the RCVJRNE command under "Journal Entries" on page 12-12.) The sample program in Figure B-3 on page B-10 writes output onto tape. See "Differences for Writing to an ICF File" on page B-9 for a discussion of changing the sample to write output to an OS/400-ICF file. You need a user-written application program to apply the entries. (See the sample program in "Using an Application Program to Apply Journal Changes.")

Writing to Tape

A separate job must be in continuous operation and dedicated to converting entries to tape. Before issuing RCVJRNE the command, your job should issue an OVRTAPF command, specifying fixed-length blocked records, to direct the RPG file TAPE to a tape device.

You should not consider this approach with a streaming tape device. A user auxiliary storage pool (ASP) is a preferred solution instead of a tape. However, this approach is similar to writing journal entries to a communications line.

The RPG program is written assuming that the largest journal entry to be passed is 300 bytes. This is the size given to the data structure JRNENT. It allows a record size of 175 bytes plus the 125 bytes of journal entry identifier and qualifier information. A check is made in the program to ensure that the record image is not being truncated:

- If a code of 1 is passed from the RCVJRNE command, the program ensures that the journal entry does not exceed 300 bytes. If it does. the program sets on the H1 indicator and returns. The program adds 1 to the counter and writes the record to the tape output file. Because this is an output-only file, RPG automatically blocks the records within the RPG program.

  When full, the block is passed to tape data management, where additional blocking can occur and double-buffering to the tape device is provided. This ensures that the tape performance is optimal. Because the records are not written directly to tape when the program requests the output, there can be some interval of time before the records are written to the external media.

- When a code of 0 is passed from the RCVJRNE command, no more entries exist in the journal. On the return to the RCVJRNE command, the DELAY parameter value specified on the RCVJRNE command is used to wait before checking for additional entries. To avoid keeping the records in the various buffers while the delay occurs, the program forces the records to the tape device by using the force end-of-data operation (FEOD).

  This causes all records in either the RPG or tape data management buffers to be written to the tape device, and a device completion notification to be received before proceeding to the next instruction. If there is less than a full block of records, a short block is written to tape. Tape data management correctly handles the short block if the tape is read in a subsequent program. When the return occurs to the RCVJRNE command, the delay time occurs whether or not any journal entries have arrived since the last time the exit program was called.

The RPG program increments a counter each time a record is written and resets it when the FEOD operation is used. The program issues the FEOD operation only if a record has been written which avoids calling tape data management when there are no records to be written. (If tape data management has no records in its buffers when the FEOD operation occurs, no empty block is written, but system overhead occurs.)

The RPG program uses the SHTDN operation code to check for requests to end the job from external functions such as an End Job (ENDJOB) or End Subsystem (ENDSBS) command with OPTION(*CNTRLD). If end-of-job is requested, the program forces the records from the buffers, sets the counter to 9 (which tells the RCVJRNE command to complete normally, and sets the LR indicator on). The RETRN operation is then issued and:

- If LR is on, the program's working storage is returned to the system.
- If LR is off, the program remains active and waits to be called again by the RCVJRNE command.

Writing to tape occurs either by the buffers being full or when the FEOD operation is used. This trade-off allows good performance when many journal entries are written and minimizes the number of times the FEOD operation is used to ensure that the entries are actually on the tape. With the sample program, the value of the DELAY parameter and the work management specifications for your job (for example, pool size and priority) are the major factors controlling the frequency with which the entries are written and the performance implications on the system for this function.

If the system ends abnormally while the job is running, so that a successful end-of-file indication is not written, the subsequent reading of the tape can produce results that cannot be predicted. Blocks that were successfully written can be correctly read. The last block and any subsequent data that is on the tape from a previous use can produce results that cannot be predicted. Copy the tape to a database file and examine the contents before using the data.

The journal sequence numbers are in ascending sequence (unless they have been reset) and can be used to determine where the logical end-of-file exists. To avoid confusion, delete the tapes used for this type of approach.

Assume, for example, that the largest record size journaled was 175 bytes and the tape record size 300 bytes, as in Figure B-3 on page B-10. If you need to increase the tape record size, change the value of 300 in the RPG file description specification, the input specification, and factor 2 of the CABGT operation code. If there are some significantly larger records being journaled, consider how much excess media is used. An alternative would be to examine the individual fields (*JOENTL*) and write two or more small records for each large record.

Differences for Writing to an ICF File

If you use an ICF file to transmit journal entries to another system, the FEOD operation does not apply. Instead, there are data description specifications (DDS) words (for example, FRCDTA) to force records from the buffers.

Usually the number of blocks transmitted to tape by records less than 175 bytes is a minimal performance consideration. On communications lines, however, this number can be significant. To avoid sending unnecessary trailing blanks, consider decreasing the length of the record being transmitted by the variable length function (VARLEN DDS keyword). For a discussion of the variable length function, see the *ICF Programmer's Guide*.

If binary synchronous equivalence link (BSCEL) is used, trailing blanks will be truncated automatically if the TRUNC parameter is specified on the Add ICF Device Entry (ADDICFDEVE) or the Override ICF Device Entry (OVRICFDEVE) command. Refer to the *BSC Equivalence Link Programmer's Guide* for more information about the function of the TRUNC parameter.

Figure B-3 contains the RPG source for your program:

```
SEQNBR *... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7

       FTAPE    O   F     300            SEQ
       IJRNENT      DS                            300
       I                                        1   50JOENTL
       C              *ENTRY    PLIST
       C                        PARM          JRNENT
       C                        PARM          CALLCD  1
       C              CALLCD    IFEQ '1'                      Entry rcvd
       C* Ensure journal entry is not being truncated
       C              JOENTL    CABGT300      RETURN      H1  If GT output
       C                        ADD  1        OUTRCD  70      Bump ctr
       C                        EXCPTOUTPUT                   Output
       C                        END                           Entry rcvd
       C              CALLCD    IFEQ '0'                      Rdy to wait
       C                        EXSR FORCE                    Force out
       C                        END                           Rdy to wait
       C                        SHTDN                     31  Test shtdwn
       C   31                   DO                            If shtdwn
       C                        EXSR FORCE                    Force out
       C                        MOVE '9'      CALLCD          Set to end
       C                        SETON                     LR  Set LR
       C                        END                           If shtdwn
       C              RETURN    TAG                           Return tag
       C                        RETRN                         Return
       C              FORCE     BEGSR                         Force out
       C              OUTRCD    IFNE *ZERO                    If rcds
       C                        FEOD TAPE                     FEOD
       C                        Z-ADDO        OUTRCD          Reset
       C                        END                           If rcds
       C                        ENDSR                         End subr
       OTAPE     E                  OUTPUT
       O                            JRNENT
```

RV2W357-0

Figure B-3. RPG Source for Writing Output

Appendix C. Example Disaster Recovery Plan

Section 1. Disaster Recovery Introduction and Overview

NOTICE

This plan will be reviewed for update every ____________ months by ______________________________ .

A copy of this plan will remain at the business site. The original of the plan is stored off site at ____________.

The continued growth in dependence on computers and on the information from them has increased the importance of the plans to prevent loss of their availability. It is becoming very difficult, costly and confusing to revert back to manual systems for any length of time. A disaster recover plan is necessary to minimize the damage that might be caused by unexpected and undesirable occurrences affecting the Information Services or other parts of the organizations information systems.

The objectives of a disaster recovery plan for information services are to make sufficient preparations, and to establish a sufficient set of agreed upon procedures, for responding to a disaster or emergency in the Information Services of responsibility, in order to minimize the effect upon the operation of the business.

Need for a Disaster Recovery Plan:

Three areas need to be reviewed: legal responsibility, financial loss, and business service interruptions.

Legal Responsibility: Management has a legal responsibility to protect its corporate resources and vital documents.

Financial Loss: Because of the efficiency, accuracy, speed and control of information services methods, organizations are more dependent on their information services in normal business operations. If the information systems services break down, a great financial loss to the company could develop, or even destroy the business if proper disaster planning has not been done.

Business Service Interruption: This can be very damaging to future relationships with customers. It can also affect the public image of the organization. The costs of not taking precautions could be much more damaging and costly than modest preparation for disaster recovery.

Levels of Security and Disaster Recovery Measures

This disaster plan is developed to minimize the costs resulting from losses or damages to the resources or the capabilities of information services and related services. The resources include people, programs, data, hardware, communication equipment and systems, electric power, physical facility and associated supplies such as forms, printer paper, and so forth.

There are three levels of security and disaster recovery measures that should be considered in balancing cost to need. These are: mandatory, necessary, desirable.

Mandatory Measures: Are those related to fire control, alarm systems, evacuation procedures, and other emergency precautions necessary to protect the lives and well-being of people in the area. They also include those measures necessary to protect the assets of the company.

Necessary Measures: Include all reasonable precautions taken to prevent serious disruptions of the operation of the business.

Desirable Measures: Include reasonable precautions taken to prevent real inconvenience or disruptions to any area of business and to keep the business under smooth control.

Types of Disasters to Consider

Natural Disasters: Storms, earthquakes, hurricanes, and tornadoes.

Man-Made Disasters: Fire, accident, theft, willful destruction, sabotage, burst pipes, building collapse, bomb threat, plane crash, riots, civil disturbance, strike, and war.

In any case, the maintenance of vital information and data must be kept in a secure place. Realistically, we must concentrate on the most probable types of disaster.

There are five types of disasters which information services could generally be vulnerable.

1. Damage to separate workstation areas - localized situation in a terminal area might require some readjustment of communication and establishment of new workstation areas.
2. Localized damage to information services offices - might affect schedules, development work, etc.
3. Damage to information services - substantial damage done as a result of fire, water leak, or airplane crash.
4. Substantial damage to business offices or manufacturing.
5. Regional damage in a broad area - storms, hurricane, or industrial accident. There could be widespread damage, loss of power, communication lines, and so forth.

Management Direction

Management should provide the following direction for the disaster recovery plan.

- Direct the establishment of disaster contingency plans.
- Direct support of the planning process by all functional areas serviced by the information services facility. In particular, identify those functions which are critically dependent upon the information services facility.
- Direct initial and subsequent tests of the workability of the plan and of costs associated with it.
- Direct periodic revision of the plan as required.

Major Goals of the Plan

The major goals of this plan are:

1. To minimize interruptions to the normal operations.
2. To limit the extent of disruption and damage.
3. To provide smooth degradation.
4. To minimize the economic impact of the interruption.
5. To establish alternative means of operation in advance.
6. To train personnel with emergency procedures.
7. To provide for smooth and rapid restoration of service.

Section 2. Personnel

Data Processing Personnel

Table C-1. Data Processing Personnel

Name	Position	Address	Telephone

User Personnel

Table C-2. User Personnel

Name	Position	Address	Telephone

Other Important Telephone Numbers

Table C-3. Other Important Personnel

Name	Position	Address	Telephone

Organization Chart

Attach the organization chart here.

Section 3. Application Profile

Use the Display Software Resource (DSPSFWRSC) command to fill in this table.

Table C-4. Applications Profile

Name	Critical Yes / No	Fixed Asset Yes / No	Manufacturer	Comments

Comment Legend:

1. Runs daily
2. Runs weekly on ______________.
3. Runs monthly on ______________.

Section 4. Inventory Profile

Use the Work with Hardware Products (WRKHDWPRD) command to fill in this table.

Table C-5. Inventory Profile

Manufacturer	Description	Model	Serial Number	Own or Leased	Cost

Notes:

1. This list should be audited every ________________ months.

2. This list should include:

- Processing units
- Disk units
- Models
- Workstation controllers
- Personal computers
- Spare workstations
- Telephones
- Air conditioner/heater
- System printer
- Tape and diskette units
- Controllers
- I/O processors
- General data communication
- Spare displays
- Racks
- humidifier/dehumidifier

Table C-6. Miscellaneous

Description	Quantity	Comments

Note:

This list should include:

Tapes	Diskettes
PC software (such as DOS)	Emulation packages
File cabinet contents/Documentation	Language software (such as COBOL and PASCAL)
Tape vault contents	Printer supplies (such as paper and forms)

Section 5. Disaster Recovery

For any disaster recovery plan, the following three elements should be addressed.

- Emergency Response Procedure - to document the appropriate emergency response to a fire, natural disaster, or any other activity in order to protect lives, limit damage.
- Backup Operations Procedures - to ensure that essential data processing operational tasks can be conducted after disruption to site.
- Recovery Actions Procedure - to facilitate the rapid restoration of data processing facility following a disaster.

1. Disaster Action Check List

   a. Initiation of Plan

      1) Notify senior management
      2) Contact and setup disaster recovery team
      3) Determine degree of the disaster
      4) Implement the proper application recovery plan dependent on extent of disaster (see "Section 6. Recovery Plan Mobile Site")
      5) Monitor the progress
      6) Contact backup site and establish schedules.
      7) Contact all other necessary personnel - both user and data processing.
      8) Contact vendors - both hardware and software.
      9) Notify users of disruption of service.

   b. Check List (Follow-up)

      1) List of teams and tasks of each.
      2) Obtain emergency cash and set-up transportation to/from backup site if necessary.
      3) Set-up living quarters if necessary
      4) Set-up eating establishments as required.
      5) List of all personnel and telephone numbers.
      6) Establish user participation plan.
      7) Set-up delivery/receipt of mail/UPS.
      8) Establish emergency office supplies.
      9) Provide for rental or purchase of equipment as needed.
      10) Determine application(s) to be run and sequence.
      11) Identify number of displays needed.
      12) Check out any off-line equipment needs per application.
      13) Check on forms requirement by application.
      14) Double check all data being taken to backup site before leaving and leave inventory profile at home location.
      15) Set-up primary vendors regarding assistance with problems incurred during emergency.
      16) Plan for transportation of any additional items needed at backup site.
      17) Take directions (map) to backup site.
      18) Check for additional magnetic tapes if required.
      19) Take system and operational documentation and procedural manual (copies).
      20) Review with all personnel involved that each knows their tasks.
      21) Notification of insurance companies.

