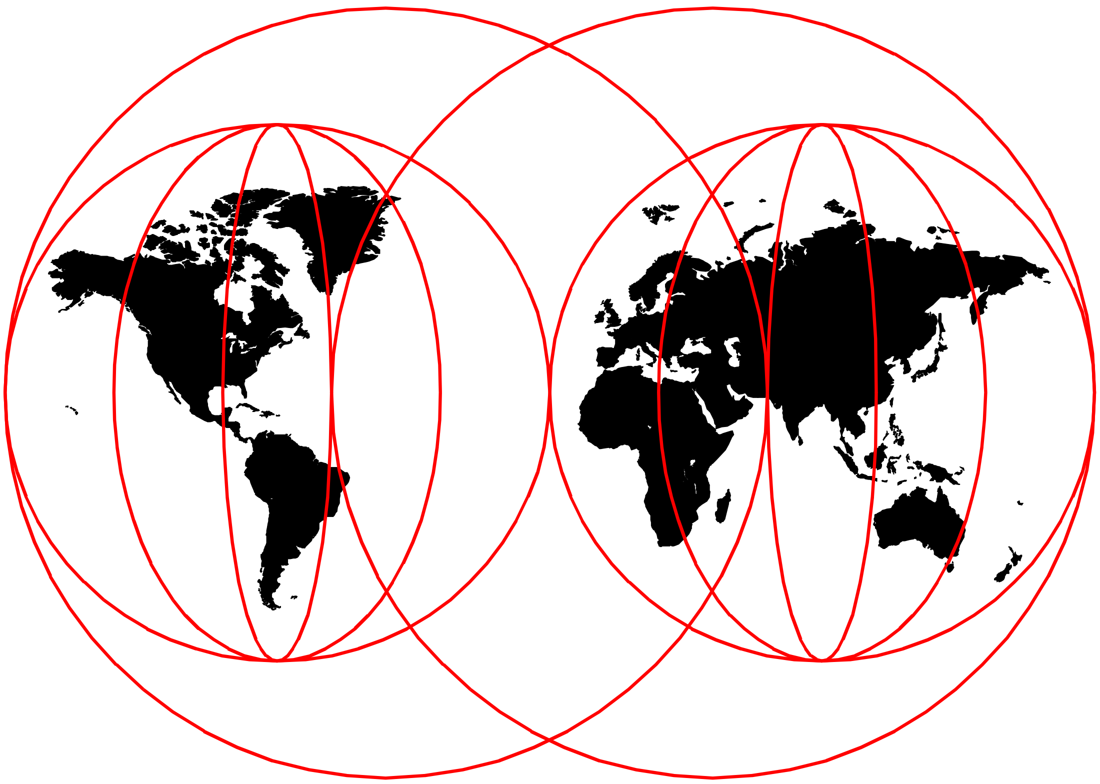
IBM

AS/400 Disk Storage Topics and Tools

Susan Powers, Bill Bradshaw, Pat Lozano, Olga T. Saldivar, Satid Singkorapoom



International Technical Support Organization

www.redbooks.ibm.com

SG24-5693-00

International Technical Support Organization

SG24-5693-00

AS/400 Disk Storage Topics and Tools

April 2000

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix F, “Special notices” on page 219.

First Edition (April 2000)

This edition applies to Version 4 Release 4 of OS/400, AS/400 Operating System, Program Number 5769-SS1.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. JLU Building 107-2
3605 Highway 52N
Rochester, Minnesota 55901-7829

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.

© Copyright International Business Machines Corporation 2000.   All rights reserved.
Note to U.S Government Users - Documentation related to restricted rights - Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

© Copyright IBM Corp. 2000

Figures

© Copyright IBM Corp. 2000

Tables

© Copyright IBM Corp. 2000

Preface

Uncover the basics of AS/400 disk storage. This redbook is designed to help AS/400 system administrators and operators gain a broad understanding of AS/400 disk storage, architecture, and management. The technology covered in this redbook is based on current AS/400 system offerings, along with pertinent history, in contrast to newer technology and methods.

This book is divided into two parts. Part 1 discusses hardware architecture and components, while Part 2 discusses tools to manage disk storage. You'll gain a base understanding of AS/400 system architecture and learn about the architecture of disk storage, disk storage components, and disk storage options. You'll also learn about the storage features that are available to maximize the availability and performance of the disk subsystem.

Plus, this redbook offers information on tools and tips to help the AS/400 system administrator and operator to better manage the AS/400 disk storage subsystem space utilization, both historically and in real-time. Management tips are based primarily on the software components offered with OS/400 V4R4.

Throughout this redbook, the term direct address storage drive (DASD) is used. For the AS/400 system, DASD is another term for disk storage. We use these two terms interchangeably.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Rochester Center.

Susan Powers is a Senior I/T Specialist at the International Technical Support Organization, Rochester Center. Prior to joining the ITSO in 1997, she was an AS/400 Technical Advocate in the IBM Support Center with a variety of communications, performance, and work management assignments. Her IBM career began as a Program Support Representative and Systems Engineer in Des Moines, Iowa. She holds a degree in mathematics, with an emphasis in education, from St. Mary's College of Notre Dame. She served as the project leader for this redbook.

Bill Bradshaw is a Senior Systems Engineer for Change Management Pty Ltd in Adelaide, South Australia. He has 20 years of experience in the computing industry as both a customer and as a supplier. Prior to joining Change Management in 1992, he was an Advisory Systems Strategist with IBM Australia. He holds a degree in Applied Mathematics from the University of Adelaide. His areas of expertise include AS/400 Solution Strategy and Technical Support. Bill wrote the chapters on disk management tools, including the sample code presented.

Pat Lozano is an AS/400 Hardware and Software Services Instructor and Developer for IBM Education in Atlanta, Georgia. She is the lead developer and instructor of numerous AS/400 courses on internals, diagnostic, installation, and hardware and software upgrades. Pat has twenty-three years experience in the technology industry serving as an IBM Customer Service Representative and AS/400 Services Developer. She co-developed a process for AS/400 System

© Copyright IBM Corp. 2000

Migrations and System Transitions. Pat holds a Certified Electronic Technician degree from the Association of Certified Electronic Technicians through Charles Stuart Mott University and is a Certified Technical Trainer recognized by the International Board of Standards for Training Performance and Instruction.

Olga T. Saldivar is an AS/400 Hardware and Software Services Instructor and Developer for IBM Global Learning Services in Atlanta, Georgia. She has 20 years of experience in the technology field, ten years as an IBM Customer Service Representative, and the remaining ten years in education. She has developed and delivered courses on DASD Management and AS/400 hardware maintenance and services primarily to internal IBM service personnel. She has presented numerous topics at the IBM AS/400 Technical Conferences. Olga holds a degree in computer repair from the Electronic Technical Institute in San Diego, California, and is a Certified Technical Trainer recognized by the International Board of Standards for Training Performance and Instruction.

Satid Singkorapoom is an Advisory Product Specialist in IBM Thailand. He has nine years experience with the AS/400 system. He holds a masters degree in Computer Engineering from the Asian Institute of Technology, Thailand. His areas of expertise include AS/400 Client/Server and AS/400 Technical Sales. He has been extensively involved with customers on AS/400 product technology updates and AS/400 technical sales and marketing activities. He also co-authored the 1999 redbook *AS/400 Client Access Express for Windows: Implementing V4R4M0*, SG24-5191.

Thanks to the following people for their invaluable contributions to this project:

Jim Cook
International Technical Support Organization, Rochester Center

Brian Podrow
AS/400 Brand Marketing

Sue Baker
AS/400 Product Support

Brian Bakke
Steve Beilby
Larry Connoy
Tom Crowley
Julie Duncan
Carl Forhan
Bob Galbraith
Jessica Gisi
Dave Legler
Bruce Long
Mark Manges
Glen Nelson
Steve Norgaard
Dan O’Hare
Jeff Parker
Luz Rink
Laurel Scaife
Kevin Swancutt

Rick Weckwerth
Joe Writz
IBM Rochester Laboratory

Comments welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “IBM Redbooks review” on page 233 to the fax number shown on the form.
- Use the online evaluation form found at http://www.redbooks.ibm.com/
- Send your comments in an internet note to redbook@us.ibm.com

Part 1. Storage architectures and components of AS/400 DASD

Part 1 contains topics, which are mostly hardware-oriented in nature. The architecture of the system, storage architecture, and disk in particular are represented, including a description of AS/400 hardware components. More detailed information on architecture can be found in Appendix A, “Architecture” on page 171. Storage management terminology is found in Appendix B, “Storage management terms used in SLIC” on page 193.

© Copyright IBM Corp. 2000

Chapter 1. System architecture

As AS/400 disk functions depend on an interaction with many components of the system, an understanding of the AS/400 system architecture is useful to build a base understanding of disk architecture. This chapter provides basic information to help you build a base knowledge for understanding and working with the content in the subsequent chapters.

1.1 Components of the AS/400 system

The purpose of this first section is to provide a base understanding of the AS/400 system architecture for administrators and system operators. Specifically, we describe the processes the AS/400 system performs during an initial program load (IPL). These processes are described beginning with the system unit in a powered down condition.

Refer to Figure 1 for a visual representation of the AS/400 Complex Instruction Set Computing (CISC)-based system architecture. Section 1.2, “Licensed Internal Code (LIC)” on page 5, contains a comparison of the CISC and Reduced Instruction Set Computing (RISC) architectural structure, together with detailed descriptions of both.

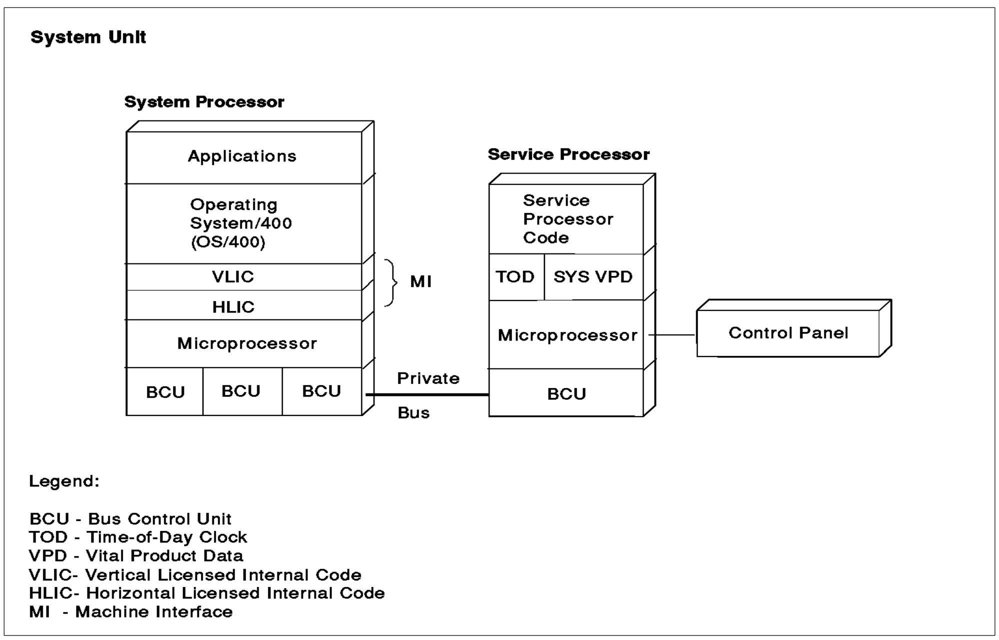


Figure 1. CISC system architecture overview

The items outlined in this section include:

- Service Processor functions
- Basic Assurance Tests (BATs)
- Service Processor and control panel interface
- Walking the bus during an IPL
- Events during an IPL

Service processor functions

The AS/400 system uses the service processor to get the system processor (also known as the central processor unit or CPU) to an operational state. The CPU also uses the service processor to provide error reporting and debug operations for the system CPU.

Some of the basic functions of the service processor are:

- Starts the system IPL
- Tests bus 0
- Tests the CPU
- Loads the CPU LIC
- Provides a system termination path (SRC display)
- Supports read and write time of day
- Supports read and write system Vital Product Data (VPD)
- Supports Programmed restart IPLs (that is, a PWRDWNSYS *IMMED RESTART(*YES) command)
- Controls delayed power off
- Provides a general control panel interface

The following hardware is specifically service processor hardware (as compared to system processor hardware):

- The time-of-day chip
- The system Vital Product Data (VPD) storage
- The Control panel interface

Basic Assurance Tests (BATs)

During the IPL process, the service processor verifies that the I/O bus, the load source I/O bus unit, the system processor, and the service processor itself are operational. It also supplies specific fault information to the user in the event of a failure during this verification process. These are known as Basic Assurance Tests (BATs).

Service processor and control panel interface

The control panel connects to the service processor and sets up an interface for the user from which the IPL source can be selected and for indicating status and error conditions. The control panel Licensed Internal Code gives the service processor and operating system access to control panel functions.

Walking the bus during IPL

On early 9406 systems (that is, the stage 1 model B), the service processor must search multiple bus units on I/O bus 0. Bus 0 is the first I/O bus on these models.

This search consists of:

- A sequence of bus commands that first identifies the bus configuration, including location and state of each bus unit, and then checks each bus unit to find the service processor load.
- An expected response from the bus unit identifying itself as the load source I/O bus unit.

- Having located the load source I/O bus unit, the read-only storage (ROS) Licensed Internal Code loads the service processor random access memory (RAM) control storage with the service processor run-time code and turns control over to that Licensed Internal Code.

On the next series of models manufactured after the model B, the service processor was designed to be part of a multiple function I/O bus unit. On these models, the service processor is combined with disk, diskette, tape, communications, and on some models CD-ROM support. Searching for the load source I/O bus unit is quicker during IPL because bus 0 does not need to be searched to get the service processor code load. The IPL is performed from disk or tape devices attached directly to the system.

Events during IPL

The sequence of events that occur during an IPL from a powered-down condition are outlined here:

1. When the system is first powered on, the service processor:
    - Assumes control of bus 0 and obtains its Licensed Internal Code load-source from disk or tape.
    - Retrieves the Licensed Internal Code for the system processor.
2. The service processor sets up the interface between the operating system and control panel functions.
3. Control storage is loaded with a sequence of BATs to check out the system processor.
4. The system processor Vertical Licensed Internal Code (VLIC) is loaded into main storage, followed by the Horizontal Licensed Internal Code (HLIC), which is loaded into system processor control storage.
5. The system processor is started and, when the horizontal code is initialized, the service processor code instructs the system processor to start reading control words out of control storage.

    Once the Vertical Licensed Internal Code startup is complete, control of I/O bus 0 is transferred from the service processor to the system processor.
6. The system processor Vertical Licensed Internal Code continues loading code from the remaining I/O devices attached to the system.
7. Once the system is running, the user sign-on screen is displayed on the system console or attached and available workstations.

For more detailed descriptions of other components of the AS/400 system, such as the various IOPs and IOAs related to storage, turn to Appendix A, “Architecture” on page 171.

1.2 Licensed Internal Code (LIC)

The AS/400 system product is made up of hardware and Licensed Internal Code (LIC). LIC supports the user interface (OS/400 instruction set, menus, help, list displays, a command set, and so forth). LIC is the group of instructions that contain basic machine control functions.

The first portion of LIC to be described is the Technology Independent Machine Interface (TIMI). Then, a detailed description of LIC is presented for both RISC and CISC systems.

1.2.1 Technology Independent Machine Interface

The AS/400 system is atypical in that it is defined by software, not by hardware. In other words, when a program presents instructions to the machine interface for execution, it “thinks” that the interface is the AS/400 hardware. But it is not! The instructions presented to that interface pass through a layer of microcode before they can be understood by the hardware itself. Through a comprehensive layer of microcode, TIMI design insulates application programs and users when hardware characteristics change. When a different hardware technology is deployed, IBM rewrites sections of the microcode to absorb fluctuations in hardware characteristics. As a result, the interface presented to the customer remains the same.

This interface is known as the Technology Independent Machine Interface (TIMI). The microcode layer is known as the System Licensed Internal Code (SLIC).

The brilliance of this design was dramatically illustrated when the AS/400 system changed its processor technology from CISC processors to 64-bit RISC processors in 1995. With any other system, the move from CISC to RISC would involve recompiling (and possibly some rewriting) of programs. Even then, the programs would run in 32-bit mode on the newer 64-bit hardware.

This was not so with the AS/400 system because of TIMI. Customers could save programs off their CISC AS/400 systems, restore them on their newer RISC AS/400 systems, and the programs would run. Not only did they run, but they became full function 64-bit programs.

As soon as they made the transition to RISC, customers had 64-bit application programs that ran on a 64-bit operating system containing a 64-bit relational database that fully exploited the 64-bit RISC hardware. These same architectural features will be exploited to fully accommodate post-RISC technologies, which may have 96-bit or 128-bit processors.

RISC-based SLIC on PowerPC systems (V3R6, V3R7, V4Rn)

Many of the frequently-executed routines that, on a non-AS/400 system typically reside in the operating system, have been moved to SLIC. As the various version and release levels of OS/400 are released, more and more functions were moved into SLIC. Because SLIC is closer to the silicon, routines placed in SLIC run faster than routines placed “higher” in the machine. This provides an important performance gain, examples of which include some of the basic supervisory functions. Examples of resource management functions that are in SLIC include validity and authorization checks.

Figure 2 shows how the structures compare between the CISC and RISC architectures.

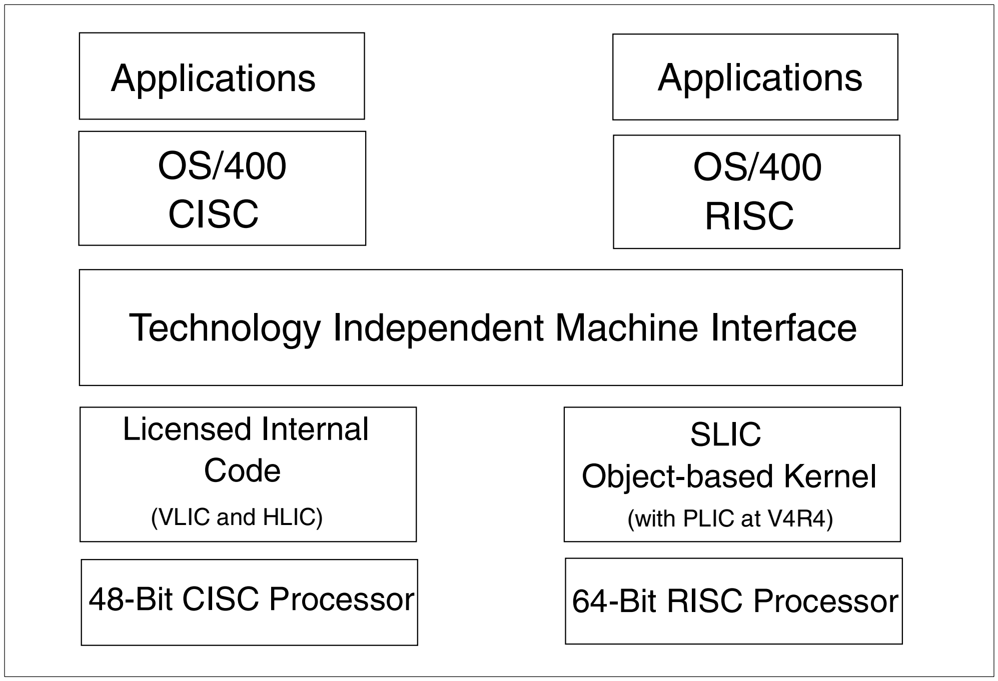


Figure 2. Comparison of CISC and RISC architectures

1.2.2 PLIC on AS/400e series systems (V4R4)

Partitioning LIC, or PLIC, is the code that communicates with each partition and controls the hardware. PLIC is used to implement Logical Partition (LPAR) functions and to control bus and storage management functions for each partition. LPAR is discussed in 2.3, “Logical partitioning on the AS/400 system” on page 22.