Recovery Start-Up Procedures for Use After Actual Disaster

1. Notify ____________ Disaster Recovery Services of the need to utilize service and of recovery plan selection.

   Note: Guaranteed delivery time countdown begins at the time ____________ is notified of recovery plan selection.

   a. Disaster Notification Numbers

      ____

      or ____________

      The above numbers are in service from ____________ am until ____________ pm Monday through Friday.

   b. Disaster Notification Number

      ____

      The above number is in service for disaster notification after business hours, weekends, or holidays. Please use the above number for notification of actual disaster only.

2. Provide ____________ with an equipment delivery site address (when applicable), a contact, and an alternate contact for coordinating service and telephone numbers at which contacts can be reached 24 hours a day.
3. Contact power and telephone service suppliers and schedule any necessary service connections.
4. Notify ____________ immediately if any related plans should change.

Section 6. Recovery Plan Mobile Site

1. Notify _____________ of the nature of the disaster and of its desire to select the mobile site plan.
2. Confirm in writing the substance of the telephone notification to _____________ within 48 hours of the telephone notification. (See sample letter attached).
3. Confirm all needed backup media is available to load backup machine.
4. Cut purchase order to cover use of backup equipment.
5. Notify _____________ of impending trailer and its placement (on _____________ side of _____________). (See -- Fig 'MBLSITE' unknown --.)
6. Depending on communication needs, notify telephone company (_____________) of possible emergency line changes.
7. Begin set-up of power and communication at _____________.
   a. Power and communication would be prearranged to hook into when trailer arrives.
      1) At point where telephone lines come into building (_____________), a break to the current linkage to the administration controllers (_____________). These lines would be rerouted to lines going to the mobile data center. They would be linked to lads modems at the mobile unit.

         The lines currently going from _____________ to _____________ would then be linked to the mobile unit via lads modems.
   b. This could conceivably require _____________ to redirect lines at _____________ complex to a more secure area in case of disaster.
8. When trailer arrives, plug into power and Do necessary checks
9. Plug into communication lines and do necessary checks
10. Begin loading system from backups (see "Section 7. Restoring the Entire System").
11. Begin normal operations as soon as possible:
    a. Daily jobs
    b. Daily saves
    c. Weekly saves
12. Plan schedule to backup the system in order to restore on home base computer when site is available. (Utilize regular system backup procedures).
13. Secure mobile site and distribute keys as required.
14. Keep a maintenance log on mobile equipment.

Mobile Site Setup

Attach the mobile site setup plan here.

Telecommunications Disaster Plan

Attach the telecommunications disaster plan here.

Communication Disaster Plan

Attach the communications disaster plan, including the wiring diagrams.

AMP Service

Attach AMP service diagram here.

Recovery Plan Hot Site

The disaster recovery service will provides an alternate site (hot site). The site has a backup system for temporary use while the home site is being reestablished.

1. Notify ____________ of nature of the disaster and of its desire for a hot site and request air shipment of modems to ____________ for communications. (See ____________ for communications for the HOT SITE.)
2. Confirm in writing the substance of the telephone notification to ____________ within 48 hours of the telephone notification.
3. Begin making necessary travel arrangements to site for the operations team.
4. Confirm that all needed tapes are available and packed for shipment to restore on the backup system.
5. Cut purchase order to cover use of the backup system.
6. Review checklist for all necessary materials before departure to the hot site.
7. Make sure the disaster recovery team at the disaster site has necessary information to begin restoring site. (See "Section 11 Disaster Site Rebuild").
8. Provide for travel expenses (cash advance).
9. After arrival at hot site, contact home base to establish communication procedures.
10. Review materials brought to the hot site for completeness.
11. Begin loading the system from the save tapes.
12. Begin normal operations as possible:
    a. daily jobs
    b. daily saves
    c. weekly saves
13. Plan schedule to backup the hot site system in order to restore on the home base computer.

Hot Site System Configuration

Attach hot site configuration here.

Section 7. Restoring the Entire System

The following procedures outline the steps that need to be taken in order to install the system back to where it was before the disaster.

Note: This example assumes you are using a complete save strategy (SAVSYS, SAVLIB *NONSYS, SAVDLO), not save storage (SAVSTG).

First, the following tapes and equipment need to be retrieved from the tape vault or the off site storage location:

1. The model-unique licensed internal code tape
2. All tapes from the last weekly save operation.
3. All tapes from the last daily save operation
4. The last save security data tapes.
5. All tapes containing journals and journal receivers saved since the last daily save operation.
6. Tape list for the weekly save
7. PTF list stored with the last weekly save tapes.
8. Tape list from daily saves.
9. History log from the last weekly save operation.
10. History log from the daily save operations.
11. The *Licensed Programs and New Release Installation Guide*, SC41-9878.
12. The *System Operator's Guide*, SC41-8082.
13. The *Backup and Recovery Guide*, SC41-8079.
14. Device description information.
15. Telephone directory.
16. Modem manual
17. Tool kit

Once these tapes and manuals are available, the restoring can begin.

Total System Restore

This procedure is used to restore the entire system after a disaster has occurred.

Task Overview

The following tasks must be completed for a total system restore. It is assumed that you have your current SAVSYS tapes.

1. Install the licensed internal code
2. Install the model-unique licensed internal code
3. Restore the operating system
4. Restore:
   a. User profiles (RSTUSRPRF)
   b. Device configurations (RSTCFG)
   c. User libraries (RSTLIB)
   d. Document library objects (RSTDLO)
   e. Restore changed objects (RSTOBJ) from the daily SAVCHGOBJ tape
   f. Document library objects (RSTDLO) from the daily SAVDLO tape
   g. Apply journal changes (APYJRNCHG)
   h. Restore authority (RSTAUT)
5. Restore program temporary fixes (PTFs)

Step 1. Installing the Licensed Internal Code

Before You Begin

Installing the licensed internal code during this procedure is done using function code **24** (Install), which is run completely from the control panel. Because it is run completely from the control panel, this function is referred to as a *stand-alone* function.

When you perform this option, the system will continuously display system reference codes (SRCs) in the control panel display lights. The yellow System Attention light will be on whenever user input is needed. SRCs that start with A6 indicate that the system is waiting for you to do something, such as answer a question or prepare a tape device. When xx is shown in the SRC (such as A6xx 6001), a variety of characters may be shown in the place where the xx appears. Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" in the *Basic Backup and Recovery Guide*, SC41-0036, has a description of the SRCs. If an SRC is shown that is not discussed in the *Basic Backup and Recovery Guide*, SC41-0036, contact your service representative.

The documentation for this function makes reference to a **licensed internal code tape**. This is the **first** volume of the customer's most recent set of complete save tapes that were used to save the system (SAVSYS, SAVLIB *NONSYS, and SAVDLO). The **model-unique licensed internal code tape** is the tape found in the service kit delivered with your system, or stored with you backup tapes.

1. Ensure the key is in the keylock switch on the control panel.
2. Turn the key to the Manual position.
3. Press the Function Select switch to display **02** in the Function display on the control panel.

4. Press the Enter button on the control panel.
5. Select IPL type D (this specifies that the IPL source comes from tape) by pressing the Function Select switch on the control panel until **D** is shown on the Data display.
6. Press the Enter button on the control panel.
7. For the 9406 system unit, ensure that the power switches for the IPL tape unit and all disk units are in the On position.
8. Find the licensed internal code tape, which is the first volume of the most recent set of save tapes.
9. Place the tape in the tape unit used for the IPL. See the setup manual for the device for more information on loading the tape.

   Note: If your tape unit cannot be loaded when the power is off, continue with the next step. You will be prompted later by an SRC code for the tape.
10. Turn on the power to the system by pushing the Power switch up. The switch returns to center after you push it. The 9402 system unit has a green button labeled Power On.
11. Wait as explained below for the tape unit to power on. See the following explanations:

    Notes:

    a. SRC A100 1938 (Tape not found) may be displayed along with the System Attention light until the tape unit is turned on. Continue with the next step.

    b. SRC A100 1933 or 1934 (Tape not ready) may also be displayed along with the System Attention light. When the tape is ready, continue with the next step. If this SRC is displayed for more than 1 or 2 minutes, call your service representative.

    c. SRC 2507 XXXX appears if manual select IPL intervention is required. If intervention is required, press the Function Select switch until 03 appears. Press the Enter button on the control panel.
12. Ensure that the tape is online or ready. No action is required for tape units that perform this step automatically (such as the tape cartridge unit).
13. Ensure that the console display is turned on.
14. Wait for the yellow System Attention light on the control panel to light up.

    There is a delay while the system loads information from the tape. SRCs showing status are continuously updated on the control panel while processing occurs. This can take from 5 to 20 minutes; the time varies depending on the speed of the tape unit and the processor speed for the specific system model.

    When SRC A6xx 6001 is displayed, the system is prepared to start installing or restoring the licensed internal code on the disk unit containing unit 1. Continue with the next step.
15. Select the correct function code by pressing the Function Select switch on the control panel until the correct function code is displayed. (See the "Before You Begin" section in this topic to determine the correct function code.)

Warning: Option 24 (Install) is used only to recover from the loss of unit 1 in the system ASP, or to recovery from a disaster. Option 24 (Install) deletes all information on the disk unit containing unit 1, including customer data. All system configuration information is also deleted. All disk units except unit 1 become nonconfigured units during the IPL.

If you selected function code 24, the System Attention light light may appear in one or two minutes and SRC A6xx 6002 is shown. If you are sure you want to install the licensed internal code, select function code 24 again.

Notes:

a. When SRC A6xx 6002 is displayed, option 23 can also be selected. If option 23 is selected, a restore of the licensed internal code is performed (not an install). Data on the disk units will not be lost.

b. If another SRC appears after A6xx 6001 that is not in the A6xx xxxx format, then the system needs additional attention. Call your service representative.

   If the following SRCs appear after SRC A6xx 6001 is displayed, see Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" for an explanation of these SRCs and the steps to follow.

   - A6xx 6002 Disk unit may contain a valid system
   - A6xx 6003 Disk unit not currently a load source
   - A6xx 6004 Disk unit not currently a load source
   - A6xx 6005 Disk unit not found

16. Press the Enter button on the control panel.

    Note: The system will start displaying status SRCs again, which will be continuously updated to show the status of installing the licensed internal code. An example of a status SRC is D6xx 6201 (stand-alone install operations is running).

17. If the yellow System Attention light is on again, and SRC A6xx 6048 (New tape volume needs to be loaded) appears, the system needs the next tape. The xx tells which volume needs to be loaded. Load the correct tape and make the device ready. The install operation will automatically continue.

    If SRC A6xx 6051 appears, the stand-alone function is requesting the model-unique licensed internal code tape found in the service kit. Unload the current tape from the tape device and load the model-unique licensed internal code tape.

    - A6xx 6051 Model-unique licensed internal code tape needs to be loaded
    - A6xx 6052 Tape loaded was not the model-unique licensed internal code,

    If another SRC A6xx xxxx is displayed, look up the displayed SRC in Appendix A, "Licensed Internal Code SRCs That Require User Input (A6xx xxxx)" on page A-1 and follow the instructions. For all other SRCs call your service representative.

18. After the install or restore operation of licensed internal code or the model-unique licensed internal code is complete, the system will automatically perform an IPL.

19. Remove the model-unique licensed internal code tape from the tape unit (if it is loaded) when the IPL or Install the System menu is shown.

20. If you have other restore steps to perform and the model-unique licensed internal code tape was loaded, load the previous save tape.

Step 2. Restoring the Operating System

The following procedure is used to restore only the OS/400 licensed program from tape. The procedure for restoring the operating system assumes that the licensed internal code is already installed or restored on the system.

You use two displays to select the install options. The IPL or Install the System display allows you to install the operating system or work with the service tools. The Install the Operating System display allows you to set the options to be used for installing or restoring of the system, and for the system date and time.

At the IPL or Install the System menu:

```
                      IPL or Install the System

Select one of the following:

     1. Perform an IPL
     2. Install the operating system
     3. Use Dedicated Service Tools (DST)
     4. Perform automatic installation of the operating system
```

1. Type a **2** (Install the operating system).

   Note: **Do not** use option 4 (Perform automatic installation of the operating system) to restore the operating system.

2. Press the Enter key.

   The Confirm Install of the Operating System display is shown.

```
                  Confirm Install of Operating System

Press Enter to confirm your choice to install the operating system.
Press F12 to return and cancel your choice to install the
  operating system.

 F12=Cancel
```

3. Press the Enter key.

4. The Select a Language Group display is shown. This display shows the primary language currently on the system.

   The value specified on the display must be the same as the national language that is on the distribution media, or on your last SAVSYS tape. You may change the primary language feature of your system by specifying a different primary language feature on this display.

```
                         Select a Language Group

Note:  The language feature shown is the language feature
installed on the system.

Type choice, press Enter.

  Language feature . . . . . . . . . . . . . . .   2924










F3=Exit          F12=Cancel
```

5. Press the Enter key.

   After the language feature is entered, the Confirm Language Feature Selection display is shown. If you need to change your system's primary language, see the *Licensed Programs and New Release Installation Guide* for more information.

```
                    Confirm Language Feature Selection

Language feature  . . . . . . . . . . . . . . . . . :   2924

Press Enter to confirm your choice for language feature.
  Installing the system will continue.
Press F12 to return to change your choice for
  language feature.










F3=Exit   F12=Cancel
```

6. Press the Enter key to confirm the information.
7. The following display is shown only if disk units have been attached to the system and are in nonconfigured status.

```
                    Add All Disk Units to the System

Select one of the following:

        1. Add all disk units to the system auxiliary storage pool
        2. Keep the current disk configuration
        3. Perform disk configuration using DST









Selection
    __
```

Option 1 (Add all disk units to the system auxiliary storage pool)

Select this option if you want to add all of the nonconfigured units to the system auxiliary storage pool. Before adding the units to the system, all data stored on the non-configured units is deleted.

Note: Adding units can change the checksum set configuration of the system ASP. You can use option 3 (Perform disk configuration using DST) to calculate the effect of adding units to the system ASP.

Option 2 (Keep the current disk configuration)

Select this option if you plan to add the nonconfigured units to user ASPs or use them as spare units. This option continues the IPL without adding units to the system configuration. The disk units that are in nonconfigured status will remain so.

Option 3 (Perform disk configuration using DST)

This option starts the Dedicated Service Tools (DST). On the DST main menu, select option 4 (Work with Disk Units).

8. Press the Enter key.

   As the disk units are configured, the following display is shown:

```
          Formatting additional disk units in progress.
                  Seconds running . . . . :  ____
```

Adding disk units takes several minutes. The time it takes depends on the size of each unit and the ability of the system to do multiple adds at the same time.

9. Status messages are displayed.

   The following is an example of a status display shown before the Install the Operating System display is shown. The status messages shown do not require any action by the user.

```
                              Running IPL Step

 Current IPL step . . . . : Storage Management Recovery
```

After the IPL steps complete, the Install the Operating System menu appears.

```
                         Install the Operating System

Type choices, press Enter.

    Default
      option . . . .  _             1=Take defaults, show no
                                      other installing displays
                                    2=Change installing options

    Date:
      Year . . . . .  _             00-99
      Month  . . . .  _             01-12
      Day  . . . . .  _             01-31

    Time  . . . . . . _    00-23    HH is hours
                      _    00-59    MM is minutes
                      _    00-59    SS is seconds
```

10. When the Install the Operating System display is shown, use the following information to respond to the prompts.

    Default Option

Value	**Description**
1	Taking the defaults will keep objects, such as the system values, system reply lists, and the edit descriptions intact if you are installing the system as part of a restore operation. If you select 1 for *Default option*, the operating system will be installed again and there are no more install steps to perform. Fill in the date and time. When the operating system is installed, the Sign On display is shown.
2	If you have not cleared the system (operating system installed), and you want to restore system values, edit descriptions, or message reply lists, change the *Default option* to 2. Select this option to do an abbreviated install of the operating system. If you select 2 for *Default option*, the Installing Options display appears.

    Date

    The system inserts the date based on the internal clock. If the date is incorrect, you can type over the date to change it.

    Time

    The system inserts the time based on the internal clock. If the time is incorrect, you can type over the time to change it.

11. Press the Enter key.

    If you selected option 1 (Take defaults, show no other installing displays) on the Install the Operating System display, messages are shown to indicate how many programs and language objects are restored. These messages are for your information only. Continue loading tapes in sequence when

messages are shown that ask you to load the next tape. After processing all the tapes, a message is shown at the bottom of the display indicating that the install of the operating system is complete and an IPL is in progress. When the IPL is complete, the Sign On display is shown. Continue with the step to sign on as QSECOFR.

If you selected option 2 (Change installing options), the following display is shown.

```
                         Specify Install Options

Type choices, press Enter.

    Restore option . . . . _          1=Restore programs and language
                                        objects from current tape
                                      2=Do not restore programs or
                                        language objects
                                      3=Restore only language objects
                                        from current tape
                                      4=Restore only language objects
                                        from a different tape




    Job and output
      queue options. . . .  1         1=Clear,  2=Keep
```

12. When the Installing Options display appears, use the following information to respond to the prompts.

Restore Option

Value	**Description**
1	Type a 1 if you want to restore the system objects from tape.
2	Type a 2 to leave the current programs and language objects on the system. Select this option to do an abbreviated install of the operating system.

Notes:

a. When option 2 (Do not restore programs or language objects) is selected during an install, nothing is restored from tape.

b. All libraries on the system are checked for damage. A library is deleted and then created again if it is damaged.

c. All system libraries (including QSYS) are created if they do not exist, or they are deleted and created again if they are damaged.

d. Information associated with the user profiles is created if it does not exist, or it is created again if it is damaged.

e. The system entry-point table is created again.

f. If damage to a system user profile is found, a message is shown to indicate that an abbreviated install is not allowed. If the user

decides to continue, the install procedure continues restoring programs and language objects.

3 or 4 Type a 3 or 4 if you want to change the system's primary language.

Value	Description
3	Entering a 3 loads only those objects that make up the national language. The search for the language files begins on the same tape.
4	Entering a 4 for this option loads only those objects that make up the system language. In this case, the language files are on a different tape. You will receive a message prompting you to insert the other tape. You can use option 4 to update the system language from a new (and separate) language tape.

Clear Job and Output Queues

When you are performing system updates, you will be directed to clear the output and job queues when it is necessary. If the output queues are damaged, you should clear the job and output queues.

1=Clear You want to clear all job queues and output queues on the system.

2=Keep You do not want to clear all job queues and output queues; any entries on the queues remain after installing. Select this option to perform an abbreviated install of the operating system.

Notes on Clearing Job and Output Queues

The first time you install the OS/400 program on your system, the OS/400 program creates a number of objects (such as the system operator message queue).

If you install the OS/400 program again later with *Clear job and output queues*=2 (Keep), the OS/400 program re-creates these objects automatically if it finds that they were damaged.

If you install the OS/400 program with *Clear job and output queues*=1 (Clear), it re-creates all objects whether they were damaged or not.

Selecting *Clear job and output queues*=1 (Clear) will also start the counter used to assign unique job numbers.

If you entered 1 for *Restore option*, the Restore Options display appears.

```
                          Specify Restore Options

Type choices, press Enter.

   Restore from tape:

     System values . . . . . 2        1=Restore,  2=Do not restore
     Edit descriptions . . . 2        1=Restore,  2=Do not restore
     Message reply list. . . 2        1=Restore,  2=Do not restore
```

13. Using the following information, respond to the prompts on the Restore Options display.

System Values

Entering a 1 (Restore) for this option restores the system values from tape. If you received a message during a previous IPL stating that the system value object was created again, select 1.

If you select a 2 (Do not restore), the system is installed with no change to the system values that are currently on the system.

A system value object must always be present on an operational system. Therefore, if the system value object does not exist on the system, it will be restored from tape regardless of what was selected for this option.

Also consider the language-sensitive system values. These are the system values that have different defaults depending on the system's primary language. These system values are:

QCHRID	Default system code page and character set
QCURSYM	Currency symbol
QDATFMT	Date editing format
QDATSEP	Date separator character
QDECFMT	Decimal data editing format
QKBDTYPE	Default work station keyboard type
QLEAPADJ	Leap year adjustment

These system values will be set to default values for the system's primary language if you are changing the primary language or doing an initial install procedure. You can override these values by entering 1 (Restore) on this display. This will cause the system to restore the system values from the tape and not reset the language-sensitive system values to the defaults.

For more information about system values, see the *Work Management Guide*.

Edit Descriptions

Value	**Description**
1 = Restore	This option loads the edit descriptions from tape. If the edit descriptions are damaged on your system, or if you want to put them back the way they were when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2 = Do not restore	The operating system is installed with no change to the edit descriptions currently on the system.

For more information about edit descriptions, see the *Programming: Reference Summary*, SX41-0028.

Message Reply List

Value	Description
1=Restore	This option loads the reply list from tape. If the message reply list is damaged on your system, or if you want to put it back the way it was when you saved your system, select 1. If you cleared the disks before installing the operating system, you should also select 1.
2=Do not restore	The operating system is installed with no change to the reply list currently on the system.

For more information about message reply lists, see the *Programming: Reference Summary*, SX41-0028.

The defaults for these options will be 2 if the operating system is loaded on the system. The defaults will be 1 if the operating system is not already loaded.

14. Press the Enter key.

    Messages are shown to indicate how many program or language objects are restored. They are for your information only and require no response.

15. Continue loading tapes in sequence when messages are shown that ask you to load the next tape.

    `Load the next volume on the tape device (1 2).`

16. Type 1 and then press the Enter key.

    The system searches through the tapes and loads the necessary programs and language information. After processing all the system save tapes, a message is displayed at the bottom of a blank display:

    `Operating system has been installed. IPL in progress.`

    Then the IPL Sign On display appears, and the system is ready to complete the IPL.

17. Type `QSECOFR` in the *User* prompt and the password required for that user ID in the *Password* prompt (if password security is active).

18. Press the Enter key. Informational messages are displayed.

19. The Work with Licensed Program PTFs display may appear. Press F3 (Exit) to continue.

20. When the IPL Options display is shown, respond to the prompts using the following information.

```
                              IPL Options

Type choices, press the Enter key.

  System date . . . . . . . . . . . . . .   07 / 26 / 88   MM / DD / YY
  System time . . . . . . . . . . . . . .   12 : 00 : 00   HH : MM : SS
  Clear job queues . . . . . . . . . . .    N              Y=Yes,  N=No
  Clear output queues . . . . . . . . . .   N              Y=Yes,  N=No
  Clear incomplete job logs . . . . . . .   N              Y=Yes,  N=No
  Start print writers . . . . . . . . . .   Y              Y=Yes,  N=No
  Start this device only . . . . . . . . .  N              Y=Yes,  N=No


  Set major system options . . . . . . . .  N              Y=Yes,  N=No
  Define or change system at IPL . . . . .  N              Y=Yes,  N=No






Last power-down operation was normal
                                     (C) COPYRIGHT IBM CORP 1980, 1991
```

Figure C-1. IPL Options Display

a. Enter the value for the system date.

   The date is displayed. The system date format shown can be YY/MM/DD, DD/MM/YY, or MM/DD/YY where MM means month, DD means day, and YY means year. For English, the default is MM/DD/YY; the default value differs according to the primary language

   If the date is not correct, you can type over the date to change it. The system date must have a year value in the range of 87 to 99, or 00 to 22.

b. Enter the value for system time.

   The current time is displayed. The time format is HH : MM : SS; HH means hour, MM means minutes, and SS means seconds. If you want to change the time, type it in accordance with the 24-hour clock. For example, for an IPL at 4:30 p.m., type 163000 for the time.

c. Enter the value for start this device only.

   Note: Change this option only if you are going to continue restoring user profiles, device configuration objects, user libraries, and authorities.

   Type a Y to start the console device only, or type an N to start all devices.

d. Enter the value for set major system options.

   Type an N for each option to leave them unchanged, or type Y to set the specific options.

   Note: The default is different, depending on the type of restore operation. If you restored the licensed internal code using option 23 (Restore), the default value is set to N. If you installed the licensed internal code using option 24 (Install), the default value is set to Y.

21. Press the Enter key.

22. The following display is shown during the IPL process (attended mode) when there are access paths marked for rebuild:

```
                 Edit Rebuild of Access Paths               RCHAS331
                                                  05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                   ------------Access Paths----------  Unique  Rebuild
Seq   Status       File        Library     Member      Keyed   Time
25__  IPL          FILE234512  LIBRARY111  MBR1234567  No      00:00:56
25__  IPL          FILE234513  LIBRARY111  MBR1234567  No      00:00:56
75__  IPL          FILE234514  LIBRARY111  MBR1234567  Yes     00:00:56
75__  IPL          FILE234515  LIBRARY111  MBR1234567  Yes     00:00:56
88__  IPL          FILE234516  LIBRARY111  MBR1234567  No      00:00:56
99__  AFTIPL       FILE234517  LIBRARY111  MBR1234567  Yes     00:00:56
*OPN  OPEN         FILE126789  L123456789  MBR4567890  Yes     12:34:56
*OPN  OPEN         FILE346789  L123456789  MBR4567890  No      12:34:56
*HLD  HELD         F123336789  L123456789  MBR4567890  No      10:30:06
*HLD  HELD         F123456789  L123456789  MBR4567890  Yes     99:56:01
                                                          More...

F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

This display does not support the F3 and F12 keys.

- A message is sent to notify the user that there is access path recovery to be performed by a journal.
- The *IPL threshold* is a value from 1 through 99 that can be set by the user (default is 50), which indicates that access paths with a sequence less than or equal to the number specified will be rebuilt at IPL time. If the IPL threshold changes, all access paths with a status of IPL and AFTIPL will be changed to reflect the new status of the IPL threshold.
- Sequence
  - *IPL threshold*-1 represents the sequence of the access paths that are to be rebuilt prior to completion of the IPL. A rebuild sequence of 25 is initially set by the system to set the sequence of access paths for the files that have MAINT(*IMMED) and RECOV(*IPL) specified. The access paths with the same sequence are built first according to rebuild time (the access paths that take the longest to rebuild are rebuilt first if the priorities are the same). The access paths are displayed in the same order.
  - *IPL threshold*-99 represents the sequence in which the access paths are rebuilt after the IPL. A rebuild sequence of 75 is initially set by the system to set the sequence of the access paths for the files that have MAINT(*IMMED) and RECOV(*AFTIPL) specified.
  - *OPN indicates the access path is to be rebuilt when the file is opened. The *OPN must be changed to 1 through 99 before the system job will initiate the rebuild. The system initially sets the sequence to *OPN for the files that have MAINT(*IMMED) and RECOV(*NO) specified.
  - *HLD indicates the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99. *HLD will also cancel the rebuilding of any access path.

- Status
  - RUN indicates that the access path is being rebuilt.
  - IPL indicates that the access path is to be rebuilt before the system completes the IPL process.
  - AFTIPL indicates that the access path is to be rebuilt after the the system completes the IPL process.
  - HELD indicates that the access path is not to be rebuilt until the user changes the sequence from *HLD to a *OPN, or 1 through 99.
  - OPEN indicates that the access path is to rebuilt when the file is opened.
- Rebuild Time
  - The time the access path will take to be rebuilt when the system is running without any other jobs on the system. For example, at IPL time. This is an estimate of rebuild time based on the file size and key length. No time for journaled access paths is displayed.
- Keys
  - Pressing the Enter key after any change has been made to the fields on the display causes the change to be made, if possible. For example, if the user attempts to change the sequence from 9 to 50, but the sequence cannot be changed because the access path has already been rebuilt, the user is sent an error message for each improper input.
  - Pressing the Enter key without making any changes to the display causes the status display to be displayed (only if there are access paths left to be rebuilt). If no access paths need to be rebuilt, the status display is not shown and the IPL continues.

The following display is shown during the IPL process when the user finishes with the Edit Rebuild of Access Paths display.