PLIC is automatically installed with SLIC, even if you choose not to implement LPAR functions on your system. Much like the fact that every AS/400 system is made up of a system Auxiliary Storage Pool (ASP), and you have the option to create user ASPs, every system has a primary partition and you have the option, (given the proper hardware) to define secondary partitions.

PLIC is a low level layer of code that runs logical partitions on V4R4 systems. This global code runs in all processors on all partitions.

Examples of PLIC functions include:

- Build and monitor the hardware page table.
- Build and monitor the Translate Control Entry (TCE) table for I/O. The TCE translates a 32-bit address to a 64-bit address.
- Interpartition communication support.
- CPU controls and support for each partition.

If PLIC code fails, the entire system fails.

Hypervisor

Hypervisor is the program that controls and monitors activity between the primary and secondary partitions. It manages the allocation of system resources, ensures that a partition uses the resources that it owns, and provides the service processor and control panel functions for the secondary partitions.

Figure 3 shows where PLIC and PLIC code fit into the internal structure on an LPAR system.

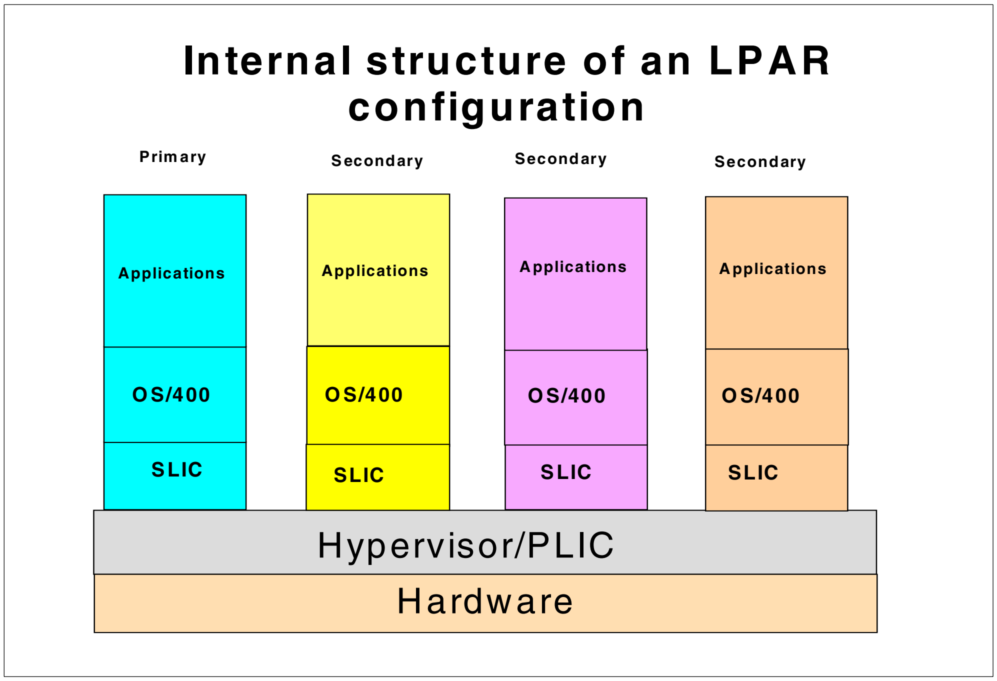


Figure 3. Partition Licensed Internal Code (PLIC) and the Hypervisor

For additional online articles about LPAR, visit this Web site:
`http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/INFOCENT.HTM`

Or refer to *Slicing the AS/400 with Logical Partitioning: A How to Guide*, SG24-5439.

IMPI-based LIC on CISC systems (all systems up to and including V3R2)

The AS/400 CISC architecture is similar to the Internal Microprogram Instruction (IMPI) architecture developed for the System/38. IMPI is divided into five distinct layers, each layer giving support to the layer above and the layer below.

Refer to Figure 4 for a visual representation of LIC on a CISC system.

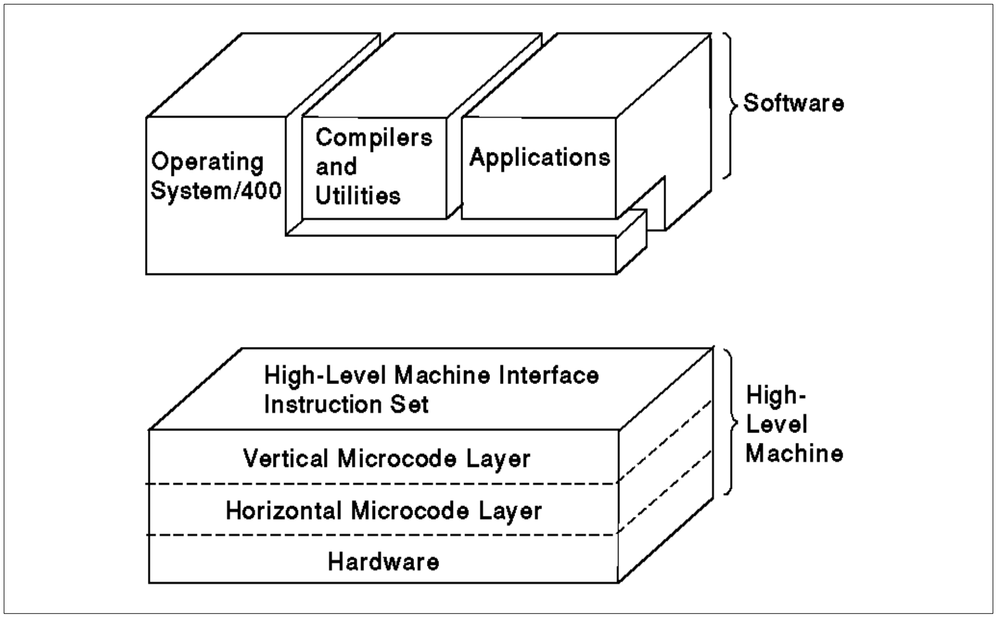


Figure 4. CISC (LIC) architecture strategy

The architecture is designed so that a change to one level does not affect a function at a different layer. And knowing how the layers work with each other gives a better understanding of the system when diagnosing problems.

1.2.3 Vertical Licensed Internal Code (VLIC)

VLIC is code that defines logical operations on data that is sequential in operation. It is the layer with support for the OS/400 instruction set.

A programmer writes fewer instructions to accomplish the same function on the AS/400 system compared with other systems. The fewer the instructions, the less the potential for human error and the greater the reliability of the program.

VLIC is separated into two distinct classes of support. One class is the operating system, including such functions as storage management, database management, and I/O support. The second class is the translator, which converts machine instructions into program language.

1.2.4 Horizontal Licensed Internal Code (HLIC)

IMPI and CISC instructions are interpreted by the next lower level of micro-programs, named horizontal microcode (HLIC). The processing of these instructions is done by HLIC routines, consisting of one or more HLIC instructions named control words. The hardware directly decodes and processes HLIC control words.

The processor hardware does not process IMPI instructions directly. IMPI instructions are converted into a series of sequentially processed HLIC control words. The control words are directly decoded and run by the service processor.

1.3 AS/400 addressing

The system needs a method to identify a single destination for any information transfer. Every hardware component (bus, I/O processor, controller, and storage unit) has a unique address. The addressing convention is used to allow a specified destination to read the data specified for the destination.

There are several addressing conventions used within AS/400 disk subsystems. Physical locations are used, as well as logical addresses, small computer system interface (SCSI) addressing, device function controller interface (DFCI) addressing, and direct select addressing (DSA).

DFCI and SCSI are further discussed in A.2.1, “Device Function Controller Interface (DFCI) bus” on page 174, A.2.2, “Small Computer System Interface (SCSI) bus” on page 174, and A.5, “DASD I/O cables within the AS/400 system” on page 184. More information on DSA is available in 1.3.3, “Direct select address (DSA) and unit address” on page 11.

IOPs are addressed in two modes:

Direct select addressing (DSA)
The method for selecting a physical card slot during IPL or servicing when only a limited set of functions is allowed.

Logical bus unit addressing
The method for selecting an IOP during normal operation, after the IOP and its subsystem is identified and configured for the system. The logical address can differ from the direct select address, as discussed in the next topics.

Addressing in the AS/400 system has evolved dramatically from the CISC platform to the RISC platform. Discussion of addressing begins with a description of how the system addresses a component in a CISC system. As the topic progresses, you can learn the addressing of components in the current technology AS/400 systems.

1.3.1 Physical location

When you look at the system, you see numbers or labels on the components marked on the card slot or position. These numbers indicate physical locations. For example, refer to Figure 5 to observe physical locations on the 9406 Stage 2 model 35 and 45. The physical locations in the system unit card cage are labeled as slots 1 through 13.

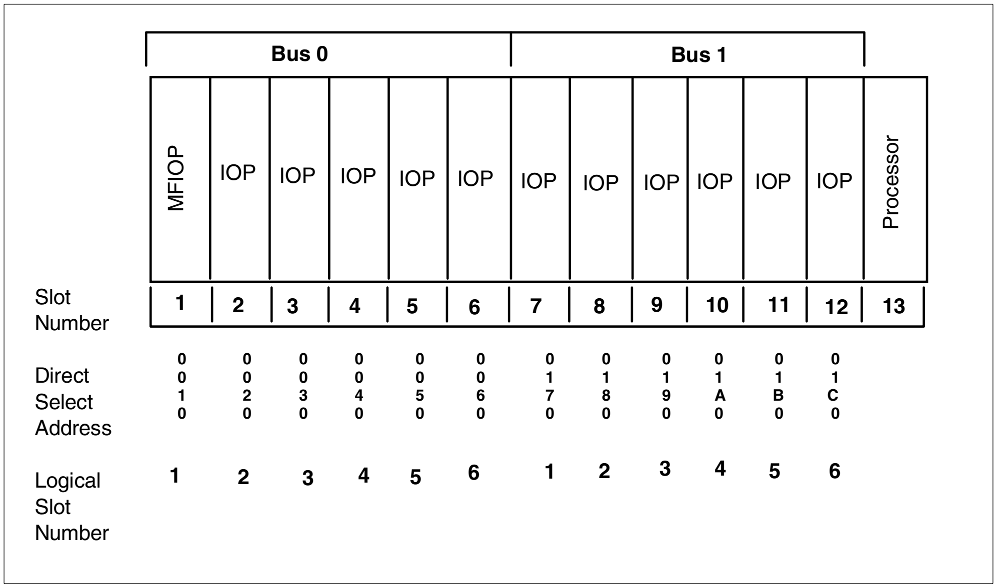


Figure 5. Model D, E, F 3xx and 4xx, physical slots, DSA, and logical address

1.3.2 Logical addressing

Logical addressing is used by the system to locate a device. The logical address may or may not be the same number as the physical location identifier. For example, refer to Figure 5 to observe the 9406 Stage 2 model 35 and 45. The logical addressing in the system unit card cage shows the first component on bus 0 followed by the second component, and so forth, through the sixth component on bus 0. The logical addressing then starts over at physical slot 7 showing the first component on bus 1, the second component on bus 1, and so forth.

Physical and logical addressing schemes are found in the Configuration Rules appendix of the *940x 170, 6xx, 7xx, and Sxx System Installation and Upgrade Guide*, SY44-5950, as well as the *AS/400 System Builder,* SG24-2155.

1.3.3 Direct select address (DSA) and unit address

The direct select address (DSA) card address has four digits. These are immediately followed by eight digits associated with the unit address of each attached device. The DSA chart in Figure 6 on page 12 and the one in Figure 5 illustrate the address format for the frame they are depicting. The first two digits of the DSA is the system I/O bus number.

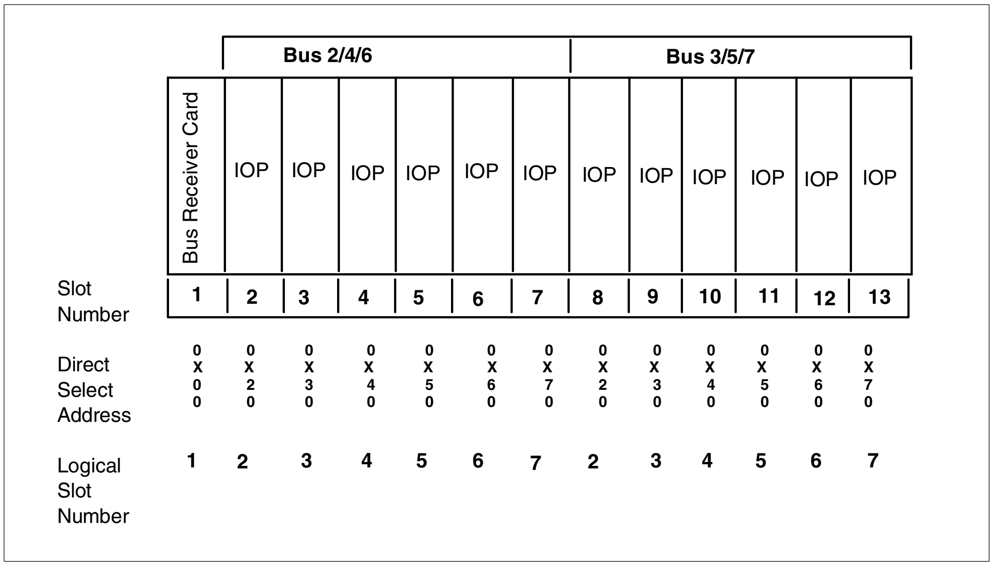


Figure 6. #5042/#5044 physical slots, DSA, and logical address

The first digit in the unit address displays a port number. This is the port on the IOP card. This port number is also referred to as the bus number. It is important to understand that this port (bus) number is not the system I/O bus, rather it is the SCSI bus related to the IOP card.

On external devices, the device controller number is often set with a switch. For internal devices, it is set by the system. For internal devices, the controller is often physically part of the logic card attached to internal Head Disk Assemblies (HDAs).

It should be understood that, although DSA addressing is a logical address, it does not always follow what we may feel is a logical flow.

Direct select addressing is how the system talks to us about individual components. The system references DSA addresses in operator's messages, problem logs, and error logs on the CISC platform. On the RISC platform, the problem logs are split into two different logs: a product activity log (PAL) for customer review, and the service action log (SAL), used by service representatives for a repair action.

For an example of the report generated when displaying the PAL, see Figure 7.

```
                          Log Analysis Report


   From...:10/06/99    14:05:18            To...: 10/08/99   14:05:18

   Type options, press Enter
   5=Display report                6=Print report


   Opt      System       Date           Time             Logical address
            Ref Code
    5       67173002     10/07/99       16:27:13          1/1/0/1-2/0/1/1/0/
            B6005120     10/07/99       16:30:08          1/1/0/1-2/0/1/1/0/
            27489051     10/07/99       16:45:14          1/1/0/1-/////



F3=Exit                                                       More.....
F11=View Physical Location          F12+Cancel
```

Figure 7. Display of product activity log (PAL)

This is the type of information you provide to your service representative. They use it for failure analysis and repair. They also use the SAL report for detailed information.

For an example of the SAL, see Figure 8.

```
                         Service Action Log



From...: 10/06/99     14:22:48      To...: 10/08/99    14:22:48

Select one valid option at a time, Press Enter


2=Display failing item information         8=Close a NEW entry
9=Delete a CLOSED entry

Opt    Status    Date       Time        SRC       Resource       Isolated?   Count
 2     NEW       10/07/99   16:27:07    67173002                      Y          1
       NEW       10/07/99   16:45:14    27489051  CMB01               N          4



F3=Exit         F12=Cancel
```

Figure 8. Service Action Log (SAL) report

Notice there is an option to obtain failing item information (option 2). After selecting the display failing item information option from the service action log report, the detail includes the DSA. Figure 9 on page 14 provides an example of this.

```
                 Display Failing Item Information


Failing Item      Description                               Location

1 DISKDRV         DISK DRIVE AND LOGIC CARD
2 STORIOA         I/O ADAPTER CARD                          0110 010100FF
3 DISKTRY         DISK UNIT TRAY                            0110 FFFFFFFF
4 F101106         REMOVABLE MEDIA OR DISK UNIT NOT ACTIVE
5 F101140         CABLE ASSEMBLY



                       PROBLEM SRC: 67173002

F12=Cancel
```

Figure 9. Failing item information

Notice how it provides a failing item listing. The items are presented in order of probable cause of failure.

It is imperative to realize that DSA does not always follow a logical flow. For example, refer to the DSA plug charts for the Stage 2 model 35 and 45s, as shown in Figure 5 on page 11. Instead of referencing the logical components on a bus (like logical addressing does), DSA counts from 1 through C on both buses. It is tempting to, therefore, assume it counts logically through the card cage instead of the bus, *but this is a wrong assumption*. For an example, refer to the DSA plug chart for a #5042 and #5044 card cage as shown in Figure 6 on page 12. The DSA starts counting with 0, skips 1, and continues with 2.

It is also worth noting that the card in physical slot 1 actually has two DSA addresses. For example, in Figure 6, if the #5042 is the first expansion in the system, it adds buses 2 and 3. Then the card in physical slot 1 has a DSA address of both 0200 and 0300. If a component on the card that only affects bus 2 fails, the system tells you to replace the card at 0200. If a component on the card, that only affects bus 3 fails, the system tells you to replace the card at 0300. If the entire card fails, the system tells you to replace two cards, both 0200 and 0300, even though this is in fact the same card.

An even more dramatic example is on the Model 500 system unit. Refer to Figure 10 to observe that the logical and DSA addressing match. However, they are nothing like the physical slot order.