```
                       Display Access Path Status

IPL Threshold . . . . . . : 88

          ----------Access Paths----------   Rebuild     Current
Status    File        Library     Member       Build Time  Run Time
RUN       F123456789  L123456789  MBR4567890               00:00:01
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
JRN       F123456789  L123456789  MBR4567890
SYS       F123456789  L123456789  MBR4567890   12:34:56
SYS       F123456789  L123456789  MBR4567890   12:34:56
IPL       F123456789  L123456789  MBR4567890   12:34:56
                                                            More...




F3=Exit and continue IPL    F12=Cancel
```

Every 5 seconds the display is updated with the current run time.

After all the access paths have been rebuilt (access paths with a sequence less than or equal to the *IPL threshold*), the IPL process continues and this display is removed.

F12 (Cancel) calls the Edit Rebuild of Access Paths display. If the user returns to the Edit Rebuild of Access Paths display using F12 (Cancel), the user must exit the Edit Rebuild of Access Paths again. Even if all the access paths are rebuilt, the user remains at the Edit Rebuild of Access Paths display until the user exits the display.

If F12 (Cancel) is pressed and there are only SYS/JRN access paths to be recovered, the edit display is shown without any access paths to be edited.

Status

- RUN—indicates that the access path is being rebuilt.
- SYS—indicates the access path is a system access path and is waiting to be rebuilt.
- JRN—indicates that the access path is being recovered from its associated journal.
- IPL—indicates that the access path will be rebuilt before the system completes the IPL and is waiting to be rebuilt.

23. Press the Enter key to continue.
24. Ensure the keylock switch is in the Normal position.
25. This completes the restore operation for the operating system if you have no other restore steps to perform.
26. If you restored the operating system as part of restoring the entire system, you must restore the user profiles, configuration objects, user libraries, document library objects, and authorities.

Step 3. Restoring User Profiles, Device Configurations, User Libraries, and Authority

Before You Begin. . .

- ☐ Clean the read and write head of the tape unit.
- ☐ Find the tape volume that contains the user profiles.

You may want to do the following:

- If you do not know where the user profiles are stored on tape, determine where they are by using the DSPTAP command with DATA(*SAVRST) specified. (This is not necessary if you are restoring user profiles from the SAVSECDTA media.)
  - Examine each tape until you find the file (label QFILEUPR) containing object type *USRPRF. If you are restoring the system from the tapes you used for the SAVSYS command (not option 21 of the Save menu), the user profiles are in the next to the last file on the last tape.
- Find the file (label QFILEIOC) containing the device configuration objects (*DEVD, *CTLD, *LIND, *COSD, and *MODD) on the tape using the DSPTAP command with DATA(*SAVRST) specified.
  - Examine each file until you find the file containing the object types just listed. If you are restoring the system from the tapes you used for the

SAVSYS command (not option 21 of the Save menu), the device configuration objects are in the last file on the last tape (immediately following the user profiles).

- Use the VRYCFG command to vary off the device configuration objects to be restored. The device configuration objects cannot be active when being restored. When doing an IPL, specify **Yes** for the *Start this device only* prompt on the IPL options display. This does an IPL with only the console and the tape unit varied on.

Other Considerations: Use the restore commands (not option 21 on the Restore menu) if:

- You prefer to enter the commands manually.
- You saved changed objects or have journal changes to apply.
- You performed individual save operations instead of SAVLIB(*NONSYS), you must use a RSTLIB command for each saved library. If you saved individual objects using the SAVOBJ or SAVCHGOBJ command, you must use a RSTOBJ command for each group of saved objects.
- You saved the security information with the Save Security Data (SAVSECDTA) command, you must restore the information using the restore commands.
- You saved logical file access paths using either the SAVOBJ or SAVCHGOBJ command, you must restore the logical files the same way you restored the physical files using the RSTOBJ command.
- Your documents and folders were saved in multiple tape files using SAVDLO DLO(*ALL) FLR(*ANY), use DSPTAP DATA(SAVRST) to display the beginning and ending sequence numbers. You will need this information later during the restore operation.

Use **one** of the following two methods to restore the user profiles, device configuration objects, user libraries, and authority:

1. Use option 21 (The system) on the Restore menu if you are restoring the user profiles from a SAVSYS tape and the considerations listed previously in "Other Considerations" do not apply.
2. Use the Restore commands (not option 21 on the Restore menu) if any of the consideration listed previously in "Other Considerations" apply.

Method 1. Using Option 21 (The system) on the Restore Menu

To restore user profiles, device configuration objects, user libraries, and authority, use the following steps:

1. Sign on the system as the security officer; type `QSECOFR` in the user prompt and the password (if password security is active) associated with that user ID on the Sign On display.
2. Press the Enter key.
3. Type the following command on the command line and press the Enter key.

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```
4. Ensure that the correct volume of your last set of save tapes is loaded and make the tape device ready. The tape should begin with a file containing the

user profiles. Run the DSPTAP using *SAVRST to find the file labeled QFILEUPR.

5. Go to the Restore menu:

   GO RESTORE

   The Restore menu is shown.

```
RESTORE                           Restore
                                                        System:   RCHAS791
Select one of the following:

     1. Files
     2. Libraries
     3. Documents and folders
     4. Programs
     5. Other objects
     6. Licensed programs

    20. All libraries other than system library
    21. The system
    22. All IBM libraries other than system library
    23. All user libraries

    50. Restore from System/36 format
                                                                   More...
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=User support
F16=AS/400 Main menu
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

6. Press the Enter Key.

7. Select option 21 (The system) on the Restore menu and press the Enter key. The following display is shown.

```
                        Specify Command Defaults

Type choices, press Enter.

  Tape devices . . . . . . . . .  : TAP01               Names
                                    ______
                                    ______
                                    ______

  Prompt for commands  . . . . .  :   Y                 Y=Yes, N=No











F3=Exit   F12=Cancel
```

Tape devices

You can specify up to four tape device names. If you specify more than one device, the system automatically switches to the next tape device after the first tape is read.

Prompt for commands

You can specify whether or not you want to be prompted for the commands. If you specify Y=Yes, the prompt display is shown and you can change the defaults on the commands. If you specify N=No, the system runs the commands without prompting and uses the default values.

Note: If you have multiple tape files from the SAVDLO DLO(*ALL), type a *Y* (the default) for the *Prompt for commands* field on the Specify Command Defaults display. When the RSTDLO command prompt is displayed, enter the beginning and ending sequence numbers to restore all SAVDLO tape files.

Option 21 will guide you through the following if you selected Y for the *Prompt for commands* prompt on the Specify Command Defaults display.

a. ENDSBS *ALL *IMMED
b. RSTUSRPRF *ALL
c. RSTCFG *ALL
d. RSTLIB SAVLIB(*NONSYS)
e. RSTDLO DLO(*ALL) SAVFLR(*ANY)
f. RSTAUT
g. STRSBS

8. Continue loading the save tapes in sequence when the system sends a message to load the next volume.

If a media error occurs....

If an unrecoverable media error occurs during the RSTLIB procedure, you can restart the procedure using the STRLIB parameter on the RSTLIB command. Do one of the following:

1. If the STRLIB parameter was used on the SAVLIB command), determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). Load the next tape and enter the following:

   ```
   RSTLIB SAVLIB(*NONSYS) STRLIB(starting-library-name)
   ```

2. If the STRLIB parameter was not used on the SAVLIB command, look at the job log to determine the library where the previous RSTLIB failed.

   a. Determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). The correct starting library is normally the second library listed after the last library restored successfully.

   b. Load the first tape of the SAVLIB media and type the following command to start the restore operation again.

      ```
      RSTLIB SAVLIB(*NONSYS) STRLIB(starting-library-name)
      ```

9. If you are restoring the system using the SAVSYS tapes, this completes the restore operation. There are no other restore steps to perform.

10. Power down the system (PWRDWNSYS OPTION(*IMMED) RESTART(*YES) to perform a normal IPL and return the system to normal operations.

Method 2. Using the Restore Commands

To use the commands to restore the system, do the following:

1. Sign on the system.
2. Change the QSYSOPR message queue:

   ```
   CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(60)
   ```

3. End all subsystems:

   ```
   ENDSBS SBS(*ALL) OPTION(*IMMED)
   ```

   Messages are sent indicating when the subsystems have ended and the system is in a restricted state.

4. Restore user profiles from the correct save tape file (label QFILEUPR).

   Note: Use the tapes from the most recent complete save operation (SAVSYS), or if the security data was saved since the last complete save operation, use the SAVSECDTA tapes.

   If the SAVSYS tape is used:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*LEAVE)
   ```

   If the save security data tape (SAVSECDTA) is used:

   ```
   RSTUSRPRF DEV(TAP01) USRPRF(*ALL) MAIL(*YES) ENDOPT(*UNLOAD)
   ```

   The time that this takes can vary significantly.

5. Restore the device configuration objects from the SAVSYS tape:

   ```
   RSTCFG OBJ(*ALL) DEV(TAP01) OBJTYPE(*ALL) ENDOPT(*LEAVE)
   ```

   The time that this takes can vary significantly.

6. Restore the IBM and user libraries in one of the following ways:

   If you used SAVLIB LIB(*NONSYS) to save the IBM-supplied and user libraries, load the correct volume and then type the following:

   ```
   RSTLIB SAVLIB(*NONSYS) DEV(TAP01) ENDOPT(*LEAVE)
   ```

   Or, if you used SAVLIB LIB(*IBM) and SAVLIB LIB(*ALLUSR) to save the IBM and user libraries, load the correct tape and then type the following two commands. The first command must complete before entering the second command.

   ```
   RSTLIB SAVLIB(*IBM) DEV(TAP01) ENDOPT(*LEAVE)
   RSTLIB SAVLIB(*ALLUSR) DEV(TAP01) ENDOPT(*LEAVE)
   ```

   Note: If you saved individual libraries and objects with the SAVLIB, SAVOBJ, and SAVCHGOBJ commands, then you will have to restore the individual libraries and objects with the RSTLIB command (not RSTLIB SAVLIB(*NONSYS)) and the RSTOBJ command.

If a media error occurs....

If an unrecoverable media error occurs during the RSTLIB procedure, you can restart the procedure using the STRLIB parameter on the RSTLIB command. Do one of the following:

1. If the STRLIB parameter was used on the SAVLIB command), determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). Load the next tape and enter the following:

   ```
   RSTLIB SAVLIB(*NONSYS, *ALLUSR, or *IBM) STRLIB(starting-library-name)
   ```

2. If the STRLIB parameter was not used on the SAVLIB command, look at the job log to determine the library where the previous RSTLIB failed.

   a. Determine the next library to be restored beyond the failure by displaying the first tape using the command DSPTAP DEV(xxxx) DATA(*SAVRST). The correct starting library is normally the second library listed after the last library restored successfully.

   b. Load the first tape of the SAVLIB media and type the following command to start the restore operation again.

   ```
   RSTLIB SAVLIB(*NONSYS, *ALLUSR, or *IBM) STRLIB(starting-library-name)
   ```

7. If you have documents and folders to restore, load the correct tape and type the following:

   Note: If the saved data is contained in one or more files with folders and documents, you must specify beginning and ending sequence numbers on the RSTDLO command. Otherwise, only the first file with folders and documents will be restored.

   ```
   RSTDLO DLO(*ALL) SAVFLR(*ANY) DEV(TAP01) ENDOPT(*UNLOAD)
          SEQNBR(beginning-sequence ending-sequence)
   ```

   Note: If you are not using journaling, or do not have changed objects or daily mail (SAVSECDTA tape) to restore, continue with the next step. Otherwise, continue with "Restoring Changed Objects and Applying Journal Changes."

8. To restore the authority, type the following:

   ```
   RSTAUT
   ```

   This completes the restore operation.

9. Power down the system (PWRDWNSYS OPTION(*IMMED) RESTART(*YES)) to perform a normal IPL and return the system to normal operations.

Restoring Changed Objects and Applying Journaled Changes

If you are using journaling and need to later apply journaled changes, continue with the following steps. Otherwise ignore these steps and continue with "Restoring Changed Objects."

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily save operations using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform the steps in "Working with Journals" for the system supplied journal QUSRSYS/QAOSDIAJRN. If you specified OBJJRN(*YES) on the SAVCHGOBJ command, you do not need to apply journal changes.

Working with Journals

1. Type the following and press the Enter key:

   ```
   WRKJRN
   ```

2. The Specify Journal Name display is shown. Specify *ALL for the *Library name* prompt and press the Enter key.

3. The Work with Journals display is shown. Type a **5** in the *Opt* field for each journal that you want to apply journaled changes to display the name of the currently attached journal receiver. Write down the name of the journal and journal receiver.

   Note: If you are using OfficeVision/400 or PC Support/400, you must apply journaled changes to the system supplied journal QUSRSYS/QAOSDIAJRN.

4. You cannot restore journal receivers from the SAVCHGOBJ media if active journal receivers are attached. To later apply all journaled changes that have occurred since the last complete save operation, you must restore the receivers to the system from the SAVCHGOBJ media.

   For each journal identified in the previous step, do the following steps:

   a. Create a journal receiver that will be used as a temporary receiver. Give it a name that will identify it as a temporary receiver, for example, TEMPnn. You can enter a description in the text (TEXT parameter) that identifies it as a temporary receiver for disaster recovery.

      ```
      CRTJRNRCV JRNRCV(library-name/TEMPnn)
            TEXT('temporary journal receiver for journal xxx')
      ```

   b. To detach the current receiver and attach the new TEMPnn receiver, type the following and press the Enter key.

      ```
      CHGJRN JRN(library-name/journal-name) JRNRCV(library-name/TEMPnn)
      ```

   c. Delete the detached journal receiver (identified in the step where you wrote down the name of the journal and journal receiver) using the Delete Journal Receiver (DLTJRNRCV) command. (This allows the journal receivers on the SAVCHGOBJ media to be restored successfully.

      ```
      DLTJRNRCV JRNRCV(library-name/journal-receiver)
      ```

      If you receive message CPA7025 *Receiver never fully saved*, enter an **I** to ignore and press Enter to continue the delete.

Restoring Changed Objects

Load the SAVCHGOBJ tape and enter the following to restore changed objects.