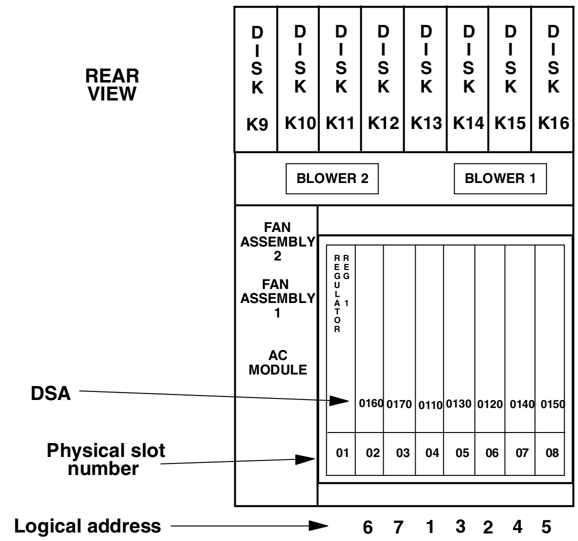


Figure 10. Model 500 physical slots, DSA, and logical addressing

Admittedly, all of this can become very confusing. The point we are making is: do not trust that you know DSA addressing. This is especially important in the Advanced Series and e-Series AS/400 systems.

Note

Using the wrong DSA can destroy critical data. Be sure to look up the address every time.

In Figure 11 on page 16, observe the physical placement of the PCI board which is the MFIOP board for the AS/400e series.

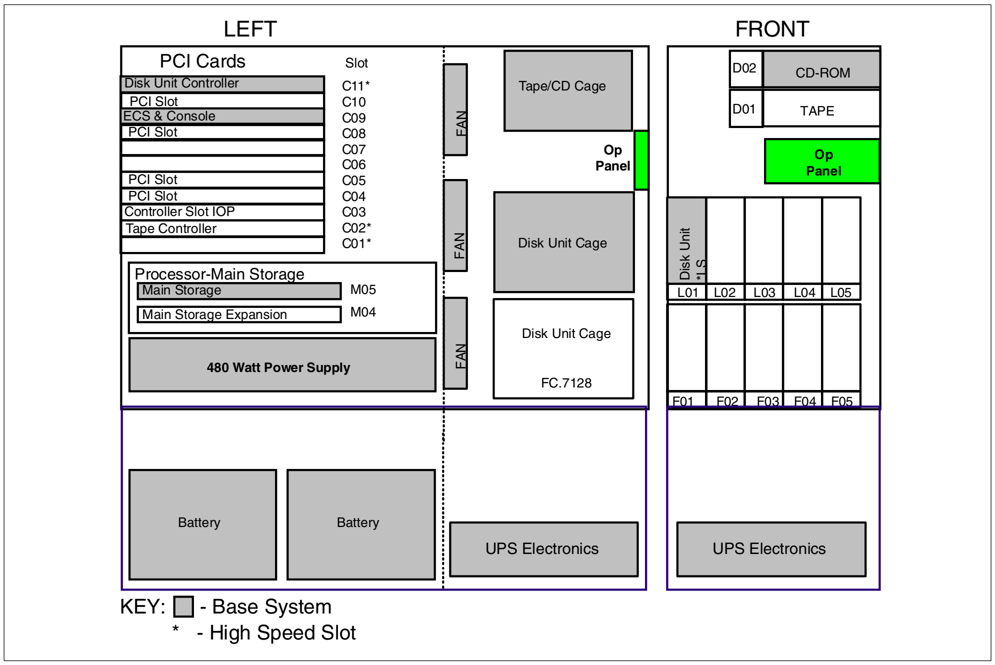


Figure 11. 9406 Model 720 physical slots

This example is of a Model 720. Due to the PCI technology, multiple IOAs can be installed in the MFIOP board. As there are multiple PCI cards installed, each slot needs to have its own unique address for the system to access the I/O attached to the individual IOA.

In Figure 12, observe that the DSA addresses vary by physical slot position.

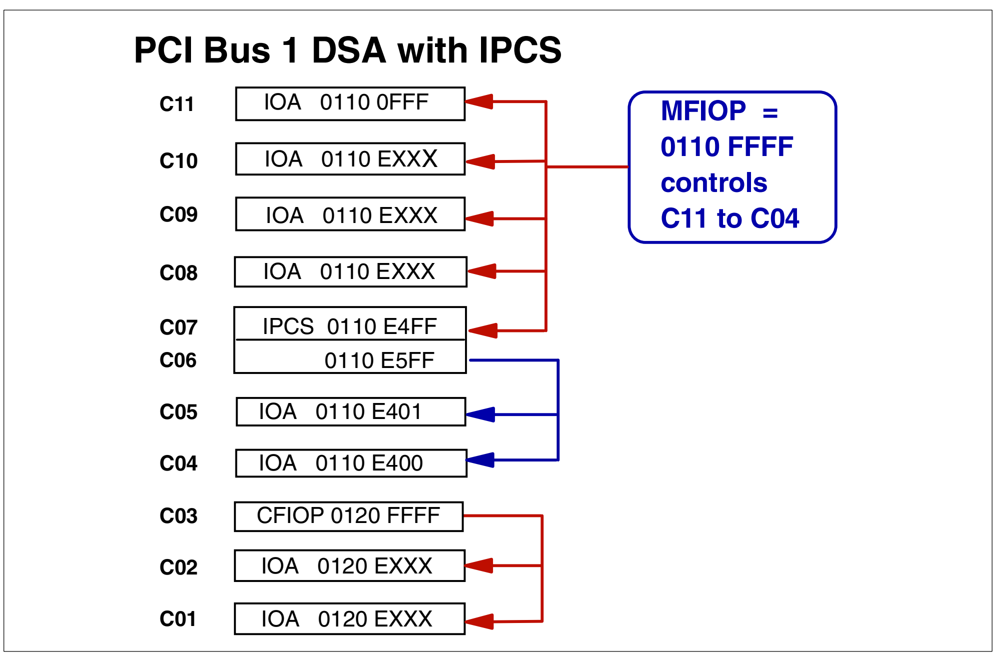


Figure 12. Model 720 DSA logical addresses

The first four digits represent the controller with which the IOA is associated. The board itself provides the first controller, and slots C04 through C11 are controlled by it. The second controller is installed in slot C03. This is an optional feature. When the second controller is installed, slots C01and C02 are controlled by it. If the second controller in slot C03 remains empty, the last three slots on this board are inactive.

The last four digits of the address on AS/400e series models vary, depending on the type of card (for example whether it is a communication or workstation controller card). This addressing scheme is a unique characteristic to this family of processors.

To properly manage and maintain the AS/400 system, it is important to understand the addressing characteristics for each component of the AS/400 Advanced Series and e-series systems. Refer to your IBM Service Representative and the problem analysis and service information guide associated with your system for more information. Schematics are also found in *AS/400 System Builder,* SG24-2155.

Note: The problem analysis guides vary by the associated system type. The correct edition must be used to properly locate a component in an AS/400 system.

For a more detailed discussion of the theory and the architecture of AS/400 buses, IOPs, IOAs, and the various controller functions, refer to Appendix A, “Architecture” on page 171.

1.4 Summary

This chapter introduced you to the concepts of the AS/400 system architecture, beginning with identifying basic system unit components and walking through the flow of an IPL sequence. It provided an overview of differences in the manner in which the AS/400 system addresses and communicates with its device components in the CISC platform versus the RISC platform. There was also an overview of the various levels of License Internal Code, including an introduction to the layer of code announced with V4R4 that aids in the implementation and management of AS/400 logical partitions (LPAR).

Chapter 2. Storage architecture

The purpose of this chapter is to provide information on the various options available for storage on the AS/400 system. While it primarily identifies and describes terminology for storage areas of a permanent nature, main storage is discussed as an option to provide a potential boost in overall system performance. Disk protection options are defined and discussed with examples in this chapter. Chapter 3, “Storage options” on page 53, provides more detailed information about individual models of disk storage options.

Note: Refer to Appendix B, “Storage management terms used in SLIC” on page 193, to understand the terminology related to disk storage.

2.1 Direct access storage device (DASD)

Direct access storage devices (another term for disk drives) are made up of flat circular plates of metal, plastic, or glass, coated on both sides with magnetic compounds. Input signals, which can be audio, video, or data, are recorded on the surface of a disk. These magnetic patterns or spots form concentric circular tracks which pass under a recording head when the disk is rotated by a drive unit. The disk platters are sealed against dust.

The DASD is used as a high-capacity storage device for AS/400 systems.

There is a wide selection of DASD available for the AS/400 system. Refer to Chapter 3, “Storage options” on page 53, for a detailed discussion.

Note

The architecture of the AS/400 system allows for other types of devices to be direct access storage devices. In this redbook, when the term *direct access* storage is used, it is meant to imply disk drives.

2.2 Cache

Cache is a supplemental memory system that temporarily stores frequently used instructions and data. Cache allows for quicker processing by the central processor of a computer. The cache augments, and is an extension of, a computer's main memory.

Both main memory and cache are internal, random access memory (RAM). Both use semiconductor-based transistor circuits.

2.2.1 Expert cache

Expert cache is a set of algorithms that execute in the main CPU. Expert cache uses designated pools in main storage to cache database operations. The size of the pools is adjusted dynamically or controlled by the system operator as a function of time. This allows the user to maximize the efficiency of the cache and main storage usage, as workloads change over the course of a typical day.

© Copyright IBM Corp. 2000

Caching theory assumes that optimal caching occurs as close to the processor as possible. By caching in main storage, the system not only eliminates accesses to the storage devices, but all associated I/O traffic on the internal system buses as well. In addition, an algorithm running in the main processor has a better view of actual application trends and should do a better job, overall, of assessing what data should be cached.

Expert Cache works by minimizing the effect of synchronous DASD I/Os on a job. The best candidates for performance improvement are those jobs that are most affected by synchronous DASD I/Os. The ten jobs with the largest synchronous disk I/O count per transaction are noted on the Job Statistics section of the Transaction Report. Other System, Component and Transaction reports available with the AS/400 Performance Tools/400 (5769-PT1) licensed program product also identify disk I/O statistics. Refer to *Performance Tools/400,* SC41-4340, for a description of each field and report relative to synchronous I/O.

Expert Cache provides a disk cache tuner option, which allows the AS/400 system to take advantage of available main storage capacity. The tuner dynamically responds to system jobs to cache pages of data in main storage, thus reducing the time to process disk I/O.

Refer to Figure 13 for the position of Expert Cache within a caching strategy.

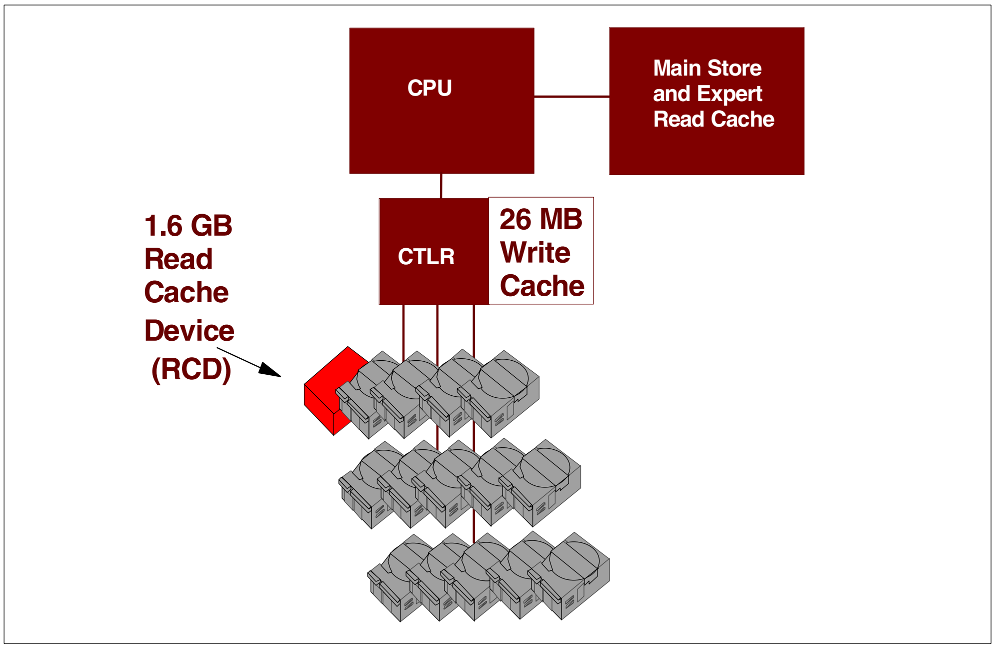


Figure 13. Example of Expert Cache

2.2.2 Extended Adaptive Cache (EAC)

Extended Adaptive Cache is an advanced read cache technology that improves both the I/O subsystem and system response times by reducing the number of physical I/O requests that are read from disk. Extended Adaptive Cache operates at the disk subsystem controller level, and does not affect the AS/400 system processor. Management of the cache is performed automatically within the I/O adapter. It is designed to cache data by using a predictive algorithm. The algorithm considers how recently and how frequently the host has accessed a predetermined range of data.

For additional detail regarding Extended Adaptive Cache, refer to Chapter 4, "Extended Adaptive Cache (EAC)" on page 57.

2.2.2.1 Read Cache Device (RCD)

The Read Cache Device is a solid state 1.6 GB cache memory assembled in a package. From the exterior, it resembles an AS/400 disk unit.

The RCD is optimized for use as cache memory on the AS/400 system. Instead of a spinning platter inside, the RCD is packed with solid state memory chips that function at electronic speeds.

It can be installed in a disk slot of the AS/400e server or expansion storage enclosure using disk concurrent maintenance, if the slot you install it in is vacant. Only one RCD can be installed per feature code #2748. If your system currently has all DASD slots filled, plan for the removal both logically and physically of one disk unit to accommodate the RCD. Once installed, the I/O adapter detects the RCD through automatic polling, and Extended Adaptive Cache goes to work.

The Read Cache Device helps reduce physical read requests to disk units attached to the same #2748 controller as the RCD.

The RCD can be used with RAID-5 protected storage devices or storage subsystems with mirrored protection. The RCD itself does not require RAID-5 or mirrored protection because all data contained on the RCD is also on the disks. RCD storage is volatile. In the event of a RCD failure, read caching is suspended and the system continues to run.

The Read Cache Device (RCD) is the memory for Extended Adaptive Cache. Purchase and install RCD for the Extended Adaptive Cache to function.

Figure 14 shows an example of how Read Cache Device (RCD) fits in the caching function architecture.

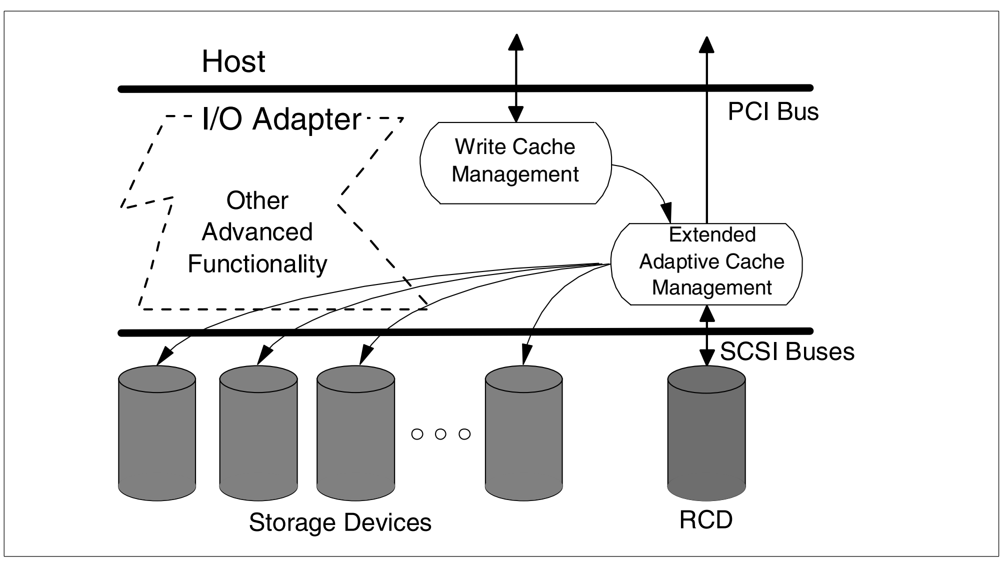


Figure 14. Example of using the Read Cache Device

For additional detail regarding implementation of the Read Cache Device, see Chapter 4, “Extended Adaptive Cache (EAC)” on page 57.

For information regarding physical characteristics and proper physical placement of the RCD, refer to Chapter 3, “Storage options” on page 53.

2.3 Logical partitioning on the AS/400 system

Logical partitions (LPAR) enable multiple independent OS/400 instances or partitions in an n-way symmetric multiprocessing AS/400e 7xx, 6xx and Sxx. Each partition requires its own processor(s), memory, disk(s), and system console, with a CD-ROM and tape drive that can be allocated to each LPAR. With LPAR, you can address multiple system requirements in a single, physical machine to achieve server consolidation, business unit consolidation, mixed production/test environments, integrated clusters, and more.

All V4R4 systems are installed with a primary partition with all resources initially allocated to it. Creating and managing secondary partitions is performed from the primary partition. Movement of processors, memory, and interactive performance between partitions is achieved with an IPL of the affected partitions. Movement of IOP resources between partitions does not require an IPL.

OS/400 is licensed once for the entire system by its usual processor group, regardless of the number of partitions. License management across partitions is not supported. OS/400 V4R4 must be installed on each partition. Previous releases are not supported on any logical partition.

For a visual representation of a Logical Partition configuration, representing partition A and partition B, each with a console and assigned DASD, see Figure 15.