```
RSTOBJ OBJ(*ALL) DEV(tape-device) SAVLIB(library-name)
        OBJTYPE(*ALL) ENDOPT(*LEAVE) MBROPT(*ALL) ALWOBJDIF(*ALL)
```

You must repeat this step for every library saved using the SAVCHGOBJ command. If you specified SAVCHGOBJ LIB(*ALLUSR), type the following to determine the libraries that were saved:

```
DSPTAP *SAVRST
```

If you are using journaling, perform the steps in "Applying Journal Changes" for each journal you wish to apply journal changes to. Otherwise, continue with "Restoring Changed Documents and Folders."

Applying Journal Changes

Note: If you are using OfficeVision/400 or PC Support/400 and are performing daily saves using SAVDLO and SAVCHGOBJ LIB(QUSRSYS) OBJJRN(*NO) commands, you must perform this step for the system supplied journal QUSRSYS/QAOSDIAJRN.

1. Determine the name of the last journal receiver (the last receiver restored) by entering the following:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

2. Press F15 (Work with receiver directory) to display the last receiver with a status of SAVED or PARTIAL. Write down the name of the receiver.

3. Create a receiver that follows the same naming convention as the last receiver but assign it a number of one greater.

   ```
   CRTJRNRCV JRNRCV(library-name/journal-receiver-nameNN)
   ```

   By doing this, you are doing what the CHGJRN command would normally do if the receiver was the current receiver being detached with a new receiver name being created. This allows your naming convention for journal receivers to continue.

4. Use the CHGJRN command to detach the temporary receiver and attach the new receiver you just created.

   ```
   CHGJRN JRN(library-name/journal-name)
         JRNRCV(library-name/journal-receiver-nameNN)
   ```

5. Determine the chain of receivers to be used in the APYJRNCHG command by entering the following command:

   ```
   WRKJRNA JRN(library-name/journal-name)
   ```

   Press F15 (Work with receiver directory) to display the receivers. Write down the first and last receiver that you restored (last receiver is prior to the TEMPnn receiver). Notice that the first and last is the same receiver if only one journal receiver was restored.

6. When applying journal changes and the ending receiver has a status of PARTIAL (saved while attached), the TOENT parameter requires that a sequence number be specified on the APYJRNCHG command. Determine the last entry to be applied by entering the following command for the last receiver (identified in previous step):

   ```
   DSPJRNRCVA JRNRCV(library-name/last-journal-receiver-name)
   ```

   Write down the value for the *Last Sequence Number* field.

7. Enter the following command to apply the journaled changes using the first and last journal receivers identified on the Work with Receiver Directory display.

   ```
   APYJRNCHG JRN(library-name/journal-name)
           FILE((library-name/*ALL))
           RCVRNG(lib-name/first-receiver lib-name/last-receiver)
           FROMENT(*LASTSAVE) TOENT(last-entry)
   ```

Attention

Do not apply journal changes to the document and folder search index database files (QAOSSH10 through QAOSSH19) for journal QAOSDIAJRN in library QUSRSYS. You must specify individual files on the FILE parameter instead of *ALL.

```
APYJRNCHG JRN(QUSRSYS/QAOSDIAJRN)
        FILE((QUSRSYS/QAOSAH05) (QUSRSYS/QAOKLY02)
             (QUSRSYS/QAOSAH07) (QUSRSYS/QAOKLY03)
             (QUSRSYS/QAOKDYX4) (QUSRSYS/QAOKNY06)
             (QUSRSYS/QAOKDY01) (QUSRSYS/QAOKDY04)
             (QUSRSYS/QAOKDY05) (QUSRSYS/QAOKDY07)
             (QUSRSYS/QAOKDY08) (QUSRSYS/QAOKDY09))
        RCVRNG(lib-name/first-receiver lib-name/last-receiver)
        FROMENT(*LASTSAVE) TOENT(last-entry)
```

Restoring Changed Documents and Folders

If you performed daily save operations for documents and folders, do the following steps. Otherwise, continue with the RSTAUT command.

1. Load the last daily SAVDLO tape.
2. If you performed daily save operations to backup all new folders and new and changed documents since the last complete SAVDLO operation, type the following and press the Enter key.

   `RSTDLO DLO(*ALL) DEV(TAP01) SAVFLR(*ANY) ALWOBJDIF(*ALL)`

   If you saved all documents referred to by the mail using SAVDLO DLO(*MAIL), type the following and press the Enter key.

   `RSTDLO DLO(*MAIL) DEV(TAP01)`
3. Restore users' authority by entering:

   `RSTAUT`

   The time it takes for the RSTAUT command to complete can vary significantly. The time depends on the number of user profiles and private authorities that were saved during the save operation.
4. This completes the restore operation.

   Power down the system by entering the following:

   `PWRDWNSYS OPTION(*IMMED) RESTART(*YES)`

   to perform a normal IPL and return the system to normal operations.

   Note: If you turned auto configuration off during the restore of the operating system, you must perform an IPL with the keylock switch in the **Normal** position.

Step 4. Restoring Programming Temporary Fixes

If you are using this procedure to finish restoring the entire system and have applied program temporary fixes (PTFs) after the last save operation, you must restore the PTFs.

If you need to restore program temporary fixes (PTFs) as part of recovering the entire system, find the most recent cumulative program temporary fix tape. This package could be on a distribution tape or on a stand-alone tape.

You can use option 31 (Install program temporary fix package) on the Work with Licensed Programs menu to install all of the PTFs. All of the PTFs in the cumulative PTF package will be installed for the licensed programs you have installed on your system. Refer to the *AS/400 System PTF Shipping Information Letter* for special instructions that are required.

If you are restoring individual PTFs, see the *System Operator's Guide*, SC41-8082, for more information about applying individual PTFs.

1. Print a list of all the licensed internal code PTFs currently on the system. Type the following and press the Enter key:

   ```
   DSPPTF LICPGM(5738999) OUTPUT(*PRINT)
   ```

2. Compare the list of PTFs in the previous step with the list you printed when you saved the system.

   Any PTFs that are not found must be loaded again.

3. If you do not have the PTFs identified in the previous step, order them and then apply them.

Section 8. Rebuilding Process

- Management Team must assemble to access damage and begin reconstruction of a new data center.
  - If the original facility must be restored or replaced, then the following are some of the factors to consider.
    1. What is the projected availability of all needed computer equipment?
    2. Will it be more effective and efficient to upgrade the computer systems with newer generation?
    3. What is the estimate of the probable time needed for repairs or construction of the data site.
    4. Is there an alternative site that could more readily be upgraded for computer purposes?
- Once the decision to rebuild the data center has been made, then go to "Section 11 Disaster Site Rebuild."

Section 9. Information Services Backup Procedures

- AS/400 System
  - Daily, journals receivers are changed at ____________ and at ____________.
  - Daily, a save of changes objects in the following libraries is done at ____________:
    - ____________
    - ____________
    - ____________
    - ____________
    - ____________
    - ____________
    - ____________
    - ____________

    This procedure also saves the journals and journal receivers.
  - On ____________ at ____________ a complete save of the system is done.
  - All save media is stored off-site in a vault at ____________ location.
- Personal Computer
  - It is recommended that all personal computers be backed up and the files loaded up to the AS/400 on ____________ at ____________ just before a complete save of the system is done. It is then saved with the normal system save procedure. This provides for a more secure backup of personal computer-related systems where a local area disaster could wipe out important personal computer systems.

Section 10. Testing The Disaster Recovery Plan

It is important in successful contingency planning to regularly test and evaluate the plan itself. Data Processing operations are volatile in nature, resulting in frequent changes to equipment, programs, documentation, etc. These actions make it critical to consider the plan a changing document.

The use of the following checklist for conducting a recovery test should be helpful.

Table C-7. Checklist For Testing The Disaster Recovery Plan

Item	Yes	No	Applicable	N/A	Comments
1. Conducting a Recovery Test					
a. Select the purpose of the test. What aspects of the plan are being evaluated?					
b. Describe the objectives of the test. How will you measure successful achievement of the objectives?					
c. Meet with management and explain the test and objectives. Gain their agreement and support.					
d. Have management announce the test and the expected completion time.					
e. Collect test results at the end of the test period.					
f. Evaluate results. Was recovery successful? Why or why not?					
g. Determine the implications of the test results. Does successful recovery in a simple case imply successful recovery for all critical jobs in the tolerable outage period?					
h. Make recommendations for changes. Call for responses by a given date.					
i. Notify other areas of results. Include users and auditors.					
j. Change disaster recovery plan manual as necessary.					
2. Areas to be Tested					
a. Recovery of individual application systems using files and documentation stored off-site.					
b. Ability to process on a different computer.					
c. Reload of system packs and IPL of system using files and documentation stored off-site.					
d. Ability of management to determine priority of systems when processing with limited.					
e. Ability to recovery and process successfully without key people.					
f. Ability of the plan to clarify areas of responsibility and chain of command.					
g. Effectiveness of security measures and security bypass procedures during the recovery period.					
h. Ability to accomplish emergency evacuation and basic first-aid responses.					
i. Ability of users of real-time systems to cope with a temporary loss of on-line information.					
j. Ability of users to continue day-to-day operations without "non-critical" processing.					
k. Ability to contact the key people, or their designated alternates, in a timely manner.					
l. Ability of data entry to provide input to critical systems using alternate sites and different input media.					
m. Availability of peripheral equipment and processing: printers, scanners, etc.					
n. Availability of support equipment: air conditioners, etc.					
o. Availability of support: supplies, transportation, communication.					
p. Distribution of output produced at recovery site.					
q. Availability of key forms and paper stock.					
r. Ability to adapt plan to lesser disasters.					

Section 11 Disaster Site Rebuild

- Floor plan of computer room.
- Determine current hardware needs and possible alternatives. (See configuration profile.)
- Data Center square footage, power requirements and security requirements.
  - Square footage _____________
  - Power requirements _____________
  - Security requirements - locked facility preferably combination lock on one door.
  - Floor to ceiling studding
  - High temperature, water, smoke, fire and motion detectors.
  - Raised floor.

Vendors

-
-
-
-
-

Floor Plan

Section 12. Record of Plan Changes

To get a list of local hardware, type the following:

```
DSPLCLHDW OUTPUT(*PRINT)
```

Attach the list here.

Appendix D. Initial Program Load (IPL) Process

This appendix describes the processing that can be performed during an IPL of your system. The AS/400 system provides a number of automatic recovery functions that are performed during the next IPL after an abnormal end of the system.

The following circumstances significantly affect an abnormal, or recovery, IPL time:

- The number of access paths that must be rebuilt
- The size of auxiliary storage
- The size of main storage
- The number of add and delete operations performed on files
- The number of objects on the system
- The number of journals on the system

The topics that follow describe in detail the recovery processing that can be performed relative to the above circumstances. This will help you understand why an abnormal IPL can take much longer than a normal IPL, and what recovery the system will automatically do for you.

Time Considerations for an Initial Program Load (IPL)

To shorten the time necessary to do an IPL of the system, do all of the following:

- Install an uninterruptible power supply. An uninterruptible power supply continues to run the system should main power fail and avoids having to load the system programs altogether.
- Specify both checksum protection and access path journaling.
- Save access paths.
- Use user auxiliary storage pools (ASPs). For best protection, place journal receivers and save files in different user ASPs than the files and journals.
- Use the delayed power off completion to power down a system that is not operating.
- Use the End Job Abnormal (ENDJOBABN) command to stop jobs that do not respond to an End Job (ENDJOB) request.
- If the configuration is stable, change the IPL performance system value QPFRADJ to '0'.

Description of Initial Program Load (IPL) Processes

The following list contains the steps the AS/400 system uses to do an IPL. IPL is divided into these stages:

- Service processor IPL
- Licensed internal code IPL
- OS/400 IPL

Only the licensed internal code IPL and OS/400 IPL steps perform special recovery processing after an abnormal end of the system. Therefore, only discussions of the recovery processing that may be performed by these two steps is provided.

Licensed Internal Code IPL

The licensed internal code stage of the the IPL communicates with the operator through status system reference codes (SRCs) on the control panel. Each step has a unique SRC associated with it. Some of the steps may not be performed or are performed so quickly that the operator may not see the status SRCs on the control panel.

IPL processing for the licensed internal code performs the following recovery processing steps during the next IPL after an abnormal end of the system.

- **SRC C6xx 4021** Recovery (storage management recovery startup)

  This SRC will be shown briefly while some recovery start up is performed.

- **SRC C6xx 4204** Recovery (main storage dump synchronization)

  This step is started if mirrored protection is in effect for the system ASP and a prior IPL resulted in a main storage dump being taken. The main storage dump was written to only one of the disk units of a mirrored pair. During this step, the main storage dump is copied to the other disk unit of a mirrored pair.

- **SRC C6xx 4205** Recovery (mirror synchronization)

  This step is performed when mirrored protection is in effect and one or more of the following conditions exists:

  - Data on one of the load source disk units (unit 1) was changed before the other load source disk unit was available for use. In this case, the changes made to the first load source disk unit are copied to the second load source disk unit. However, if the status of one of the load source disk units is *suspended*, the changes are not copied until the *suspended* load source disk unit is *resuming*.
  - The status of one of the disk units of a mirrored pair is *active* and the status of the other disk unit of the same mirrored pair is *resuming*. In this case, the data is being copied from the *active* disk unit to the *resuming* disk unit.
  - An abnormal end of the system occurred while one or more write operations were in progress for a disk unit that has mirrored protection. In this case, the licensed internal code ensures that the areas of the disk unit being written contain the same data by copying the data from one disk unit of the mirrored pair to the other disk unit of the same mirrored pair.

- **SRC C6xx 4210** Recovery (checksum protection — subset validation)

  After an abnormal end with checksum protection in effect, and when 4230 Recovery is not required to start again, checksum data must be validated. If a copy of main storage is available on unit 1, licensed internal code can determine what operations were in progress and complete them to assure that all checksum data is correct. This part of the validation process requires little time. If a copy of main storage is not available on unit 1, complete validation is done later with 4260 or 4270 Recovery.

- **SRC C6xx 4220** Recovery (checksum configuration change)

  This step is started when checksum protection is first started or when a new disk unit is added to an existing checksum set. When checksum protection is started, the system will put each unit in a checksum set. The number of units in checksum sets determines how long this process will be.

  When checksum protection is first configured, the following occurs:

  - All disk units in the system ASP must be formatted.
  - The proper disk layout must be established (such as checksum stripes, protected storage areas, unprotected storage areas, control information).
  - The initial checksum data must be calculated and written to disk.

  When a disk unit is added to an existing checksum set, the following must be done:

  - The disk unit must be formatted.
  - The layout of the previously existing disk units in the set must be adjusted (although these previously existing disks are not initialized).

- **SRC C6xx 4230** Recovery (checksum protection — data recovery)

  The Save Disk Unit Data option of the disk device recovery function of DST allows a service representative to copy the contents of a disk unit. It is not intended as a method for normal backup. It is important that a service representative attempts to copy data from a failed unit before the system IPL is done. If it is successful in copying data from a failed unit, recovery time for the next IPL is significantly reduced by copying the saved data back to the replacement unit.

  The Save Disk Unit Data option marks unrecoverable sectors with a special code that 4230 Recovery uses to start the rebuilding of data in that sector. If it is not run, or if it fails to complete, all protected sectors are rebuilt on the replacement unit.

  If a disk unit must be replaced, and the save operation is not entirely successful, the data in sectors that could not be saved must be rebuilt from checksum data stored on other disk units in the set.

Checksum protection does not cancel the requirement to recover access paths. It primarily avoids auxiliary storage initialization and reloads from a backup copy after loss of data on a failed disk unit.

- **SRC C6xx 4240** Recovery (reclaim main storage copy)

  If a copy of main storage is available on the disk unit that contains storage unit 1, changed pages in main storage are written to their proper locations on disk.

- **SRC C6xx 4250** Recovery (subset directory recovery)

  If 4240 is successful, the storage management directories are updated. If this step is successful, 4260 is unnecessary.

- **SRC C6xx 4260** Recovery (storage directory rebuild)

  Storage management maintains two directories to keep track of disk use: the **permanent directory** (for all permanently-stored objects) and the **free-space directory** (to keep track of free space on disk). An entry in the permanent directory consists of a segment identifier (SID) and up to 4 blocks of sectors next to each other (called *extents*). An entry in the free-space directory consists of a single extent only (no SID.)

  This recovery step starts simultaneous processes to rebuild directories. Each process reads 64 sectors and then decides if the first sector is the beginning of an object extent or a free-space extent.

  If the sector is the beginning of an object extent, an entry is made in the permanent directory. Because the first sector of an object extent includes the number of sectors in the extent, processing continues at the sector following the extent.

  If the extent is the beginning of a free-space extent larger than 4M, 8M, or 16M, an entry is made in the free-space directory. Because the length of the sector is recorded in the first sector, the next read operation starts at the sector following the free-space extent.

  If the sector is the beginning of a free-space extent of less than 4M, the size of the extent is not recorded anywhere. All sectors in the extent must be read to find the end. When the end is found, an entry is made in the free-space directory, and the process continues.

  Notes:

  1. A free-space extent of greater than 16M is recorded as multiple extents of 16M or less.
  2. If 4260 Recovery is done, 4025 Recovery is also done.
  3. If the last system end was abnormal and a copy of main storage was saved to the disk unit that contains storage unit 1, the system will attempt to use the copy to shorten this step.
  4. If the last system end was abnormal and the system was running with checksum protection in effect, and checksum recovery (4230) was not required; then checksum data validation must be run. If a copy of main storage was saved, little time is required to validate checksums. Otherwise, step 4260 can run up to three times longer to perform validation. In rare cases where 4260 is not required after an abnormal end, checksum validation is performed by a special checksum validation step (4270).

     This recovery step initializes disk devices as needed if ASP configuration changes require it.

- **SRC C6xx 4260 or 4270** Recovery (checksum protection — validation)

  After an abnormal end with checksum protection in effect, and when 4230 Recovery is not required to start again, the checksum data must be validated. If a copy of main storage is not available on the disk unit that contains storage unit 1, a complete validation is done during 4260 Recovery. This validation takes up to three times as long as 4260 Recovery without checksum protection because adjacent blocks of sectors that could contain permanent data or checksum data cannot be bypassed.

  In the rare cases for which complete checksum data validation is required but 4260 Recovery is not required, the validation is performed during 4270 Recovery.

- **SRC C6xx 4280** Recovery (get space on the disk unit that contains storage unit 1)

Main storage is copied to the load source device on selected abnormal ends related to external power failures with an uninterruptible power supply installed, or related to failures of disk units.

To provide space for this copy, space as large as the system main storage is allocated on the disk unit that contains storage unit 1. This process normally takes place at the following times:

- First IPL after increasing the size of main storage.
- First IPL after initializing the disk unit that contains storage unit 1 and installing the system licensed internal code.

• **SRC C6xx 4023** Recovery (rebuild damaged contexts)

When an object is created, renamed, or deleted in a library, or if an object is moved between two libraries, and if an abnormal end occurs before the library pages are forced to auxiliary storage, the contexts involved require rebuilding. The amount of time for potential error is usually quite small, so context rebuilding is rarely required.

If rebuilding is required, the permanent directory is read to locate all objects. The first page of each object is read to get the library SID. This library is then checked to verify an entry for that object. If the object is not in the library, an entry is made for it. If the library itself does not exist, the object can be recovered with the RCLSTG command.

• **SRC C6xx 4025** Recovery (authorization rebuild)

User Profile Recovery

This step is done every time the system performs an IPL. Any partially completed user profile table or index rebuild operations are completed. Any user profiles that have not been reformatted to a capacity of approximately 1 000 000 entries are reformatted.

Recovery From an Abnormal End

This recovery step is started for any of the following reasons:

- The system performs an IPL for the first time with Version 2, Release 1 (or later) of the licensed internal code.
- The system performs an IPL following an abnormal system end, where the machine interface (MI) boundary was not reached.
- The system performs an IPL and either 4260 Recovery or permanent address recreation is done.

Because the above conditions can result in storage accounting inaccuracies, all object sizes and auxiliary storage owned by user profiles are suspicious until verified or corrected by the system during normal usage.

The object size verification or correction occurs the first time the object is referenced in an MI instruction following the abnormal system end. User profile owned auxiliary storage is verified or corrected by the first use of either the MATAUOBJ or RECLAIM MI instruction; until that occurs, the amount of auxiliary storage owned by a given user profile may appear incorrect.

The following VLOG entries may be created:

0B00 0009	The current IPL is considered abnormal for any of the reasons previously mentioned. Object size and auxiliary storage owned by user profiles will be verified and corrected as they are used.
0B00 0001	An object's size was corrected.
0B00 0008	An object's associated space size was corrected.
0B00 0002	The auxiliary storage owned by a user profile was corrected.
0B00 000B	The auxiliary storage owned by a user profile was verified and found to be corrected.

The following information is used to perform authority recovery the first time the object is referenced.

- The system maintains the IPL number of the last abnormal IPL. This is set to the current IPL number when the IPL is considered abnormal for any of the reasons previously mentioned.
- Each user profile contains a field totalling the amount of recovered auxiliary storage owned by the user profile. This field is reset to zero during each

abnormal IPL. The field that contains the *materialized* value for the amount of owned auxiliary storage is left alone, and may or may not be accurate.

— The header of each object contains the object authority recovery IPL number. If this number is less than the system abnormal IPL number, then when the object is first referenced it is verified or corrected as follows:

- The object size is corrected if it is incorrect.
- The object size is added to the *recovered* field for the user profile that owns the object.
- The object authority recovery IPL number is updated to contain the current IPL number.

As *recovered* objects are truncated, extended, or deleted, or the owning user profile is changed, changes are made to both the *recovered* and *materialized* fields for the auxiliary storage of the owning user profile.

— Each user profile contains the IPL number under which the auxiliary storage owned by the user profile was verified or corrected. If that IPL number is less than the last abnormal system IPL number when the MATAUOBJ instruction requests owned objects, then the auxiliary storage owned by the user profile is verified and corrected.

1. The size of each object owned by the user profile is verified or corrected as described above if it has not been done since the last abnormal IPL.
2. The *recovered* owned auxiliary storage size replaces the *materialized* owned auxiliary storage size. The amount of owned auxiliary storage will now appear correct.
3. The IPL number in the user profile is updated with the current IPL number to indicate that the auxiliary storage owned by the user profile has been verified or corrected.

When the RECLAIM instruction is run, the above is done for all user profiles that have not been verified or corrected since the last abnormal IPL.

Object Domain Initialization

If this is an install IPL, and object domains have not been initialized, a background job is started to initialize the domain for each object that has not already had its domain established. It is hoped that this will complete asynchronously during installation in order to avoid synchronous initialization on the next IPL.

If this is not an install IPL, and object domains have not yet been initialized, then the initialization takes place synchronously in this recovery step. In addition, all object size and user profile owned auxiliary storage verification and correction takes place synchronously as well.

- **SRC C6xx 4026** Recovery (journal recovery)

  This recovery step validates the network of objects that supports journaling. It verifies that each file being journaled has a usable journal and usable journal receivers. For two receivers, the system will select which receiver to use for possible recovery.

- **SRC C6xx 4027** Recovery (database recovery)

  The system determines which database files and access paths are not synchronized. If the last system end was abnormal and a copy of main storage was saved, the copy will be used for this analysis.

  The licensed internal code uses an internal recovery table known as the **database in-use table** to limit the number of database files subjected to recovery. This table resides on disk and is readable.

  If the database in-use table is usable:

  — Use it to locate all database files that were open at the last system end and to verify whether the physical files and access paths are synchronized.

  — For those not synchronized, mark the access path as not valid (to be rebuilt later).

  — If any delete operations have been done since the last time the physical file was forced to disk, read all the records in the file to get an accurate count.

- If any add operations have been done since the last time the physical file was forced to disk, read all recently added records to get an accurate count.

  Because the database in-use table resides on disk, it is possible that a portion of the disk surface containing this table is no longer readable.

  Approximately 90 seconds are required to read each 16M of records.

  If the database in-use table is *not* usable, locate all database objects by reading the object directory built during 4025 Recovery and perform the steps for 4027. If 4025 Recovery found it unnecessary to build an object directory, build it now; then perform the steps for 4027.

- **SRC C6xx 4028** Recovery (journal synchronization)

  A **journal synchronization point** is caused when a file is closed. This recovery step locates the physical files being journaled by each journal, locates the last synchronization point for each physical file, then starts an APYJRNCHG command.

  The journaling function assures there are no more than approximately 50 000 journal entries from the oldest synchronization point to the present, automatically limiting the number of journal entries that must be applied during recovery. When the fifty thousand threshold is reached, an implicit synchronization point is established.

  If access path journaling is in effect, the journal is read from the last synchronization point for the journaled access path. This is done before images presenting the original state of access path pages are written to the existing access path, thereby returning it to the condition existing at the last synchronization point. In addition, a delayed maintenance log entry is made for each journal entry following any synchronization point, which represents an access path change. Thus, access path rebuilding during the OS/400 IPL becomes a delayed maintenance rebuilding of the changes that occurred since the last synchronization point.

- **SRC C6xx 4029** Recovery (commitment control recovery)

  The system performs ROLLBACK operations to the previous commitment boundary for any open commit cycles for jobs that were active at the time the system ended abnormally.

- **SRC C6xx 4030** Recovery (update object recovery list)

  The object recovery list is a table built during this stage of the IPL. It serves as the repository for information about objects that the OS/400 IPL must recover. During this step of the IPL, the table is updated to identify database objects that will be recovered.

- SRC C6xx 4031 Recovery (journal cleanup)

  This recovery step reinitializes the journal's internal tables to show that all journaled files have been synchronized.

- SRC C6xx 4032 Recovery (commitment control cleanup)

  This recovery step empties commit blocks of runtime values, such as locks.

OS/400 Initial Program Load (IPL)

The OS/400 part of the IPL communicates status to the operator by displaying system reference codes (SRCs) on the control panel. Each step has a unique SRC.

During an attended OS/400 IPL, the system also communicates with the operator through messages that appear as *BREAK messages sent to the QSYSOPR or as status messages sent to the work station message queue. If the SEVERITY filter is set too low on QSYSOPR, the messages are shown, and the IPL process is suspended until the operator exits the message display by pressing the Enter key or F3. When the controlling subsystem is started, QSYSOPR is left in the *BREAK type of operation to the console.

IPL processing for the OS/400 program performs the following processing steps.

1. **SRC C9xx 28C0** verifies the work control block table.

   The work control block table has an entry for each job on the system. Each entry contains pointers to the following permanent objects:

   - Job message queue
   - Spool control block
   - Local data area

This recovery step first checks the work control block table for damage.

If the work control block table has no damage, this recovery step verifies the existence of the job message queue and spool control block for the IPL start control program function (SCPF) job, then continues. The SCPF job does not have a local data area.

If the work control block table has damage, or if clear job and output queues was specified during the OS/400 install operation, this recovery step rebuilds the work control block table with an entry for only the IPL SCPF job. In addition, because the work control block table no longer relates the job message queues, spool control blocks, and load data areas to specific jobs, they must be deleted. These objects are not in a library so the QSYS user profile is searched to find them for later deletion during the work control block table processing step.

2. **SRC C9xx 28C0** retrieves the object recovery list.

   The object recovery list was built during the licensed internal code IPL and contains the information regarding objects that the IPL SCPF job must deal with during its portion of IPL.

3. **SRC C9xx 2910** verifies selected message queues.

   The message handler validates the existence of the QSYSOPR and QHST message queues, and re-creates those it finds missing or damaged. If clear job and output queues has been specified, both message queues are re-created. The QSYSOPR message queue is cleared and its size reduced to its original allocation.

   The QSYSOPR message queue is also reduced to its original allocation whenever all messages are cleared (that is, by pressing F13 on the Message Queue display).

4. **SRC C9xx 2920** – library cleanup.

5. **SRC C9xx 2940** – console configuration. Creating QCTL and QCONSOLE and varying on during an attended IPL.

6. **SRC C9xx 2960** – attended IPL processing. Attended IPL processing occurs at this point and the IPL displays appear. For more information, see the discussion on starting the system in the *Operator's Guide*.

7. **SRC C9xx 2970** – recovers databases.

   This recovery step verifies the object recovery list and sends a message to QHST for each database file that was being used or changed during an abnormal system end, and has not yet been recovered.

   This recovery step sends a message to QHST for each journal object and each commitment control object that could not be successfully recovered. It also writes the commit ID to the user's notify object.

   This recovery either rolls back or completes certain uncompleted operations. Each of the following functions keeps an audit trail of its progress in the QRECOVERY library to be used in case of an abnormal system end.

   - Rolled back:
     - Add member
     - Create file
   - Completed:
     - Change file
     - Change object owner
     - Delete file
     - End journal access path
     - End journal physical file
     - Grant object authority
     - Start journal access path
     - Start journal physical file
     - Move object
     - Recover SQL view
     - Rename member
     - Remove member
     - Rename object
     - Restore file
     - Revoke object authority

8. **SRC C9xx 2970** – recovers access paths.

   Journaled access paths are recovered at this time if MAINT(*IMMED) is specified for the file. A *STATUS message appears for each access path as it is being built. Recovery can usually be accomplished very quickly because only the changes that have occurred since the last synchronization point need to be applied.

   This recovery step shows the Edit Rebuild of Access Paths display only during an attended IPL, and only if files with keyed access paths are in a nonsynchronized condition (they require rebuilding) and were not journaled. Any open keyed access path that has had at least one change will generally be in a nonsynchronized condition unless the path is journaled. Functions that can force synchronization include:

- RPG III FEOD operation
- FRCRATIO of 2 or greater
- FRCACCPTH
- Full CLOSE

Each of the functions just listed causes the index header page to also be forced to disk with the synchronized bit set on. On the next unforced index change, the index header page is again forced to disk with the synchronized bit set off.