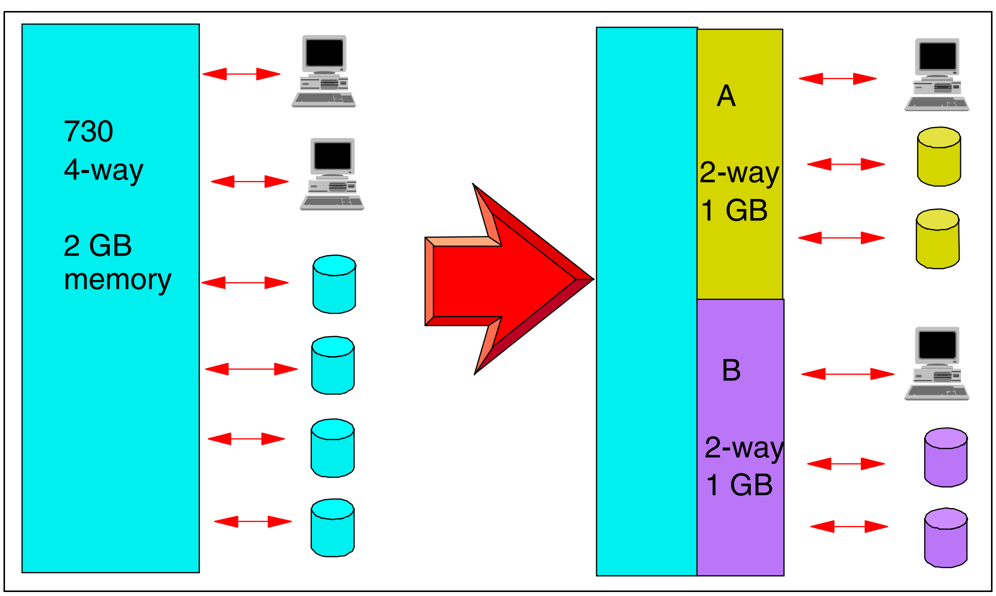


Figure 15. Example LPAR configuration

Rely on this Web site for direction and management tips on Logical Partitioning:
`http://www.as400.ibm.com/lpar/`

For additional online articles about LPAR, visit this Web site:

`http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/INFOCENT.HTM`

2.4 Data storage and management on the AS/400 system

Information is a company asset. Computer systems are based on information, otherwise known as data. Business decisions are made every day based on the content of stored information. Business growth depends upon efficient access to and storage of its information resource.

Data is stored in objects on the AS/400 system, the object providing a template to describe and access the information. The AS/400 database consists of file objects whose associated data is stored permanently in the system. Each file type has unique characteristics that determine how the file is used and what capabilities it provides.

The Integrated File System provides structure for all information stored on the AS/400 system. As a part of the operating system, it supports stream files so that input, output, and storage management of Integrated File System objects is similar to personal computer or UNIX operating systems. The Integrated File System is discussed in 2.5, “Integrated File System” on page 24.

The Database and File Systems category of the Information Center contains a variety of topics about storing data on your AS/400 system. Other such topics include managing data, files, libraries, and databases, and managing and accessing data in the DB2 Universal Database (UDB) for AS/400. To view the Information Center, refer to:

`http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/INFOCENT.HTM`

2.4.1 Data management

Data management is the part of the operating system that controls the storing and accessing of data by an application program. The data may be on internal storage (for example, a database file), on external media (diskette, tape, printer file objects), or on another AS/400 system.

The Data Management topic of the Information Center describes the data management functions that your application uses in creating and accessing data on the AS/400 system and in ensuring the integrity of the data.

2.4.2 Database management

DB2 Universal Database for AS/400 is the integrated relational database manager on your AS/400 system. As well as providing access to and protection for your data, DB2 UDB for AS/400 provides advanced functions such as referential integrity and parallel database processing.

The Database Management topic of the Information Center describes how to take advantage of DB2 UDB for AS/400 to access and manage AS/400 data through an application or a user interface.

For detailed information on these subjects refer to the *AS/400 Information Center,* SK3T-2027 (CD version which comes with your AS/400 system). This package also contains your AS/400 Softcopy Library. Or you can visit the Information

Center Web site at: `http://www.as400.ibm.com/infocenter` or
`http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/INFOCENT.HTM`

2.5 Integrated File System

As previously mentioned, the Integrated File System is a part of OS/400 that lets you support stream input, output, and storage management similar to personal computer and UNIX operating systems, while providing structure for all information stored in the AS/400 system. The key features of the Integrated File System are:

- Support for storing information in stream files that can contain long continuous strings of data
- A hierarchical directory structure
- A common interface that allows users and applications to access not only the stream files, but also database files, documents, and other objects that are stored in the AS/400 system
- A common view of stream files that are stored locally on the AS/400, an Integrated Netfinity Server for AS/400, or a remote Windows NT server.

Two file systems providing flexible storage management are UDFS and QFILESVR. For a description of the User Defined File System (UDFS), refer to Chapter 5, "Extending the Integrated File System with UDFS" on page 73, and Chapter 6, "Using the QFILESVR.400 file system" on page 81, for the OS/400 File Server file system.

For additional reading on using the Integrated File System on the AS/400 system, refer to *Integrated File System Introduction,* SC41-5711.

2.6 Digital Versatile Disk (DVD) on the AS/400 system

DVD is a family of optical discs that have the same overall dimensions as a CD, with significantly higher capacities. DVD is a writable media, where CD-ROM is read-only. DVDs are double-sided, where CDs are single-sided. Dual-layer DVD versions are planned. DVD drives read most CD media, and other pre-recorded DVD media (DVD-Video and DVD-ROM).

DVD technology is becoming as important for the computer world as for the video world. DVD originally stood for Digital VideoDisc. The "video" was dropped, and it became simply DVD. It is now dubbed Digital Versatile Disk, which is the name endorsed by the DVD Forum.

2.6.1 How DVD applies to the AS/400 system

DVD will be offered as an alternative to internal magnetic tape devices. It will be offered as the primary save and restore and software distribution device on select AS/400 systems. DVD is not intended to be a full tape replacement, but is intended to provide those AS/400 customers, who do not need tape, a lower-cost removable media alternative.

Watch for more information on DVD on the AS/400 system in future releases. For physical characteristics of DVD, see Chapter 3, "Storage options" on page 53.

2.7 Storage Area Network (SAN)

Storage Area Network (SAN) is a high-speed network that allows the establishment of direct connections between storage devices and processors. It can be compared to a local area network (LAN), providing the same function with different technology.

Note: In the AS/400 environment, SAN has also been referred to as System Area Network.

The information needs of business can grow quickly. An IBM SAN solution can help keep the information available as growth is incorporated.

IBM plans to deliver our SAN strategy in phases. This will help to leverage new technologies as they mature, and to help businesses seamlessly integrate SAN technology into their computing system infrastructure, while leveraging investments in storage, server, and application resources.

As SAN is evolving, IBM SAN technology is evolving in three phases:

- SAN-attached storage, leveraging the any-to-any hub connectivity of SAN technology
- SAN optimized storage exploiting SAN characteristics and delivering early SAN solutions
- SAN optimized systems leveraging the mature technology and delivering on the SAN promise of system-wide solutions

For an introduction to SAN, refer to the redbook *Introduction to Storage Area Network, SAN,* SG24-5470.

For additional information on SAN, refer to the following Web site:
`http://www.storage.ibm.com/ibmsan/index.htm`

Refer to the following Web site for a white paper to explore how deploying an IBM enterprise Storage Area Network (SAN) can help make the corporate information utility a reality: `http://www.storage.ibm.com/ibmsan/whitepaper/strategy.htm`

2.8 IBM Versatile Storage Server (VSS)

The IBM 2105 Versatile Storage Server (VSS) is designed to provide a flexible approach to storage centralization in support of server consolidation. By using the IBM 7133 Serial Disk Subsystem as its storage building block, Versatile Storage Server provides investment protection. With the IBM Versatile Storage Server, disk storage can be consolidated into a single, powerful system that offers many levels of advanced function. Examples include remote Web-based management, data sharing for like servers, and dynamic capacity allocation.

The highlights include:

- High-performance serial disk
- High-availability storage with RAID protection
- Flexible Web-based management with StorWatch Versatile Storage Specialist
- Exceptional scalability from 200 GB to 4.1 TB usable capacity

See Chapter 3, “Storage options” on page 53, for additional physical characteristics and IOP requirements for the VSS Storage option for the AS/400 system.

2.9 IBM Enterprise Storage Server (ESS)

The Enterprise Storage Server (ESS) offers high performance, attachment flexibility, and large capacity. Data from different platforms can be stored within a single high performance, high availability storage server. Storage consolidation can be the first step towards server consolidation, reducing the number of boxes you have to manage and allowing the flexibility to add or assign capacity when and where it is needed.

Note: The ESS is also a 2105, with different model numbers than the VSS.

ESS supports all the major server platforms: S/390, AS/400, Windows NT, and many flavors of UNIX. With a capacity of up to 11 TB, and up to 32 host connections, an ESS can meet both your high capacity requirements and your performance expectations. The maximum addressable storage varies by AS/400 model, up to 4.2 TB on a fully populated AS/400e server 740. Refer to the *Summary* sections of the *AS/400e System Handbook,* GA19-5486, to identify the maximum disk storage allowed by system model.

The enclosure of the Enterprise Storage Server incorporates dual RISC 4-way Symmetrical Multiprocessor (SMP) processors. Sixteen standard configurations range from 400 GB to over 11 TB. Each standard configuration provides a support infrastructure base of 6 GB cache, 384 MB nonvolatile storage (NVS), four device adapter pairs, and 9 GB, 18 GB, or 36 GB serial disks in a serial loop configuration. The ESS provides RAID-5 protection.

Enterprise Storage Servers support all platforms supported by the Versatile Storage Server, including RS/6000 running AIX and many leading UNIX variants, Netfinity, and other PC servers running Windows NT or Novell Netware, and Compaq AlphaServers running OpenVMS. In addition, the ESS supports System/390 servers running OS/390 code, VM, VSE, and TPF. Storage capacity is partitioned among the attached servers using the Web-based StorWatch ESS Specialist management tool.

See Chapter 3, “Storage options” on page 53, for additional physical characteristics and IOP requirements for the ESS storage option in the AS/400 system.

2.10 AS/400 DASD availability options

In today's environment, availability of the information system is a critical part of a company's needs. In this environment, it is critical for IT specialists to be skilled in understanding and developing plans to enable high availability. This section presents the concepts and procedures that maximize the availability of the system from a DASD perspective, with information on a variety of availability options.

The topics in this section provide an overview of ASP and single level storage protection options available for your DASD. This section covers topics involving

DASD management, including mirroring, check summing, data parity protection, integrated hardware disk compression, Auxiliary Storage Pool (ASP), and Hierarchical Storage Management (HSM). In order for you to understand and implement the following availability concepts, it is *imperative* that you read the *AS/400 Backup Recovery Guide*, SC41-5304. Be sure to use the correct version and release level of documentation for your system when planning your DASD configuration and protection.

2.10.1 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 (such as the bus, IOP, controller, or disk unit). It can be implemented on all models of the AS/400 system as it is part of the Licensed Internal Code.

Different levels of mirrored protection are possible, depending on what hardware is duplicated—bus, IOP, controller or disk unit. Mirroring allows the system to remain available during a failure of a disk-related hardware component, if the failing hardware component (and hardware components attached to it) are duplicated. For some system units, the failed hardware components can be serviced while the system remains available.

2.10.1.1 DASD mirroring within an ASP

Prior to OS/400 V3R7, there were four levels of mirrored protection offered for the CISC platform as well as in RISC.

A level of mirroring known as *power domain mirroring* occurs when the DASD IOPs and the buses they are attached to are isolated to different power sources, as in racks (for “white box” models) or towers (for the AS/400 Advanced Series models). With power domain mirroring, one rack or tower could lose power and all data would still be available to the users and the system would continue to operate. This level of mirrored protection is viewed as bus-level protection on disk protection screens visible through the use of DST or SST menu options.

Refer to Figure 16 on page 28 to observe the structure of each level of mirroring: device, controller, IOP, and bus.

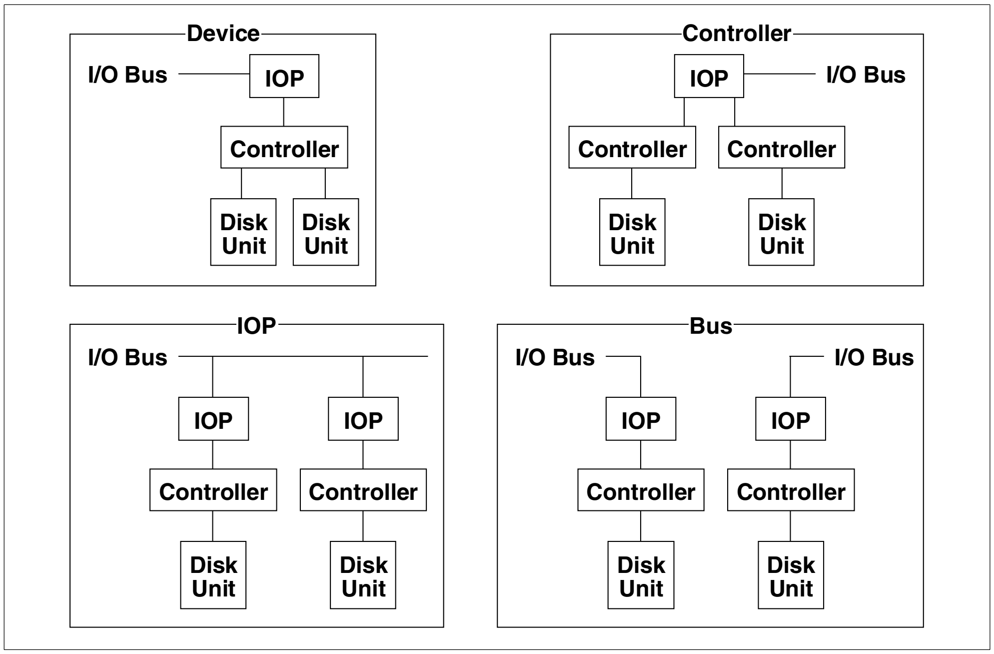


Figure 16. Mirroring levels

2.10.1.2 Remote DASD mirroring of the entire system or of ASPs

Remote mirroring support makes it possible to divide the disk units on your system into a group of local and a group of remote disk units. The remote disks attach to one set of optical buses. The local disks attach to a different set of buses.

The local DASD and the remote DASD can be physically separated from one another at different sites by extending the applicable optical buses to the remote site. The distance between the sites is restricted by the distance an optical bus can be extended. Limited cable distance enhances acceptable performance for synchronous operation, and local-like performance and reliability.

Note: Although optical technology supports distances up to 2 km (when 16 or less disk units are installed, up to 500 meters for System Unit Expansion Towers with 17 to 32 disk units), disk and high speed tape operations must be reviewed to support distances beyond 100 meters. Refer to RPQ 841958 *System Cabling Information* (available from IBM marketing and service) when ordering a variation from the default cable length.

For specifics about your installation, contact your service provider and refer to Site Preparation and Physical Planning at the AS/400 Technical Studio site:
`http://www.as400.ibm.com/tstudio/planning/plngstrt.htm`

When remote DASD mirroring is started, the local disk is mirrored to remote disk units. If a failure occurs at either the local or remote location, a complete copy of all data for the system still exists. The system configuration can be recovered from the unaffected mirrored members, and processing can continue.

If the distance between the main and remote location is such that a site disaster does not affect both systems, mirror all DASD in all ASPs of the system in local-remote pairs to help protect against a single site disaster.

Refer to Figure 17 for a visual representation of remote DASD configuration. The drawing represents a simple configuration of a 9406 model 7xx, with a remote bus mirrored to a single I/O bus tower.

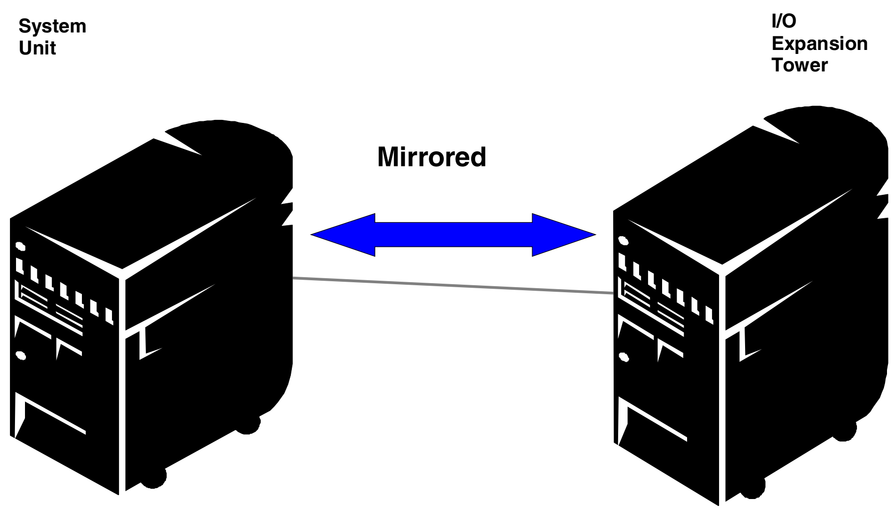


Figure 17. Remote DASD mirroring of the entire system or of ASPs

2.10.2 System level checksum protection

System level checksum is a precursor to RAID-5 protection. The use of system level checksum protection is restricted to the CISC platform family of products, which equates to those systems at V3R2 and earlier. The latest CISC release available is V3R2. We include an explanation here to allow a contrast to a similar, yet more system available-oriented protection method known as Device Parity Protection (DPY), discussed in 2.10.3, “Device Parity Protection (RAID-5)” on page 31.

System level checksum protection originated on the System/38 platform (the predecessor to the AS/400 system). One of the primary goals of this form of protection was to allow the user two opportunities to recover system information without a reload in the event of a single DASD failure. Those two opportunities came in the form of the following scenarios:

- Single DASD failure with a successful “save disk unit data” performed by a service representative, who then replaces the failed drive and performs a “restore disk unit data” process to complete the repair action.
- Single DASD failure without a successful “save disk unit data” performed. The service representative physically replaces the failed drive, verifying it is operational. Then during the IPL process, while in storage management recovery, the replaced disk units data is reconstructed using the other members of that checksum set for this rebuild.

Thus provided are two opportunities to “rescue” the missing data without the need for a total reload of the ASP associated with the single failed drive. In the following graphics, Figure 18 shows the checksum theory, Figure 19 shows the checksum recovery process, and Figure 20 shows checksum striping.

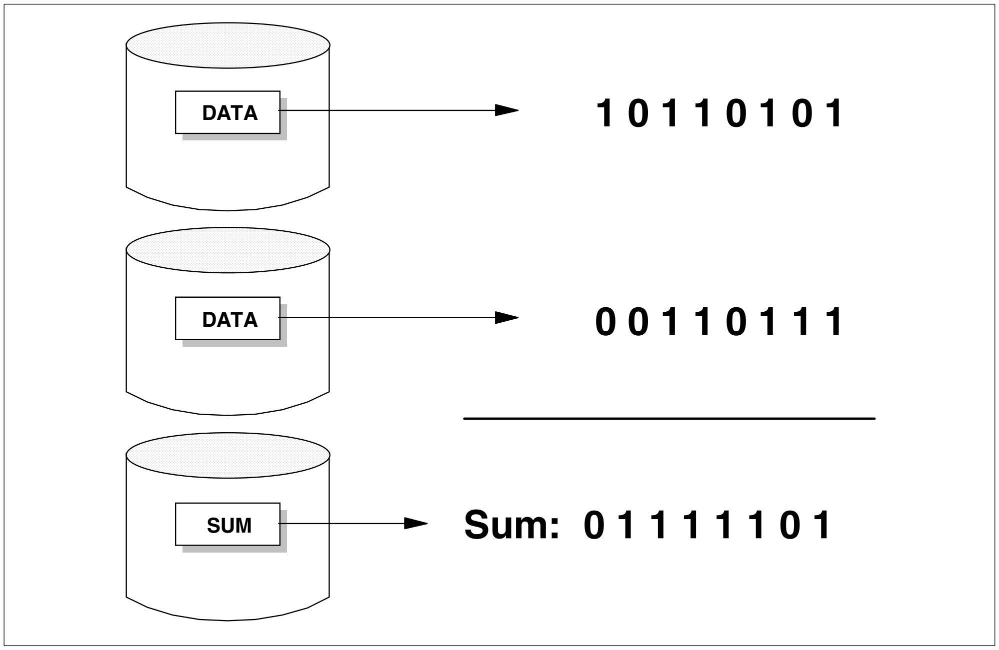


Figure 18. Checksum theory

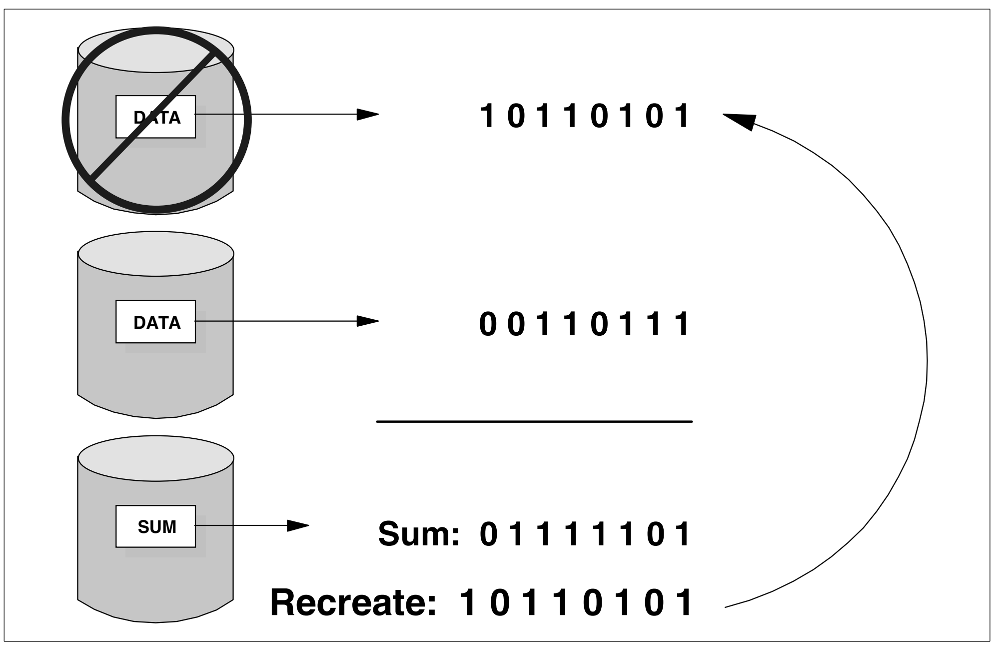


Figure 19. Checksum recovery

As portrayed, checksum data is calculated based on a pairing of like 0s or 1s. Identical twin numbers are written as “0”. A pair of numbers that are not alike is written as a “1” on the sum disk unit. In this way, it is always possible to calculate the data on any missing unit of the set.

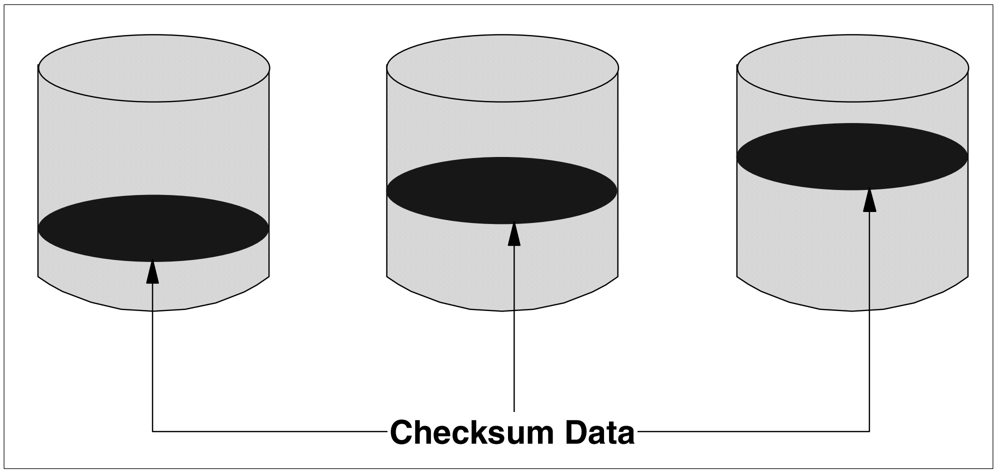


Figure 20. Checksum striping

System level checksumming is a valid protection method for CISC systems. However, there are times (particularly when older technology is involved) when more than one disk fails at the same time. In most cases, this requires a complete reload of the affected ASP to recover. While a single disk failure can be recovered from, concurrent maintenance does not apply. The system is unavailable to users for the duration of the recovery.

A more reliable protection method was designed and introduced, and so came about a similar, yet higher system availability-oriented method, known as Device Parity Protection (DPY). DPY allows for concurrent maintenance in a single failed DASD repair action, which translates to no associated downtime. The following section describes this method of protection.

2.10.3 Device Parity Protection (RAID-5)

Device parity protection is a hardware function that protects data from loss due to a single disk unit failure or because of damage to a disk. Calculating and saving a parity value for each bit of data enables the protection of that data. Conceptually, the parity value is computed from the data at the same location on each of the other disk units in the device parity set. When a disk failure occurs, the data on the failing unit is reconstructed using the saved parity value and the values of the bits in the same locations on the other disks in that set.

Device parity protection is a high-availability function that allows the AS/400 system to continue to operate when a single disk failure occurs. The system continues to run in exposed mode until the repair operation is complete and the data is rebuilt. If a failure occurs, the problem should be corrected quickly. In the unlikely event that a second disk fails, you can lose data.

For an example of a device parity-protected 9337, refer to Figure 21 on page 32.

DPY protection has been available for external DASD (such as the 9337) since OS/400 V2R2. The next DASD eligible for this protection method was DASD installed in a storage expansion unit. DPY is visually represented in Figure 21 and explained in the following text.

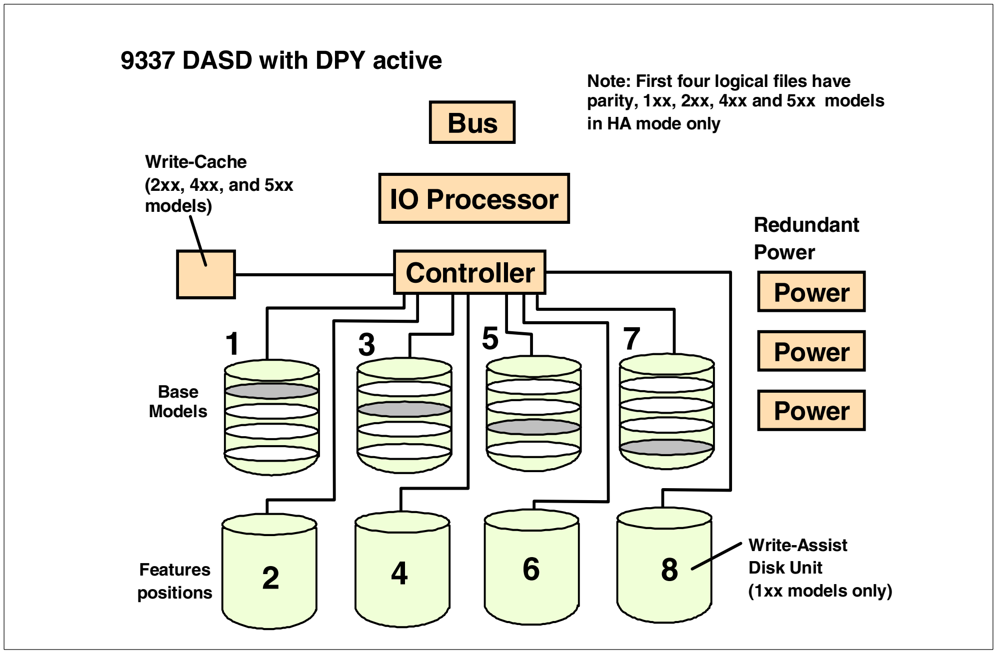


Figure 21. Device parity protection on 9337

Internal DASD installed in a #5051 or #5052 is available for the AS/400 Advanced series 3xx models. All internal DASD (except those attached to the MFIOP), are eligible for DPY if the proper DASD IOP is installed. Previously, DASD designated as internal were those installed in the card cage and logically associated with the MFIOP. These were not eligible for DPY.

For a detailed description of which DASD IOPs support RAID, see Appendix A, “Architecture” on page 171.

A minimum of four disk units of the same capacity is required for a valid Redundant Array of Independent Disk (RAID-5) configuration. Parity information can be spread across four or eight of the disk units in an array. Any number of drives (up to ten) can be protected. The set is automatically maintained as part of the RAID-5 protection feature. Internal disk units of different technology (that is, different feature numbers), but of the same capacity, can be either mirrored or RAID-5 protected.

RAID-5 protection is supported for all 1.03 GB, 1.96 GB, 4.19 GB, 8.58 GB, and 17.54 GB (1-byte or 2-byte disk units), provided that the disk controller supports RAID-5 protection.

When less than seven DASD are installed in a storage expansion unit (#5051, #5052 or #5058), the maximum number of drives to contain parity stripes is four. When using the storage expansion units #5052 or #5058, up to sixteen DASD can be installed. With this physical configuration, when all DASD installed are unprotected, the code creates two sets of 8 parity striped drives when device parity is started.

Refer to Figure 22 to observe the striping for internal DASD with less than eight drives installed.

Given the correct number of drives being physically available, attached to the same IOP and operational, the system stripes the maximum number of DASD for a set, to include eight members.

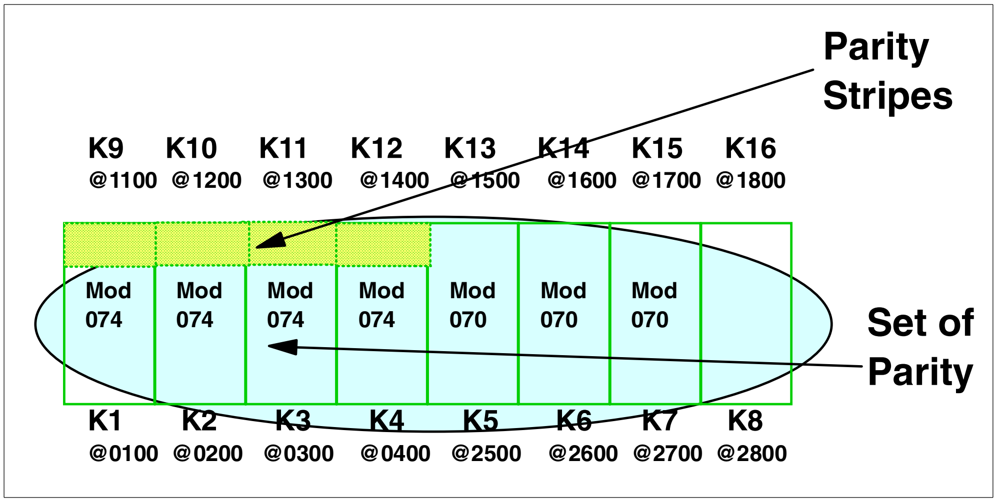


Figure 22. DPY: Less than eight drives installed

Note the model numbers of the drives in Figure 22. The models with a zero as the last digit do not contain parity stripes.

Spreading parity across eight disk units can provide better performance in the event of a disk unit failure, since the information required to dynamically rebuild the data on the failed disk after replacement is accessed from an eighth of the disk units, as opposed to one-fourth. The disk unit controller reads the parity and data from the same data areas on the other disk units to dynamically rebuild the original data from the failed disk unit to satisfy ongoing read requests. When data needs to be recreated, the controller generates the parity information for the failed disk unit as if it were still operating. As far as the AS/400 system is concerned, the subsystem continues to respond to I/O even though a single disk unit has failed.

A RAID controller is necessary when concurrent maintenance support is required. Of course protection must be activated to allow concurrent maintenance to take place. Use of concurrent maintenance is supported only to replace a failed drive with the same size replacement drive.

Figure 23 on page 34 shows an example of a fully populated #5052 or #5058 when DPY is activated. Note that two sets are generated, each having all members with parity stripes.

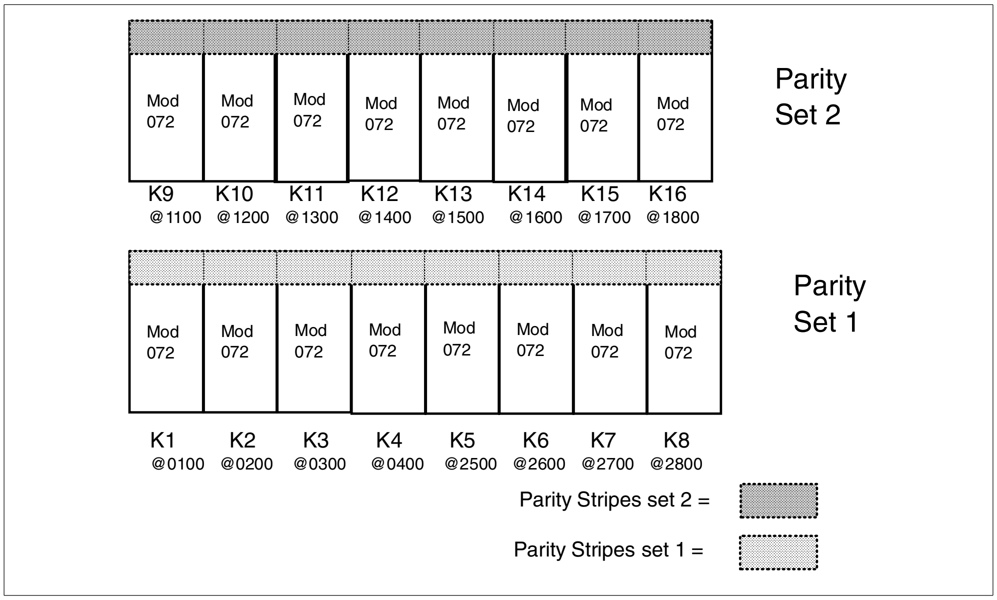


Figure 23. DPY: 16 drives (two DPY sets)

Notice that the last digit in the model number is a two. This is an indication that the drive is a member of an 8 striped drive parity set. Note in Figure 22 that the last digit on four of the DASD was a four. This is an indication that those DASD are members of a parity set that has only four striped drives. The other drives have full capacity available for customer data. In either case, no matter whether there are four or eight drives that contain parity stripes, each DPY set must provide the total capacity of one member of that set to store the parity information on. For example, if the DPY set is made up of four, 8 GB DASD, parity stripes add up to 8 GB of information for that set. If there were eight members (8 GB each) of the DPY set, each containing parity stripes, the set still requires 8 GB of space to store parity information for that set.

2.10.4 Integrated Hardware Disk Compression (IHDC)

Until V4R3, only software data compression could be implemented on the AS/400, and hardware data compression was performed by some tape drives. Thus, to compress data on DASD, software data compression had to be used. Software data compression is performed by the system and because of this, it can negatively impact system performance.

Beginning with OS/400 V4R3, the AS/400 system can implement DASD data compression utilizing specialized DASD IOPs. Just as Device Parity Protection (DPY) performed by DPY-enabled IOPs relieved the system processor of protection work, therefore, freeing the processor for other work, compression performed by compression-enabled IOPs compresses data, relieves the system processor of work otherwise performed by software compression.

Refer to Figure 24 for a visual representation of integrated hardware disk compression (IHDC).

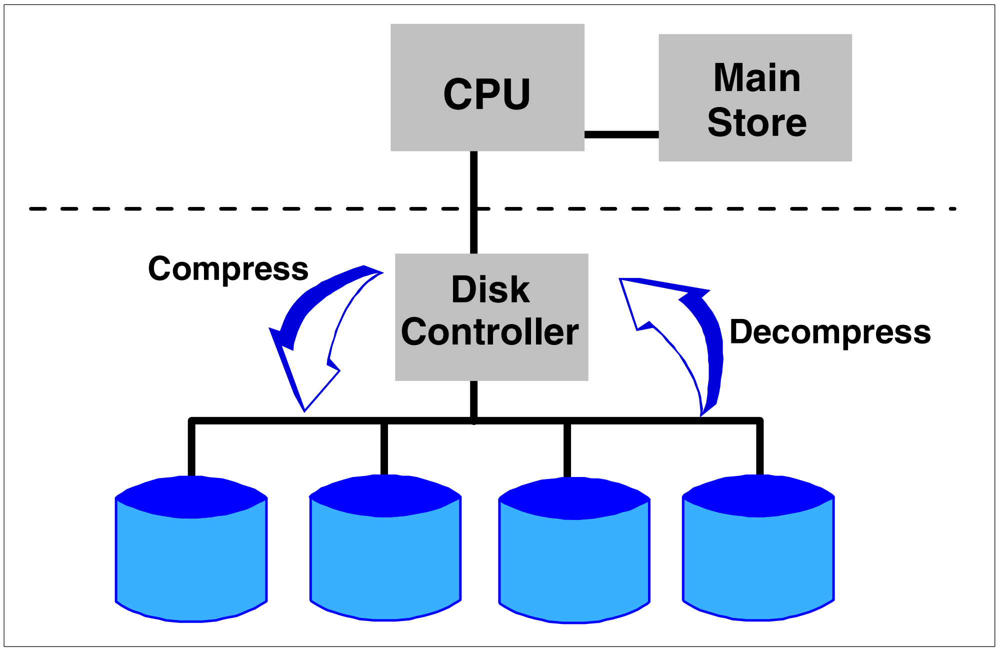


Figure 24. Integrated Hardware Disk Compression (IHDC) architecture

With hierarchical storage management (HSM), the DASD to receive demoted data must be in a user ASP. The user ASP can contain any combination of disk units—slower technology, compressed units, or faster technology. HSM provides an automated method to distribute data based on frequency of use across all levels of single level storage.

While compression is not required to use this form of data management (HSM), it provides a greater amount of capacity for the data.