```
                      Edit Rebuild of Access Paths                 RCHAS331
                                                         05/12/90  13:49:34

IPL threshold . . . . . . . . . . . . . 88_  0-99

Type sequence, press Enter.
Sequence:  1-99, *OPN, *HLD

                      ------------Access Paths----------  Unique  Estimate
Seq   Status          File       Library     Member       Keyed   Time
25__  IPL             FILE234512 LIBRARY111  MBR1234567   No      00:00:56
25__  IPL             FILE234513 LIBRARY111  MBR1234567   No      00:00:56
75__  IPL             FILE234514 LIBRARY111  MBR1234567   Yes     00:00:56
75__  IPL             FILE234515 LIBRARY111  MBR1234567   Yes     00:00:56
88__  IPL             FILE234516 LIBRARY111  MBR1234567   No      00:00:56
99__  AFTIPL          FILE234517 LIBRARY111  MBR1234567   Yes     00:00:56
*OPN  OPEN            FILE126789 L123456789  MBR4567890   Yes     12:34:56
*OPN  OPEN            FILE346789 L123456789  MBR4567890   No      12:34:56
*HLD  HELD            F123336789 L123456789  MBR4567890   No      10:30:06
*HLD  HELD            F123456789 L123456789  MBR4567890   Yes     99:56:01
                                                               More...

F5=Refresh   F11=Display member text   F13=Repeat all   F15=Sort by
F16=Repeat position to   F17=Position to
```

For additional information on recovering access paths, refer to the *Database Guide*.

On the Edit Rebuild of Access Paths display, the user can select which access paths are to be rebuilt during the IPL and which can be built later. Those paths that must be rebuilt during the IPL can be done at a rate of approximately 10 000 index entries per minute (this rate can vary significantly, depending on key size, processing unit speed, and the amount of main storage available for the function).

Rebuilding access paths at IPL time is always the fastest because no other processes are running, and no main storage pools have yet been established (that is, all of main storage is available for the function).

The estimated rebuild times apply only to access paths that are rebuilt during the IPL when the system is in a restricted state.

Rebuilding access paths after the IPL is usually the slowest because it is done in the BASE pool (usually the smallest pool). The function must share the processing unit with other processes and is running at low priority.

Rebuilding access paths at OPEN time is done in the pool of the requesting process and so may be faster than rebuilding after the IPL.

Because rebuilding tends to be process-bound, no benefit can be expected from attempting multiple rebuild processes at the same time on a single processor.

9. **SRC C9xx 2990** — adjusts pool sizes and activity levels, if requested.

   If the system value QPFRADJ is set to '1' (the default), the machine pool, *BASE pool activity level, pool and activity level for spooled jobs, and pool size and activity level for interactive jobs are determined. Subsystem descriptions QSPL, :QINTER, and QBASE are changed to use *INTER and *SPOOL shared pools. The shared pools are changed to reflect the calculated values.

10. **SRC C9xx 29B0** — processes spooled files.

    The QSPL library is searched to get the names of all files and members in those files. Spooled file data is stored within members of a physical file in the QSPL library. When a spooled file is no longer in use, the associated member is kept for possible reuse by other spooled files (thus avoiding the overhead of Remove Member (RMVM) and Add Physical File Member (ADDPFM) commands). However, if a spooled file is not reused after seven IPLs, it is removed. To control the extra time required for this, up to 200 members are removed on each IPL, and an *STATUS message appears during the process. If the last system end was abnormal, this removal process is bypassed.

    All libraries on the system are searched to get the names of all job queues and all output queues. The status in the object header for each queue is updated from this information.

If the default output queue associated with a device has been damaged, the output queue will be deleted and re-created by using the CRTOUTQ command.

11. **SRC C9xx 29C0** — processes work control block table.

    This step reorganizes the work control block table:

    - If the work control block table was not rebuilt:

      Verify if the three permanent objects exist for each work control block table entry. For the objects that represent jobs, determine the status or location of the job. If a job was active, close all related spooled files. If clear incomplete job logs was specified on the IPL Options display, the job message queue is cleared for each job that was active at the last system end.

      Compress the work control block table to improve the job search time and reduce the storage used (delete the unneeded job message queues, spool control blocks, and load data areas). During normal operations, the work control block table grows to the size necessary to accommodate the greatest number of *JOBQ, *OUTQ, and active jobs at any one time. Compression occurs only during the IPL, and then only to value specified for the initial total number of jobs (QTOTJOB system value) and sufficient additional total number of jobs (QADLTOTJ system values) to account for all jobs that will exist when the IPL is complete (primarily *JOBQ, *OUTQ, and incomplete job log jobs).

    - If the work control block table was rebuilt:

      Delete all job message queues, spool control blocks, and local data areas that existed when it was discovered that the work control-block table required rebuilding. Create a job message queue, spool control block, and local data area for each of the jobs specified in the system values QTOTJOB and QADLTOTJ.

12. **SRC C9xx 2A80** — creates temporary job structures.

    Processing access groups (PAGs) and QTEMP libraries are created for the number of jobs represented by the sum of the system values QACTJOB and QADLACTJ.

13. **SCR C9xx 2B20** — automatic configuration, if requested.

    If the system value QAUTOCFG is set to '1' (the default). the local configuration elements, as determined during the IPL, are compared to the corresponding control unit and device descriptions. The differences are corrected.

14. **SRC C9xx 2B30** — start QLUS job.
15. **SRC C9xx 2B40** — vary on ONLINE(*YES) configuration.
16. **SRC C9xx 2C20** — DIA recovery.
17. **SRC C9xx 2C30** — SNADS recovery.

Simultaneous Initial Program Load

Several IPL steps take place after the controlling subsystem is started (for example, parallel with normal user processing):

1. Clear QRPLJOB.
2. Additional DIA recovery.
3. Access path rebuilding specified for *AFTIPL (in the SCPF job running at RUNPTY 52) takes place at the same time as normal user processing.
4. Create command analyzer work spaces, two for each active job as represented by the sum of the system values QACTJOB and QALDACTJ.
5. Complete job logs for jobs which were not completed normally during the previous system end.

   Note: During an attended IPL, there is an option to clear these incomplete job logs. The default is *NO.

Licensed Internal Code Completion

Forced licensed internal code completion (delayed off) allows the system to complete AS/400 machine interface instructions (MI instructions) before ending the system abnormally.

Without forced licensed internal code completion, when an MI instruction is not allowed to complete, the system and user objects it refers to may be marked as unusable. At the next IPL, these objects are examined and a recovery process is required to make them usable again. (For example, access paths marked as unusable will have to be rebuilt.) However, when you use forced licensed internal code completion, the system attempts to complete the MI instruction(s). If completed, the objects referred to by it are marked as usable, and a long recovery process is avoided.

Using Forced Licensed Internal Code Completion

To use this function, turn the Power switch, not the Unit Emergency switch on the control panel to the Delayed Off position. The keylock must be in the Manual position. This requests that the system do a delayed power down of the entire system. The actual power down is delayed while the machine attempts to complete all interrupted MI instructions. After the forced licensed internal code completion processing is finished, the power is turned off.

Note: If you do an IPL of the system, or switch off the emergency power without first turning off the power, you will cause an immediate interruption of machine processing. This does not provide any delay to force licensed internal code completion and can result in significant additional recovery processing due to the immediate interruption of machine processing that occurs.

If every interrupted instruction can be completed, the amount of time needed to make every object usable can be significantly reduced.

How Forced Licensed Internal Code Completion Ends: Forced licensed internal code completion will stop if:

- All interrupted instructions are completed within 16 minutes, and the machine shuts down normally.
- The system is unable to successfully complete all MI instructions within 16 minutes. In this case, a main storage copy is performed and the machine automatically shuts down. The main storage copy is used on the next IPL to avoid directory recovery and minimize database recovery. In this case, some access paths may be marked as unusable. That is, the MI instructions that could not complete successfully may have been performing an operation on an access path when they were stopped. When you restart the system, it must then rebuild any unusable access paths (unless they are journaled), and perform all normal recovery steps. The IPL is labeled abnormal to allow certain OS/400 functions to perform additional recovery operations.

Normally, there are few invalidated access paths that are not usable if interrupted machine interface (MI) instructions are completed. If access paths are unusable, the Override Access Path Recovery display is shown allowing you to select how the system will rebuild these access paths. (If all access paths are unusable, this display is not shown.)

Appendix E. Main Storage Dump Space Not Available (CPI0987)

It is assumed that you are at this procedure because you have received message CPI0987: `Main storage dump space is not available.`

The system does not have sufficient free space in the system ASP (ASP 1) for the operating system to reserve main storage dump space.

For main storage dump space, the system reserves 1MB of auxiliary storage for each 1MB of main storage. The main storage dump space is used to isolate problems and reduce recovery time. Failure to provide sufficient main storage dump space may critically affect service and system recovery time.

Not having enough main storage dump space can be caused by the following:

- On all systems, insufficient disk storage space in the system ASP (ASP 1).
- On 9406 Model B70, D70, and D80 system units, insufficient disk storage space on units eligible to contain main storage dump space.

Determining the Cause for Insufficient Main Storage Dump Space

Task Overview

1. Display the disk configuration capacity
2. Determine the problem

To determine the cause of insufficient main storage dump space, do the following:

Start SST by typing the following:

```
STRSST
```

Press the Enter key. The following display is shown.

```
                    System Service Tools (SST)

Select one of the following:

     1. Start a service tool
     2. Work with active service tools
     3. Work with disk units
     4. Work with disk data recovery











Selection
     3

F3=Exit        F10=Command entry        F12=Cancel
```

Step 1. Display the Disk Configuration Capacity

1. Select option 3 (Work with Disk Units) and press the Enter key. The following display is shown.

```
                       Work with Disk Units

Select one of the following:

     1. Display disk configuration
     2. Display checksum configuration
     3. Calculate checksum configuration
     4. Work with ASP threshold
     5. Work with disk unit recovery
     6. Work with disk unit information
     7. Calculate mirrored capacity








Selection
     1

F3=Exit        F12=Cancel
```

2. Select option 1 (Display disk configuration) on the Work with Disk Units display and press the Enter key. The following display is shown.

```
                    Display Disk Configuration

Select one of the following:

     1. Display disk configuration status
     2. Display disk configuration capacity
     3. Display disk configuration protection
     4. Display non-configured units

Selection
     2

F3=Exit          F12=Cancel
```

3. Select option 2 (Display disk configuration capacity) and press the Enter key.

Step 2. Determine the Problem

The following display is shown.

Note: The example of the Display Disk Configuration Capacity display (Figure E-1) is for a system that has mirrored protection for the system ASP (ASP 1). For a system without mirrored protection, use the value in the *Unprotected % Used* column to determine the problem.