Create what may be termed a “slow” ASP and assign older technology drives to that ASP. They operate with less throughput than new technology drives. Then, add new and faster technology drives to this ASP with compression active for them. Objects spread across this group of drives is compressed or not, depending on which DASD the data is stored.

This ASP with slower DASD assigned is a likely candidate to use for archiving, journaling, Internet applications and other processes that do not require high speed access to the data. Keep the production work that needs high speed disk access in the system ASP.

To improve performance of your production ASP (ASP 1), move less often used data to a separate ASP. The data remains available to programs, independent of ASP assignment. By compressing the less-used data, it takes less DASD space, thus lowering the overall cost of DASD. The process is managed by software and is transparent to end users. You can “grow” the storage by adding additional drives either through purchasing new drives or by using drives you already have on the system. User ASPs may be used for journals or to perform backup processes to minimize downtime.

Once disk compression is initiated on a system, status can be viewed through the Display Disk Configuration screen in Dedicated Service Tools (DST) or System Service Tools (SST).

There are minimum requirements for the type of hardware necessary to implement HSM.

There is no special requirement for drives. Any drive supported on the system can be compressed as long as the version/release of the software supports compression, and the appropriate DASD IOPs are installed for the specific drive type to be compressed.

For specific configuration rules, refer to Appendix C of the *AS/400 940x Installation and Upgrade* (publication number varies by version and release and AS/400 system model). At V4R4, all internal disk drives not in the system ASP can be compressed. On V4R3, the 17.54 GB drive is not supported for compression.

Refer to Appendix A, “Architecture” on page 171, for a discussion and listing of which IOP’s support compression.

Compression IOPs were introduced in tandem with OS/400 V4R2. To implement integrated hardware disk compression, the IOP must support compression. Only specific Ultra-SCSI controllers are compression-capable.

When a drive is compressed, there is at minimum a 2-to-1 compression ratio. It is possible to see as much as a 4-to-1 compression ratio. Most systems see something between the 2-to-1 and the 4-to-1 ratio. You can expect approximately 8 GB of capacity on a 4 GB drive, 2 GB capacity on a 1 GB drive, and so on. The amount of compression that takes place depends on the characteristics of the data that is being compressed. For example, source code compresses more than program code. Images are already stored in compressed form and gain little (if any) by further hardware compression.

Storage costs can be reduced when you use HSM and compressed DASD. Figure 25 shows an example of a system with one TB of data—100 GB of data used within the last 30 days, 300 GB of data used within one year but not within the last 30 days, and 600 GB of data not used within the last year.

HSM Example (1 TB of data):

300 GB Compressed Disk
600 GB Tape

	Price/MB	Pre-HSM Price	With HSM Price
High Perf Disk	.60	600,000	60,000
Compressed Disk (2x)	.30		90,000
Tape (3X)	.06		36,000
Total Price		600,000	186,000

Dramatic storage price reduction possible with HSM

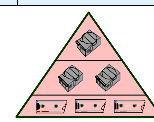


Figure 25. Savings example with HSM and DASD compression

Figure 25 shows a pre-HSM cost of storage versus the cost of storage when using HSM. Be aware that this is an example only. The cost savings for each system varies. The numbers are based on a unit cost.

2.10.5 Hierarchical Storage Management (HSM) on the AS/400 system

HSM allows the automatic, transparent migration of data from expensive or fast media to less expensive or slower media according to user-defined needs. Movement is in both directions between disks used for low-need objects and disks that have high-need objects.

Refer to Figure 26 on page 38 to observe the flow of HSM architecture.

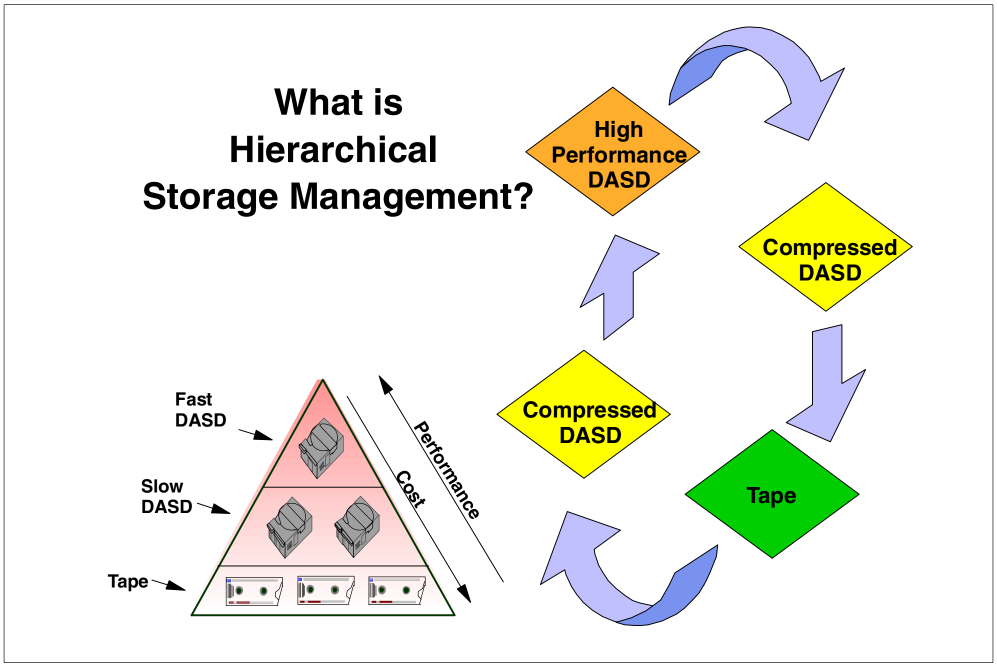


Figure 26. Hierarchical Storage Management

Data compression is used as one component of HSM. Data compression is discussed in more detail in 2.10.4, “Integrated Hardware Disk Compression (IHDC)” on page 34.

You have always had some ability to manage the hierarchy of data. In the past you could use software, for example Backup and Recovery Media Service (BRMS), or Report/Data Archive and Retrieval System (R/DARS), to manage moving data to a lower hierarchy. You could set up your software so that if an object is not used in a defined number of days, the software demotes it to tape or optical storage.

An automated tape library is the ideal for the archive. A single tape drive may be used but requires more operator intervention.

Starting with OS/400 Version 4 Release 3, this data movement strategy is significantly enhanced with the availability of HSM APIs. HSM enables the software to take advantage of another layer in the single level storage hierarchy, the DASD layer.

Note: BRMS/400 is an IBM product designed to automate this archive, as well as retrieval. In BRMS you set up policies to demote any object that has not been used in a set amount of days (for example 30 days). Demotion happens to slower DASD, typically to compressed DASD. If the data is not used for another set amount of days, it is further demoted to tape. The reverse is also true. For example, if you start accessing data on tape often, HSM (through defined rules with BRMS or other storage management products) promotes it back to DASD.

Note: This promotion and demotion applies to libraries only.

Refer to *Backup Recovery and Media Services/400,* SC41-5345, *Complementing AS/400 Storage Using Hierarchical Storage Management APIs,* SG24-4450, and

Hierarchical Storage Management Use, SC41-5351, for further information on BRMS/400.

2.10.6 Auxiliary storage pools (ASP)

ASPs were originally developed as a means of recovery to restore after system or storage failure. Other reasons that warrant the use of multiple ASPs are:

- You cannot afford to mirror or RAID-5 protect the entire disk storage subsystem, but you can mirror part of it.
- You want performance gains in specific high-need processes.
- You use functions that are not supported in the system ASP, for example, compressed DASD.
- You need to separate journal receivers from their associated journal.
- You need to separate application data requiring large objects and data access performance is not critical, for example, using 17.54 GB disk drives for image data.

Refer to Figure 27 for a visual example of multiple ASPs.

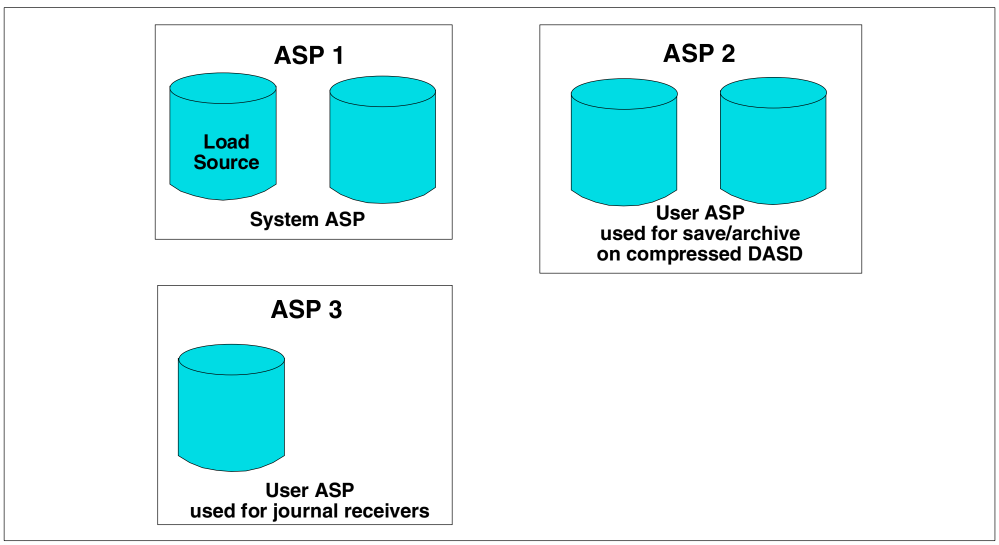


Figure 27. Auxiliary storage pools

2.10.6.1 Single level storage

To understand ASPs, it is valuable to first understand single level storage, also known as single address architecture. The following provides an analogy of single level storage.

When you put your money into a bank account at your local branch, in your view, the funds now sit in a drawer in that branch. This is your “virtual” view of your funds. In all actuality, the funds are invested in your neighbor's mortgage, a foreign business, a local business research department, or the government. All you (the system processor) are concerned about is that when you make a withdrawal, the funds are delivered to you. This is much like an application’s view of data. It does not care where the data is stored, only that it has fast access to the data when it chooses to do so.

Single level storage is achieved through the Technology Independent Interface, as pictured in Figure 28.

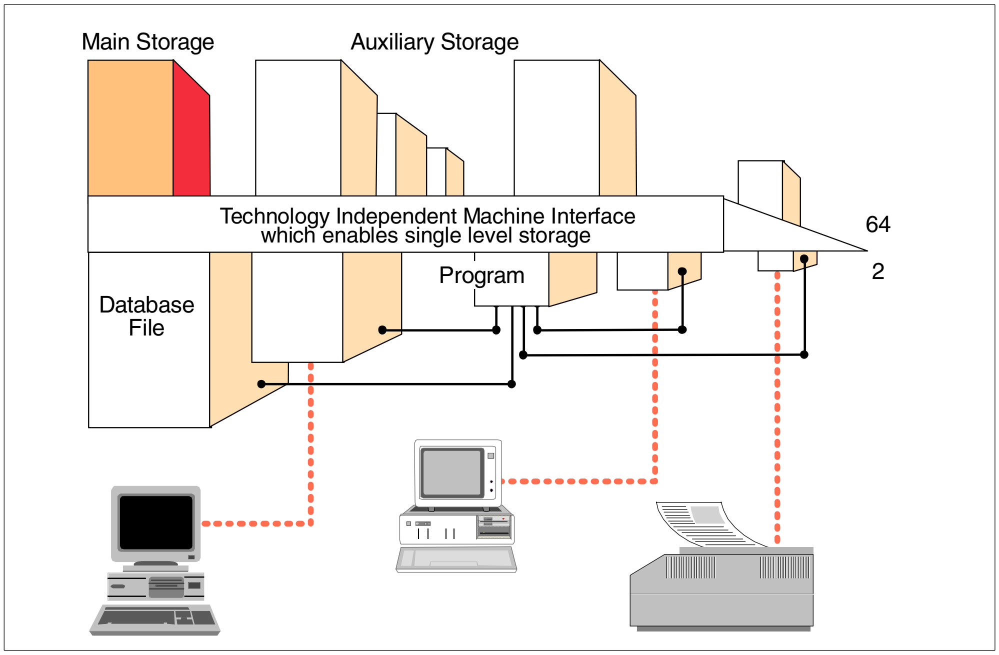


Figure 28. Single level storage

By scattering pieces of any given object among multiple drives, single-level storage enables multiple paths that can be used simultaneously by each portion of data to travel back to the processor. The speed of this data transfer is directly related to the speed of the slowest component in the architecture. This is typically the mechanical movement of disk actuators. Because this travel depends on the speed of each hardware component, single-level storage assists in overcoming the bottleneck.

In addition to understanding single level storage, it is important that you understand how disk units attach to the system.

To improve disk performance, data resides on auxiliary storage in disk areas that are not connected, as shown in Figure 29. This is referred to as a *scatter load*. Because you do not know where a given part of an object resides on disk, it is not evident what is stored on a disk when it fails. Because the system spreads the data for any object across more than one disk, it is likely that pieces of many objects are lost when a single disk fails. The system cannot continue without the object's missing data. No direct function exists to tell you which piece of an object was lost. It can be difficult to determine what was lost.

If you are unable to save the failing disk's data, your only recovery option is to reload the entire object. Or in many cases, reload the entire system from backup media. An alternative is to divide the pool of storage units and protect the availability of the DASD.

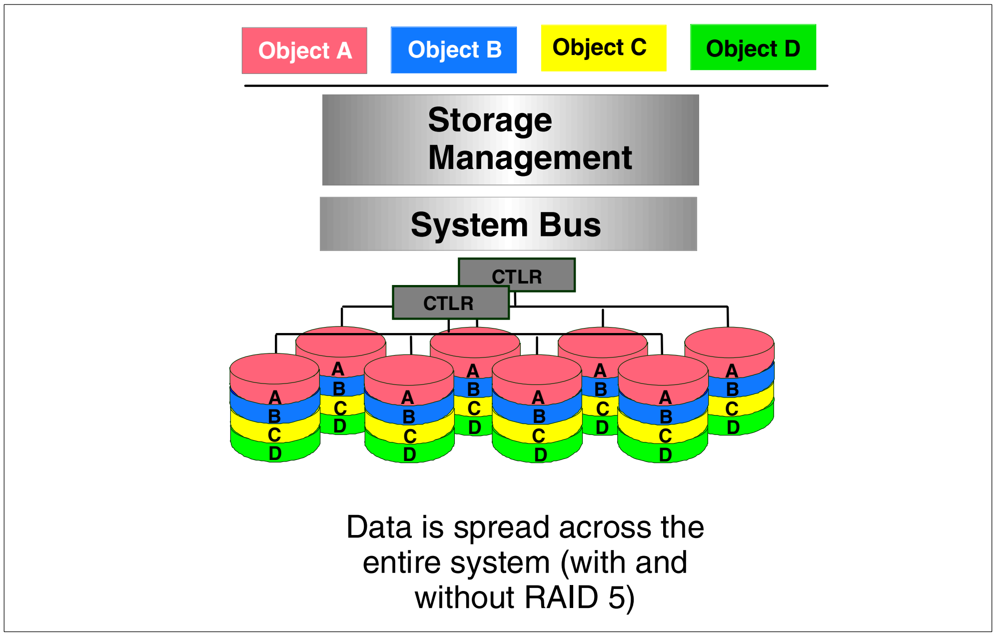


Figure 29. Scatter load on a single ASP AS/400 system

It is important to understand auxiliary storage pools (ASPs), because one of the tasks ASPs accomplish helps avoid potential scatter load concerns. ASPs are also required to implement a hierarchical storage solution to migrate, archive or retrieve data.

Means of isolating objects

An ASP is a group of disk units defined from all the available disk units that make up auxiliary storage. Non-configured drives are not assigned to an ASP. 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. See Figure 30 on page 42 for a visual representation of a multiple ASP system with objects scattered only within each ASP.

Disks are assigned to an ASP on a storage unit basis. When new disk units are physically attached to the system, the system initially treats each disk unit as a non-configured storage unit.

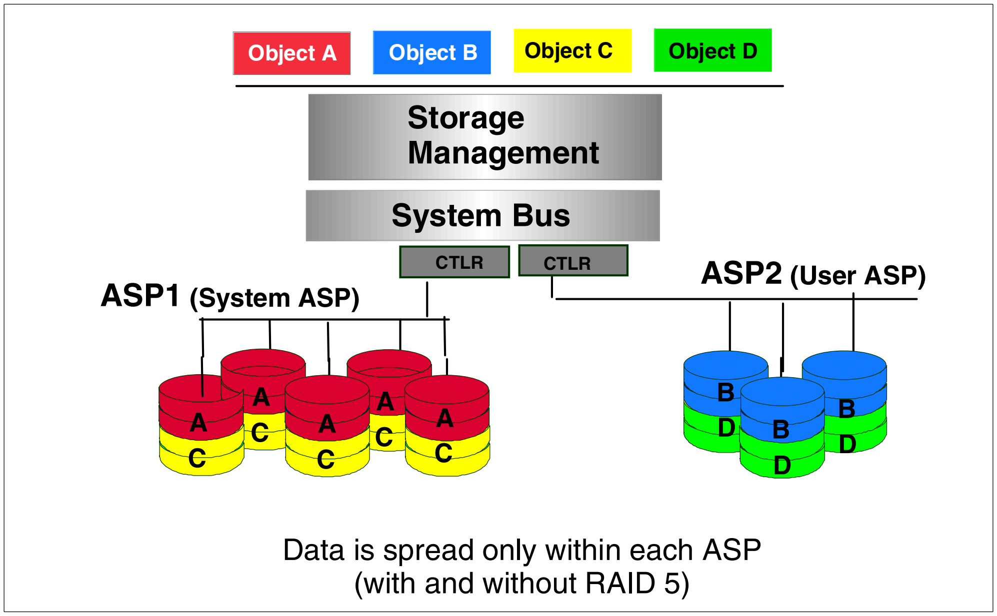


Figure 30. Scatter load on a multi ASP AS/400 system

Through Dedicated Service Tool (DST) options you can allocate these non-configured storage units to either the system ASP or a user ASP. The individual storage units within the disk unit are identified through the address field on the DST Display Disk Configuration screen. When you allocate a non-configured storage unit to an ASP, the system assigns a number to the storage unit.

Limited data loss with single disk failure

If the system experiences a disk failure with data loss, recovery is required only for the information in the ASP containing the failed disk. System and user objects in other ASPs are protected from the disk failure.

Possible performance improvement

In addition to a recovery advantage, placing information in an ASP can improve performance because the system dedicates the disk units associated with that ASP to the objects in that ASP. Performance can improve by isolating heavily referenced objects in a user ASP or by removing lightly referenced objects from the system ASP. For example, in a heavy journaling environment, contention between the journal receivers and the files is reduced if they are in different ASPs. This separation of workload can improve journaling performance.

It is often difficult to really predict whether multiple ASPs enhance (as in ability to use for journaling) or degrade (as in fewer arms available for high use system ASP) performance. Performance tools, such as PM400 and BEST/1, can assist you in sizing your system for DASD capacity needs.

Note

If the disk configuration contains disk units of significantly different capacities within an ASP, performance can degrade. Performance can also be significantly impacted by too few disk arms. The *Disk Arm Requirements* white paper offers more information and access to a tool to help assess this factor. You may also want to refer to the *Version Performance Considerations* white paper. Both white papers are available on the Web at:
`http://www.ibm.com.as400/developer/performance`

System ASP

The System ASP (ASP 1) is created by the system when LIC is installed. It is always configured. ASP 1 contains the LIC, licensed programs, and system libraries. The System ASP also contains all other configured disk units that are not assigned to a User ASP.

User ASPs

A user ASP is created by grouping together a physical set of disk units and assigning them a unique number, 2 through 16 (ASP 1 is always reserved as the System ASP).

User ASPs were originally developed for journaling, receivers, and save files. (Save files can store data, then offload to tape later, as in system backup, to reduce system down time). Later development allowed creating a library in a user ASP. This allowed pointers for the user ASP to be in the user ASP, allowing better ASP recovery. 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.

Refer to *AS/400 Backup and Recovery,* SC41-5304, for a list of object types that are not allowed in a user ASP.

Overflowed user ASP

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 handled as soon as possible because if a data-loss failure occurs that causes either the system ASP or the user ASP to be cleared, both ASPs are cleared. When creating user ASPs a threshold value can be defined using DST or SST menu options. From SST, select the Work with disk configuration option. Then, select the Work with ASP threshold option from the next menu and make your change. If the threshold is not specified, the default setting by the system is 90% for a system ASP or a user ASP.

As of V2R2, if a user ASP overflows, as objects are deleted from the user ASP, the overflowed objects in the system ASP are “copied” back into the user ASP during an IPL, as space becomes available.

Full system ASP

If the system ASP fills to capacity, the system ends abnormally. You must IPL the system and take corrective action, such as deleting objects, to reduce the storage utilization within the system ASP. To avoid filling the system ASP, specify a threshold value for the ASP. Use the system value QSTGLOWLMT or use SST or DST screens by selecting the menu option Work with disk configuration. Then, select the Work with ASP threshold. When the specified threshold value is reached, OS/400 warns the system operator of potential shortage of space.

For example, if you set the threshold value at 80% for the system ASP (the value specified in QSTGLOWLMT), the system operator message queue (QSYSOPR) and optionally the system message queue (QSYSMSG, if it exists) are sent messages when the system ASP is 80% full. It is up to the operator to take steps to delete files or obtain additional DASD before the system ASP becomes full.

Auxiliary Storage Lower Limit Action (QSTGLOWACN) is a system value that specifies the action to take when the available storage in the system ASP is below the amount specified in QSTGLOWLMT. The selected actions range from sending a message to the message queue, to a critical message queue, executing a program registered to handle the QIBM_QWC_QSTGLOWACN exit point, shutting the system down to a restricted state, to immediately powering down the system and restarting it.

Read more about full system ASP and overflowed user ASPs in Chapter 8, “Runaway DASD” on page 121.

2.10.7 Other protection and performance topics of interest

This section identifies other forms of protection for your data that typically adds flexibility and more complete recovery capabilities to your backup and recovery strategy.

2.10.7.1 Journaling

The main purpose of journal management is to enable recovery of the changes to a database file that occur since the file was last saved. You can also use journal management for:

- An audit trail of activity for a database file or other objects on the system
- Recording activity that occurs for objects other than database files
- A quicker recovery of access paths if your system ends abnormally
- A quicker recovery when restoring from save-while-active media
- Assistance in testing application programs
- Cross system journal and apply
- High availability and remote backups

Use a journal to define what files and access paths you want to protect with journal management. You can have more than one journal on your system. A journal may define protection for more than one file. Conversely, a given file can be protected by more than one journal receiver.

Figure 31 gives you a visual representation of one of the many uses of journaling.

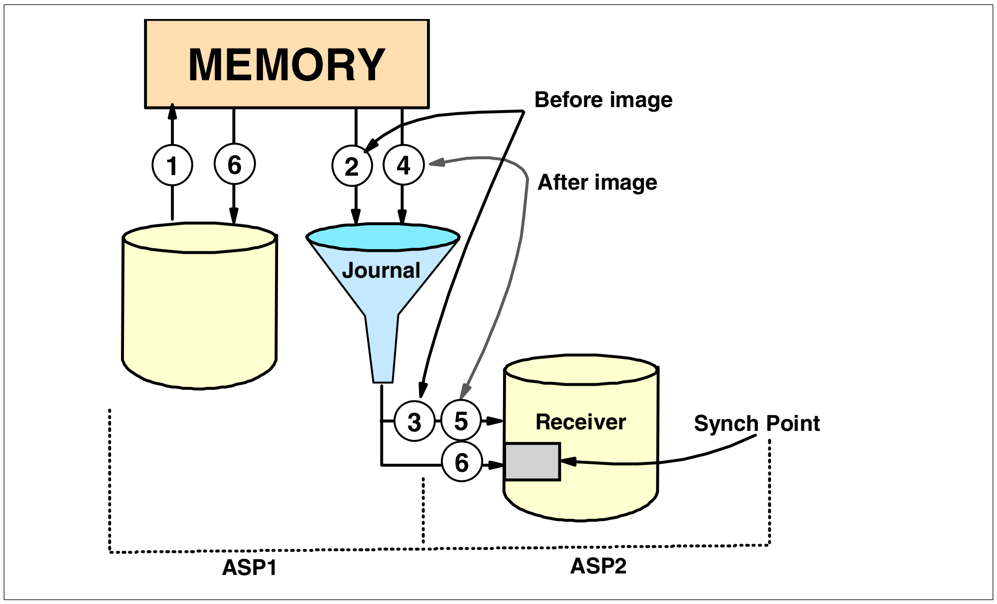


Figure 31. Journaling

In journaling, the following processes occur:

1. The file to be edited is paged into memory.

2 and **3**. A before image of the record is taken (this is optional). This before image copy is also taken for the access paths if access path journaling is also in effect.

4 and **5**. After the edit, the after image record update is journaled.

6. A sync point is reached when either 50,000 transactions have occurred since the last sync point, or a Change Journal (CHGJRN**)** command is issued.

Remote journal function

The addition of the remote journal capability in OS/400 at V4R2 offers a reliable and fast method to transfer journal receiver data to a remote AS/400 system. This function is offered in the form of APIs at OS/400 V4R2. Remote journals are ideal for use in data replication or high availability environments. At V4R3 the functions were enabled through CL commands.

Remote journals allow you to establish journals and journal receivers on a target system that are associated with specific journals and journal receivers on a source system. Once the remote journal function is activated, the source system continuously replicates journal entries to the target system.

The remote journal function is a part of the base OS/400 system and is not a separate product or feature. It is implemented at the Licensed Internal Code layer.

The benefits of the remote journal function include:

- It lowers the CPU consumption on the source machine by shifting the processing required to receive the journal entries from the source system to the target system.

- It eliminates the need to buffer journal entries to a temporary area before transmitting them from the source machine to the target machine. This translates into less disk writes and greater DASD efficiency on the source system.
- Since it is implemented in microcode, it significantly improves the replication performance of journal entries and allows database images to be sent to the target system in real-time. This real-time operation is called the *synchronous delivery mode*. If the synchronous delivery mode is used, the journal entries are guaranteed to be in main storage on the target system prior to control being returned to the application on the source machine.
- It allows the journal receiver save and restore operations to be moved to the target system. This way, the resource utilization on the source machine can be reduced.

For additional reading on remote journals, refer to the redbook *AS/400 Remote Journal Function for High Availability and Data Replication,* SG24-5189.

2.10.7.2 Clustering

Cluster technology is implemented on the AS/400 to provide availability during both planned and unplanned outages. Starting from 2.1, "Direct access storage device (DASD)" on page 19, and onward, it should be clear that the AS/400 developers have spent a significant amount of design resource to enable high levels of availability in a single system environment. This highly reliable design (as measured in a Gartner Group report) has long been used for unplanned outages by the AS/400 customer. In today's world, however, it is becoming increasingly clear that there is no time for any unplanned outages, or even planned outages, for that matter. Clusters are implemented to reduce down time due to planned outages and site disasters. In an AS/400 installation, the lion's share of down time comes from planned outages. Providing system availability during planned outages results in increased unplanned outage coverage as a side benefit.

To further address the needs of the continuous availability market, the AS/400 development team is investing in cluster technology. Significant advanced features and functions were introduced in V4R4 that put the AS/400 system in a leadership position. Cluster technology has been around for many years but only recently have common terms and concepts begun to emerge. The AS/400 system design has adopted these concepts and integrated the infrastructure needed to achieve a level of standardization, an advancement of cluster technology, and the involvement of the Solution Developer (SD) and the cluster middleware business partner in the total solution. While the AS/400 system provides the basic cluster infrastructure, the cluster middleware business partner provides data resiliency and the Solution Developer provides application resiliency. Together, these three parts provide the whole solution to continuous availability.

Cluster Resource Services represents the cluster technology that AS/400 provides in V4R4. It consists of an open set of APIs which applications can use to create and manage a cluster.

Data Resiliency is the maintaining of one or more copies of the application data on one or more separate systems. DataMirror, Lakeview Technology, and Vision Solutions have provided sophisticated data replication solutions for many years.

New or updated products that take advantage of cluster resource services are available by the cluster middleware business partners.

DataMirror announced availability of their iCluster product suite beginning July 14, 1999. Vision Solutions announced availability of their Vision Suite Version 6.3 starting July 15, 1999.

Cluster Management is new for V4R4. This is the entry point for the customer who has the responsibility for first choosing a cluster middleware business partner's product. When installed, the cluster management features are used to configure the customer's systems into a cluster and define how the systems participate in the backup strategy. Then the application which is to be activated as highly available is selected. Cluster Management handles ClusterProven for AS/400 applications, providing automatic configuration and activation.

Vision Solutions' Vision Suite version 6.3 handles ClusterProven for AS/400 applications.

For a simple example configuration, refer to Figure 32 on page 48.

To read more about clustering, visit these Web sites:

- `http://www.as400.ibm.com/ha/sec2.htm`
- `http://www.as400.ibm.com/ha/sec23.htm`
- `http://www.as400.ibm.com/ha/sec24.htm`

2.10.7.3 Multiple Systems

Prior to the availability of a clustering implementation, duplication of systems was performed in a less efficient manner—through a communications link (Token-Ring, OptiConnect, dial-up, or other). With a central site or home office installation in place, a second (and sometimes third) system is separately maintained at a remote site location. The systems are kept in synchronization by periodic transmission of updates through the communication link. For example, the home office sends updates to the remote location for a duplicate set of data. The reverse could also be true.

In this implementation of multiple systems, the system administrator is responsible for twice the amount of documentation and management.

An example of a multiple system configuration can be observed in Figure 32.

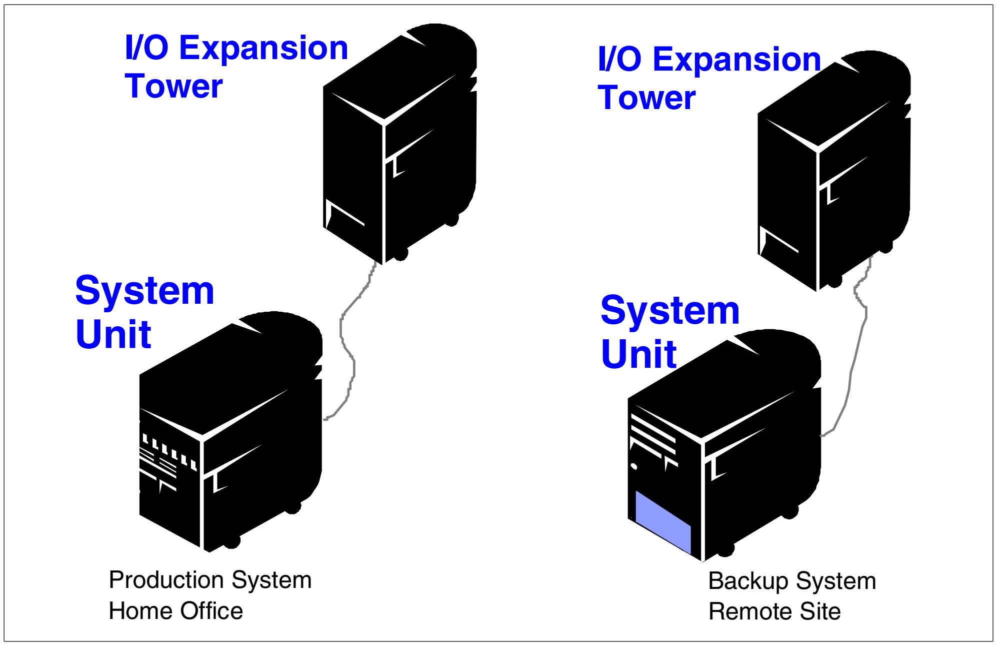


Figure 32. Example of multiple system setup

2.10.7.4 Considerations for measuring availability

The availability options described in this chapter may or may not be appropriate for your installation. There are many possible solutions available in the AS/400 product line. Each site must consider what best suits them.

When measuring availability, a question to ask is, “What is actually being measured?” Single system availability measurements can be measurements of hardware reliability, or hardware and operating system software, or the measurement can include applications. Solution availability takes into account all these components.

There is also a distinction to be made between two types of server outages:

- Planned outages take place when the operations staff takes the server offline to perform backups, upgrades, maintenance, and other planned events
- Unplanned outages occur due to unforeseen events such as a power loss, a hardware or software failure, system operator errors, security breaches, or a natural disaster.

Measuring unplanned outage time reflects the inherent reliability of the system. Measuring planned outage time brings into focus the world of clusters.

There are five general levels of system availability. The application running and the losses to occur in the event of a system outage determine which level you set a strategy for:

- **Base availability**: Systems are ready for immediate use, but will experience both planned and unplanned outages.
- **High availability**: Systems incorporate technologies that sharply reduce the number and duration of unplanned outages. Planned outages still occur, however, the servers include facilities that reduce their impact.

- **Continuous-operation environments**: Use special technologies to ensure that there are no planned outages for upgrades, backups, or other maintenance activities. Companies use high-availability servers in these environments to reduce unplanned outages.
- **Continuous-availability environments**: Go a step further to ensure that there are no planned or unplanned outages. To achieve this level of availability, companies use dual servers or clusters of redundant servers in which one server automatically takes over if another server goes down.
- **Disaster Tolerance environments**: Require remote systems to take over in the event of a site outage. The distance between systems is very important to ensure no single catastrophic event affects both sites. However, the price for distance is loss of performance due to the latency time for the signal to travel the distance.

To make an informed decision as to the appropriateness of these or any other options discussed in this redbook, refer to *AS/400 Backup and Recovery,* SC41-5304, for more detail.

2.11 Considerations for ASP management

Operations on libraries and objects contained within a given ASP is possible through selected ASP management tools. Backup Recovery Media Services/400, (BRMS/400) licensed program 5769-BR1 provides a user interface for some management functions, as described in this section. Control Language (CL) commands, such as Create Library (CRTLIB) and Work with Library (WRKLIB), have an Auxiliary Storage Pool parameter (ASP) to manage and control the placement of objects within the system or a given user ASP.

Note: On the CISC platform, there is an optional purchasable product called WRKASP Utility (WRKASP), which assists in the management of multiple ASP configurations. V4R1/V4R2 was the last release to support the WRKASP PRPQ. Beginning with the release of OS/400 V4R3, some of the functions of the WRKASP utility are incorporated into base OS/400 code and BRMS/400.

2.11.1 OS/400 command usage or BRMS/400

Refer to Table 1 for a cross-reference of WRKASP functions built into OS/400 or Backup Recovery Media Services/400 (BRMS/400) code, and in which release they are available.

Table 1. WRKASP functions cross-reference table for OS/400 and BRMS

WRKASP functions	V4R3	V4R4
STRSST menu **Work with disk units,** **Display disk status:**		
- Disk Units allocated to the ASP	No	OS/400 (0)
- Serial number	No	OS/400 (0)
- Disk unit type	No	OS/400 (0)
- Disk unit model type	No	OS/400 (0)
- Size	BRMS	BRMS, OS/400 (0)

WRKASP functions	V4R3	V4R4
- % Used	BRMS	BRMS, OS/400 (0)
- Protection provided	BRMS (1)	BRMS, OS/400 (0) (5)
- Protection status	No	OS/400 (0)
Save Libraries in an ASP:		
- Save entire libraries	BRMS	BRMS
- Save changed contents of libraries	BRMS	BRMS
Print libraries in an ASP	No	No
Print ASP analysis	No	No
Display recovery for Access Paths	OS/400 (DSPRCYAP)	OS/400 (DSPRCYAP)
List types of ASP (system-, object-, or library-based ASP)	No	No
ASP overflowed (Yes/No)	No	OS/400 (0)
Size of ASP that is protected and unprotected	No	OS/400 (0)
% of ASP that is protected and unprotected	No	No
ASP overflow amount	No	OS/400 (0)
ASP threshold message	No	OS/400 (0)
Move libraries from one ASP to another	BRMS or OS/400 (QHSMMOVL API)	BRMS or OS/400 (QHSMMOVL API)
Copy libraries to another ASP	No	No
Work with contents of ASP:		
- List the libraries in a library-based/system ASP	No	OS/400 QYASPOL API Open List of APIs
- Copy library to ASP	No	No
- Display	No	Investigating
- Print DB	No	No
- Display description	No	Investigating
- Move library to ASP	BRMS or OS/400 (QHSMMOVL API)	BRMS or OS/400 (QHSMMOVL API)
- Work with library	No	WRKLIB LIB ( ), ASP ()
- Check dependencies	OS/400 QHSMMOVL HSM API (2)	OS/400 QHSMMOVL HSM API (2)
- Display overflow status	QUSRTOOL (4)	QUSRTOOL (4)
- Reset overflowed objects	No	No

WRKASP functions	V4R3	V4R4
Work with journal objects on the system	OS/400 WRKJRN, WRKJRNRCV	OS/400 WRKJRN, WRKJRNRCV
Work with SAVF objects on the system	OS/400 WRKF FILE(*ALL) FILEATR (SAVF)	OS/400 WRKF FILE(*ALL) FILEATR (SAVF)
Print disk mapping	No	No
Allow naming (description) of an ASP	BRMS	BRMS
Codes: BRMS = Function is in BRMS/400 Product (5769-BR1) Investigating = We are considering this function for a future release Comments: (0) = New Retrieve ASP Function. (1) = Shows mirroring only. (2) = *Validate can be specified on the check object dependencies. (3) = WRKJRN and WRKJRNRCV can be used to work with journaled objects. (4) = QUSRTOOL contains the source for the DSPOVFOBJ which is a tool to help you find all the overflowed objects. (5) = Does not provide the mirror level of protection for a disk unit.		