```
                    Display Disk Configuration Capacity

                                                   --Protected--  --Unprotected--
ASP  Unit  Type  Model  Threshold  Overflow       Size  %Used       Size  %Used
 1                            90%   No             2354 81.04%         0  0.00%
        1  2800   001                                 0  0.00%         0  0.00%
        1  2800   001                                 0  0.00%         0  0.00%
        5  9336   020                                 0  0.00%         0  0.00%
        5  9336   020                                 0  0.00%         0  0.00%
        6  9336   020                                 0  0.00%         0  0.00%
        6  9336   020                                 0  0.00%         0  0.00%

Press Enter to continue.

F3=Exit   F5=Refresh    F11=Display non-configured units     F12=Cancel
```

Figure E-1. Display Disk Configuration Capacity Display

- *ASP*. The auxiliary storage pool number.
- *Unit*. The number assigned by the system to identify the disk unit.

- *Type*. The disk unit type assigned by the manufacturer.
- *Model*. The disk model that identifies the feature level for a specific type of disk unit.
- *Threshold*. The current threshold. The system notifies you when this percentage of storage is full.
- *Overflow*. Indicates if this ASP is overflowed.
- *Protected Size*. Amount of storage in the ASP that is protected by checksum or mirrored protection.
- *Protected % Used*. Percent of protected size that is used.
- *Unprotected Size*. Amount of storage in the ASP that is unprotected.
- *Unprotected % Used*. Percent of unprotected size that is used.

If the value shown in the *Protected % Used* column for ASP 1 is greater than 97%, then the problem is insufficient disk storage space in the System ASP. Use "Dump Space Recovery Procedure 1" to recover.

If the value shown in the *Protected % Used* column for ASP 1 is less than 97%, then the problem is insufficient disk storage space on units eligible to contain main storage dump space. Use "Dump Space Recovery Procedure 2" on page E-5 to recover.

Recovery Procedures

The following recovery procedures are used when there is not enough main storage dump space available in the system ASP.

Dump Space Recovery Procedure 1

Use one of the following options to recovery.

Option 1. Reduce Storage Use and Perform an IPL

Use the following to reduce storage:

- Delete any unused objects.
- Save the old versions of the system log QHST that are not being used and then delete them.
- Print or delete spooled output files on the system.
- Save objects by specifying STG(*FREE) on the save command.

Option 2. Add Storage Units to the System ASP

- If the system does not have mirrored protection, use the procedure "Adding Units to an Existing ASP" on page 15-17.
- If the system has mirrored protection, use the procedure "Adding Disk Units to a Mirrored ASP" on page 22-1.

Dump Space Recovery Procedure 2

There are two options for this recovery procedure, depending on the type of hardware:

1. Option 1 is the recovery procedure for the 9406 model B70 (stage 1) system unit.
2. Option 2 is the recovery procedure for the 9406 model D70 or D80 system units.

Option 1. Stage 1 Hardware, 9406 Model B70

To provide the additional main storage dump space, you must do the following:

1. Save the entire system using the procedure "Saving the Entire System" on page 10-8.
2. Upgrade the current load source device by contacting your service representative.
3. Install the licensed internal code using the "Procedure for Installing or Restoring Licensed Internal Code Using the SAVSYS or Distribution Tapes" on page 10-14.
4. Recreate any user ASPs using the procedure "Creating a User ASP and Adding Disk Units to the New User ASP" on page 15-1.
5. Restore the operating system using "Restoring the OS/400 Licensed Program" on page 10-19.
6. Restore the remaining parts of the system using "Restoring User Profiles, Device Configurations, User Libraries, and Authority" on page 10-33.

If the system ASP had mirrored protection, a mirrored unit must be provided for pairing with the new load source unit. After the ASPs are recreated, start mirrored protection on the ASPs that previously had mirrored protection.

Option 2. Stage 2 Hardware, 9406 Model D70 or D80

To provide for additional main storage dump space, you must add at least 1 2800-001 storage unit to the system ASP (ASP 1). If all non-load source 2800-001 storage units are configured to user ASPs, you must save the the user ASP that the unit is being moved from, move the unit, and then restore the user ASP.

- If the system does not have mirrored protection:
  - To move a disk unit from one user ASP to another, use the procedure, "Moving a Disk Unit from One User ASP to Another ASP" on page 15-57.
  - To add units to an existing ASP, use the procedure "Adding Units to an Existing ASP" on page 15-17.
- If the system has mirrored protection:
  - To move a disk unit from a mirrored ASP to another, use the procedure "Moving a Disk Unit from a Mirrored ASP to Another ASP" on page 22-9.
  - To add units to a mirrored ASP, use the procedure "Adding Disk Units to a Mirrored ASP" on page 22-1.

Bibliography

This section lists publications that provide additional information about topics described or referred to in this manual. The manuals in this section are listed with their full title and order number, but when referred to in text, a shortened version of the title is used.

Programming Information

- *Communications and Systems Management Guide (Alerts and Distributed Systems Node Executive)*, SC41-9661
  Short Title: *Alerts and DSNX Guide*

  This manual provides the system operator, programmer, or system administrator with information for configuring the AS/400 system to use the remote management support.

- *Communications: Distribution Services Network Guide*, SC41-9588
  Short Title: *Distribution Services Network Guide*

  This manual provides the system operator or system administrator with information about configuring a network for Systems Network Architecture Distribution Services (SNADS) and the Remote Spooling Communications Subsystem/Professional Office System (RSCS/PROFS) bridge. In addition, object distribution functions and document library and distribution services are discussed.

- *Communications: Intersystem Communications Function Programmer's Guide*, SC41-9590
  Short Title: *ICF Programmer's Guide*

  This manual provides provides the AS/400 programmer with information to write application programs that use the AS/400 communications and OS/400 intersystem communications function.

- *Device Configuration Guide*, SC41-8106
  Short Title: *Device Configuration Guide*

  This manual provides the system operator or system administrator with information on how to do an initial configuration and how to change the configuration. This manual also contains conceptual information about device configuration.

- *Licensed Programs and New Release Installation Guide*, SC41-9878
  Short Title: *Licensed Programs and New Release Installation Guide*

  This manual provides the system operator or system administrator with step-by-step procedures for initially installing, installing licensed programs, program temporary fixes (PTFs), and secondary languages from IBM.

  This guide is also for users who already have an AS/400 system with an installed release and want to upgrade to a new release.

- *Migrating from System/38 Planning Guide*, GC41-9624
  Short Title: *Migrating from System/38 Planning Guide*

  This manual provides the application programmer, system administrator, or data processing manager with information to help them migrate their products and applications using the System/38 to AS/400 Migration Aid. It includes information for planning the details of migration and an overview of the functions on the System/38 Migration Aid.

- *Office Services Concepts and Programmer's Guide*, SC41-9758
  Short Title: *Office Services Concepts and Programmer's Guide*

  This manual provides information about writing applications that use OfficeVision/400 functions. This manual also includes an overview of directory services, document distribution services, document library services, document and folder save and restore and storage management, security services, word processing services, and information on finding new ways to integrate your applications with OfficeVision/400.

- *Physical Planning Guide and Reference*, GA41-9571
  Short Title: *Using OfficeVision/400* Word Processing*

  This manual provides provides the data processing manager, system administrator, and installation planning representative with information for planning to set up the AS/400 system. This guide also includes information on cable considerations, physical specifications, electronic customer support (ECS), and unpacking considerations.

Operations

- *New User's Guide*, SC41-8211
  Short Title: *New User's Guide*

  This manual provides display station operators with information about how to sign on and off; send and receive messages, respond to keyboard error messages, use function keys; and use display, command, and help information to control and manage their own jobs.

- *System Operator's Guide*, SC41-8082
  Short Title: *Operator's Guide*

This manual provides the system operator or system administrator with information about how to use the system unit control panel and console, send and receive messages, respond to error messages, start and stop the system, use control devices, work with program temporary fixes (PTFs), and process and manage jobs on the system.

- *Programming: Control Language Programmer's Guide*, SC41-8077
  Short Title: *CL Programmer's Guide*

  This manual provides the application programmer or programmer with a wide range discussion of the AS/400 programming topics.

- *Programming: Control Language Reference*, SC41-0030
  Short Title: *CL Reference*

  This manual provides the application programmer with a description of the AS/400 control language (CL) and its commands. Each command includes a syntax diagram, parameters, default values, keywords, and an example.

- *Cryptographic Support/400 User's Guide*, SC41-8080
  Short Title: *Cryptographic Support/400 User's Guide*

  This manual provides the system operator or programmer with a description of the data security capabilities of the AS/400 Cryptographic Support. Cryptographic support is not a part of the operating system. You can order the cryptographic licensed program from the IBM Software Division.

- *Distributed Data Management Guide*, SC41-9600
  Short Title: *DDM Guide*

  This manual provides the application programmer or programmer with information about remote file processing. It describes how to define a remote file to OS/400 DDM (distributed data management), how to create a DDM file, what file utilities are supported through DDM, and the requirements of OS/400 DDM as related to other systems.

- *Systems Application Architecture* Structured Query Language/400 Reference*, SC41-9608
  Short Title: *SQL/400* Reference*

  This manual provides the application programmer, programmer, or database administrator with information that describes SQL/400 statements and their parameters.

- *Data Management Guide*, SC41-9658
  Short Title: *Data Management Guide*

  This manual provides the application programmer with information about using files in application programs. Files allow data that is external to an application program to be read from or written to devices attached to the system, such as database files, device files, and files used to communicate with the system.

- *Programming: Performance Tools/400 Guide*, SC41-8084
  Short Title: *Performance Tools/400 Guide*

  This manual provides the programmer with information about what AS/400 Performance Tools are, gives an overview of the tools, and tells how the tools can be used to help manage system performance.

- *Security Concepts and Planning*, SC41-8083
  Short Title: *Security Concepts and Planning*

  This manual provides the programmer (or someone who is assigned the responsibilities of a security officer) with information about system security concepts, planning for security, and setting up security on the system. This guide does not describe security for specific licensed programs, languages, and utilities.

- *System Operator's Quick Reference*, SX41-9573
  Short Title: *Operator's Quick Reference*

  This manual provides the system operator with quick reference information when working with the AS/400 system. This guide contains summaries of information such as system values and OS/400 DDS keywords.

- *Programming: Work Management Guide*, SC41-8078
  Short Title: *Work Management Guide*

  This manual provides the programmer with information about how to create a work management environment and how to change it.

- *System Operator's Guide*, SC41-8082
  Short Title: *Operator's Guide*

  This manual provides the system operator or system administrator with information about how to use the system unit operator display, send and receive messages, respond to error messages, start and stop the system, use control devices, work with program temporary fixes (PTFs) and process, and manage jobs on the system.

- *System/36 to AS/400 Migration Aid User's Guide and Reference*, SC09-1166
  Short Title: *System/36 to AS/400 Migration Aid User's Guide and Reference*

  This manual provides the system operator, applications programmer, systems programmer and data processing manager with information about using the S/36* to AS/400* migration aid to move S/36 items to the AS/400 System using menus and displays or commands.

- *System/38 to AS/400 Migration Aid User's Guide and Reference*, SC09-1165
  Short Title: *System/38 to AS/400 Migration Aid User's Guide and Reference*

This manual provides the system operator, application programmer, programmer, or data processing manager with information about using the System/38 to AS/400 Migration Aid to move System/38 objects to the AS/400 system using menus and displays, or commands.

- *9406 System Installation and Upgrade Guide*, SY44-0700

Short Title: *9406 System Installation and Upgrade Guide*

This manual provides the service representative with information about upgrading equipment on the 9406 System Unit. It provides information on an entire range of upgrades such as simple memory card additions, device and rack additions, and model upgrades. It is used with the instruction packets that are shipped with the upgrade equipment.

Index

D

L

M

P

Q

R

T

U

V

W

Numerics

Readers' Comments

Application System/400™
Backup and Recovery Guide
Version 2

Publication No. SC41-8079-00

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name

Address

Company or Organization

Phone No.

Readers' Comments
SC41-8079-00

Cu
Al

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

ATTN DEPT 245
IBM CORPORATION
3605 HWY 52 N
ROCHESTER MN 55901-7899

Fold and Tape

Please do not staple

Fold and Tape

Cut
Alon

Readers' Comments

Application System/400™
Backup and Recovery Guide
Version 2

Publication No. SC41-8079-00

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name	Address
Company or Organization	
Phone No.	

Readers' Comments
SC41-8079-00

IBM®

Cut
Alo

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

ATTN DEPT 245
IBM CORPORATION
3605 HWY 52 N
ROCHESTER MN 55901-7899

Fold and Tape | **Please do not staple** | Fold and Tape

Readers' Comments

Application System/400™
Backup and Recovery Guide
Version 2

Publication No. SC41-8079-00

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name

Address

Company or Organization

Phone No.

Readers' Comments
SC41-8079-00

Cu
Al

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

ATTN DEPT 245
IBM CORPORATION
3605 HWY 52 N
ROCHESTER MN 55901-7899

Fold and Tape | **Please do not staple** | Fold and Tape

SC41-8079-00

Cu
Al

Readers' Comments

Application System/400™
Backup and Recovery Guide
Version 2

Publication No. SC41-8079-00

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name

Address

Company or Organization

Phone No.

Readers' Comments
SC41-8079-00

IBM®

Cut
Alor

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

ATTN DEPT 245
IBM CORPORATION
3605 HWY 52 N
ROCHESTER MN 55901-7899

Fold and Tape | **Please do not staple** | Fold and Tape

Cι
Al

Readers' Comments

Application System/400™
Backup and Recovery Guide
Version 2

Publication No. SC41-8079-00

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name	Address
Company or Organization	
Phone No.	

Readers' Comments
SC41-8079-00

C
A

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

ATTN DEPT 245
IBM CORPORATION
3605 HWY 52 N
ROCHESTER MN 55901-7899

Fold and Tape | **Please do not staple** | Fold and Tape

SC41-8079-00

C
A

IBM®

Program Number: 5738-SS1

Printed in Denmark by Herrmann & Fischer A/S

SC41-8079-00