As you can see by the table, additional functions from WRKASP are being investigated to determine the feasibility of incorporation for future releases.

Refer to *Complementing AS/400 Storage Management Using HSM APIs*, SG24-4450, for more information on ASP management.

2.12 Summary

This chapter explored many levels of the storage architecture in the AS/400 system environments. The other issue discussed in this chapter includes identifying and describing various means of DASD protection. The final choice remains that of the user. The intent of this chapter was to identify options available and offer an overview and description of each of them, and provide pointers to additional reading so that a decision can be made to meet your system requirements for availability and ease of recovery.

Chapter 3. Storage options

This chapter identifies many options available for storage media of a permanent nature. The components described include current devices, as well as an introduction to storage options new to the AS/400 system and how they can fit into the structure of your AS/400 system environment to enhance and strengthen your storage needs.

Non-removable storage is described in Appendix C, “Non-removable storage (DASD)” on page 195.

3.1 Digital Versatile Disk (DVD)

Digital Versatile Disk-Random Access Memory (DVD-RAM) is a rewriteable, optical storage technology that offers significant cost savings over the current tape and CD-ROM technologies. The AS/400 DVD device will be functionally equivalent to tape and CD-ROM for save and restore, software installation, backup and disaster recovery, and for service personnel to use when taking main store dumps. Since the DVD drive is downward read compatible to CD-ROM media, CD-ROM media from software distribution (PID) can be read using DVD.

DVD RAM media has a capacity of 4.7 GB on a single side compared to CD-ROM 650 MB capacity. A DVD device will occupy the same integrated tray now used by CD-ROM devices on the AS/400 system. Similar to CD-ROM, the DVD media is 120 mm optical media. The DVD-RAM media may require the use of a cartridge carrier similar to that used by the 130 mm IBM optical media. The drive reads CD-ROM media as well as DVD-ROM media. Neither requires a cartridge and can be loaded as is into the same media tray that also fits a cartridge. In either case, all media used is front-loaded into the drive.

For additional detail regarding uses of DVD in the AS/400 system, refer to Chapter 2, “Storage architecture” on page 19.

3.2 CD-ROM

AS/400e server code is distributed on CD-ROM media. The CD-ROM drive is standard on all AS/400 models and is therefore not identified with a separate feature on the system unit. It can also be used for alternate IPL, but not as a save and restore device for the system. A maximum of one CD-ROM can be ordered per expansion tower.

LPAR support and CD-ROM feature descriptions

See Table 2 on page 54 for specific feature code and installation rules and requirements.

© Copyright IBM Corp. 2000

Table 2. CD-ROM options

Feature Code (FC)	Description/Rule
#4425	Installed in Storage/PCI Expansion Tower #5065. Used as alternate IPL (IBM distributed CD-ROM media only) and program distribution. One per secondary partition. This feature is customer-installable. #2748 Storage Device Controller in Slot 6 in the #5065.
#6325 (Optional CD-ROM)	Installed in System Unit Expansion (SUE) #5072, and #5073 for Models Sxx, 6xx, and 7xx. Maximum of one per I/O tower and Model 740. One per Model 730. Requires FC 2624 Storage device controller. Limits use of tape in same tower to #6380 and #6390.
#6425 (Optional CD-ROM)	Installed in Models S20, 620, and 720 or 9329 PCI Integrated Expansion Unit. Requires PCI disk unit controller - #2726, #2740, #2741 or #2748. Not supported in #9331 SPD Integrated Expansion Unit.

These CD-ROM features are only usable when installed in conjunction with the Logical Partitioning Support in OS/400. All features require OS/400 V4R4.

For more detailed information regarding your CD-ROM options by individual system models reference the *AS/400 System Builder*, SG24-2155, or the *AS/400 Installation and Upgrade Guide*, SY44-5950.

3.3 Optical storage device

Optical storage on the AS/400 system provides an efficient way to store and retrieve large amounts of information at a high performance level and an economical price. Compared to other high-capacity storage devices, such as tape and microfilm, optical storage offers faster access times and a hierarchical-type file organization. AS/400 optical storage uses files stored in directories and subdirectories similar to PC-based file systems or UNIX. Optical storage devices, such as CD-ROM and LAN-connected optical storage, use the Hierarchical File System (HFS) software interface that optical storage has always used.

Three categories of optical media are available to meet most storage requirements:

- CD-ROM
- WORM (write once read many) optical cartridge
- Erasable optical cartridge

CD-ROMS are single-sided disks and are the same size as those commonly used in audio CD players. The CD-ROM data format is identical to the one used with personal computers. Therefore, it may be possible to develop CD-ROMs for use on both personal computers and the AS/400 system. The other two forms (WORM and erasable) are two-sided disks that are contained within a cartridge shell.

CD-ROM is a read-only format that is optimized for read performance. CD-ROMs are ideal for the wide-scale distribution of programs and data. Access to CD-ROM is supported through the Hierarchical File System, Integrated File System, and save and restore interfaces.

All AS/400 Advanced Series systems include a rack-mounted CD-ROM drive that is ideal for program and data distribution. The CD-ROM drive is primarily intended as a program and data delivery device. Even though many users can potentially access it simultaneously, the drive accesses one CD-ROM at a time.

WORM storage is an economical way to archive data, yet still have it quickly and easily accessible. WORM media is available in 1x (650 MB), 2x (1.3 GB), and 4x (2.6 GB) capacities. An erasable cartridge offers the most flexibility with the same capabilities as magnetic (DASD) storage, but with much higher capacities and lower cost. Erasable media is available in 1x (650 MB), 2x (1.3 GB), and 4x (2.6 GB) capacities. Both WORM and erasable cartridges have a sector size of 1024 bytes/sector.

Optical media libraries come in a variety of configurations designed around the different forms of media and connection options. Optical media libraries range from a single cartridge standalone model through models that are capable of loading 258 optical cartridges and four disk drives. Optical media libraries may be directly connected to the AS/400 system for best functionality and performance. Or they may be connected through a LAN to allow independent access by PCs or other AS/400 systems.

For additional reading on optical library solutions, especially the 3995 optical library, refer to this site on the Web:
`http://www.storage.ibm.com/hardsoft/opticalstor/index.html`

Optical storage device connectivity

You can connect optical media libraries to your AS/400 system in two ways: directly attached and LAN attached.

In the directly-attached method, a multi-wire cable connects the library to an I/O processor (IOP) card within the AS/400 system. Directly-attached libraries support nearly all Hierarchical File System (HFS) application programming interfaces, most Integrated File System commands, and AS/400 save and restore commands. These libraries may be accessed by other LAN-connected systems by using the Integrated File System.

LAN-connected optical media libraries can be used only through the HFS interface. The libraries can be accessed simultaneously either by several AS/400 systems or any other devices on the LAN. LAN-connected optical media libraries have a controlling PC and do not require an AS/400 system that acts as a controller. LAN-attached optical media libraries are often referred to as optical servers.

3.4 Removable storage (tape)

Removable storage is a functional unit that houses a magnetic media onto which data is written, can be retained at a location physically apart from the system unit, and from which data can be retrieved onto online storage (usually disk). For the purposes of our discussion, “tape units” are termed *removable storage*.

Tape media provides relatively large amounts of storage space with a greater capacity than direct access storage drives (DASD or disks), but slower speed (and, therefore, less cost on a MB per MB scale). Rewritable tapes provide as storage capacity that is practically unlimited with the capacity divided into separate media units. Removable tape allows you to transport data from system to system (assuming each system is equipped with compatible drives) and provides offline storage for archival and recovery purposes.

In selecting tape storage to solve a business requirement, both the financial and data requirements need to be examined. Aspects of data requirements include the amount of data to be stored, the performance when reading and writing data, archive and retrieval capabilities, and the level of automation required.

For a table that compares tape subsystems that can attach to the AS/400 system, refer to the *AS/400e System Handbook*, GA19-5486. This table indicates whether the attachment IOP supports Hardware Data Compression (HDC) and whether the tape subsystem controller supports a compaction algorithm, either Improved Data Recording Capability (IDRC) or Lempel Ziv 1 (LZ1). These algorithms enable more data to be written to tape up to the maximum shown. You can also locate this table on the Web at: `http://www.as400.ibm.com/handbook/5486MagC.html`

The *AS/400e System Handbook*, GA19-5486, also offers a detailed listing of magnetic media controllers, and devices and the AS/400 models to which they attach.

3.5 Summary

For additional reading on storage options, visit the storage solutions Web site at: `http://www.storage.ibm.com`

Chapter 4. Extended Adaptive Cache (EAC)

Extended Adaptive Cache (EAC) is an intelligent *read cache* function and is *built into the firmware* of the PCI RAID Disk Unit controller (feature code #2748 for V4R4 system). It is designed to identify and keep pieces of frequently-read data from AS/400 disk storage in a Read Cache Device (RCD) to help reduce I/O read requests to the disk units. This can help improve performance of read-intensive applications.

Prior to EAC, OS/400 Expert Cache was the main tool for read-intensive applications. Expert Cache is designed to primarily serve applications that perform sequential access to data. While EAC is more tuned to random access, it also helps sequential access. So, the two caches work in complement with each other.

EAC is different from the 26 MB *write cache,* which is also built into the #2748 disk controller adapter. The write cache *keeps* pieces of data to be written to AS/400 disk units attached to the controller. Because the data is available in the cache, the I/O wait time for applications that write data to disk storage is reduced.

The RCD contains 1.6 GB of solid-state memory components packaged in a physical form that resembles an AS/400 disk unit. It attaches to the #2748 controller by mounting into a disk slot. EAC makes use of the Read Cache Device (separately orderable feature codes #4331 and #6831) to store pieces of cached data. See 4.4, “The 1.6 GB Read Cache Device” on page 63, for further information.

With a #2748 disk controller installed on a V4R4 system, the performance benefit of EAC can be estimated by running an EAC Simulator. The simulator is available from V4R4 Management Central as a tool to collect information on disk I/O operations. The Performance Tool/400 program product is then used to analyze the collected information and produce a report that provides an estimation of performance improvement in the disk read operations. See 4.2, “Extended Adaptive Cache Simulator” on page 60, for further information.

4.1 Extended Adaptive Cache benefits

EAC algorithms are streamlined for performance. They do not create the bulky overhead found in other, more complicated cache management designs. EAC uses predictive algorithms designed through statistical means to understand the flow of data to the disk storage. Then the read cache device is populated only with the pieces of data that have a history of actively being read and likely to be read again.

EAC continually observes all I/O activities handled by the #2748 disk controller without putting any burden onto the system processor workload. When it determines that a certain piece of data is worth being cached, it uses subsequent disk I/O operations and manipulates them to bring the desired piece of data into the read cache device. This eliminates the need for additional disk I/O operations to the disk units in order to populate the read cache device.

The design of EAC is based on specific data management strategies of OS/400. Whether the disks under the supervision of the #2748 disk controller are RAID-5

© Copyright IBM Corp. 2000

protected, mirrored, or unprotected, the data stored on the disks has a tendency to be identified in *bands*, as illustrated in Figure 33. This means that there are physical contiguous areas of disk storage which fall under one of the following categories:

- Areas where the data is actively read (a read-only band)
- Areas of data that are both actively read and written to (a read/write band)
- Areas that are frequently written to (a write-only band)
- Areas of storage that are not frequently accessed (a random band)

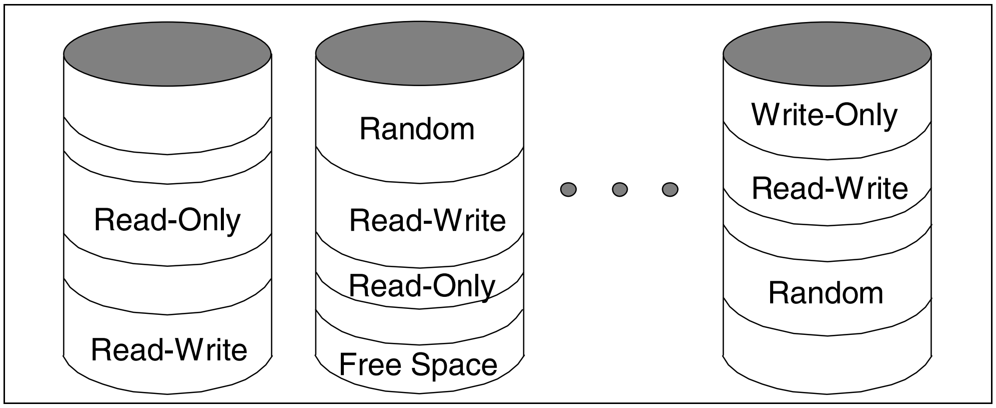


Figure 33. A conceptual illustration of data "banding" on disk storage

This "banding" of data is accounted for in the design of EAC. The goal is to cache bands identified as read/write, and read-only. A band that is identified as write-only remains largely unaffected.

EAC is also designed not to degrade the performance of large blocks of data that are either sequentially written or sequentially read. They are handled separately by the read buffer in the disk unit, OS/400 Expert Cache, and write cache on the #2748 disk controller.

EAC is designed to work effectively with other caches in the system. Although the EAC works independently from Expert Cache and does not require Expert Cache to be activated, it takes the Expert Cache strategy into account as it tracks the physical read requests flowing to disk units.

EAC is logically positioned directly below the write cache of the #2748 disk controller, as shown in Figure 34. As the write cache is tailored to ensure that 100% of disk write operations are cache write operations, EAC can optimize performance on read requests without degrading the write cache effectiveness.

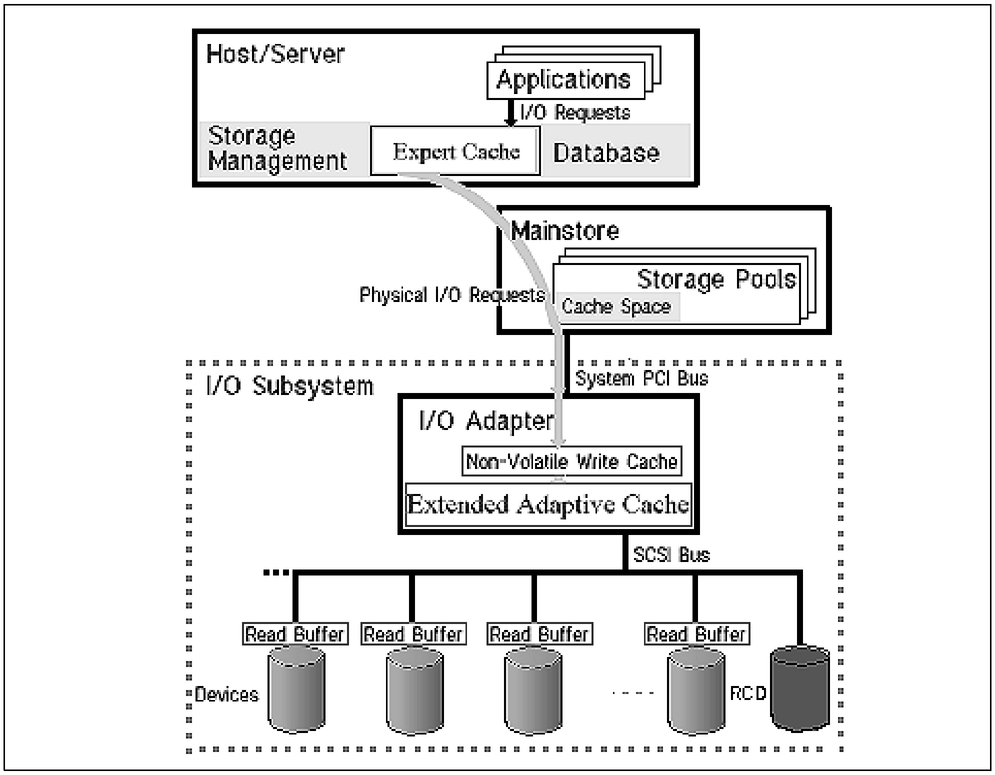


Figure 34. Extended Adaptive Cache architecture

The use of EAC improves the performance of database read operations, as well as all kinds of read operations. This includes read requests generated by other system components, such as the Integrated Netfinity Server, or AS/400 Integrated File System read requests. It also works effectively with a group of disks that have RAID-5 protection or mirroring active. However, EAC c*annot* be activated in a #2748 disk controller that has the *disk compression* function active, as the memory used for disk compression is the same as that used for the EAC algorithm.

To realize the best performance improvement, use EAC with the group of disk units which are most active and performance-critical in nature. To obtain the full benefit of EAC, all disk units in this group should also be attached to the same #2748 disk controller as the RCD.

A Commercial Processing Workload for EAC benchmark was conducted on the AS/400 4-way processor system model 620 with EAC. The results are shown in the following figures. Notice that EAC complements Expert Cache to improve response time.

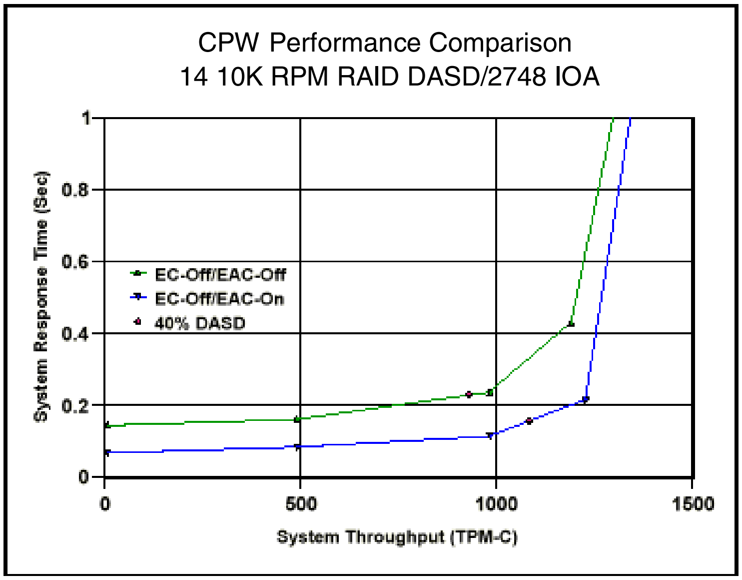


Figure 35. CPW response time: EAC without Expert Cache

EAC extends system throughput and positively affects response time.

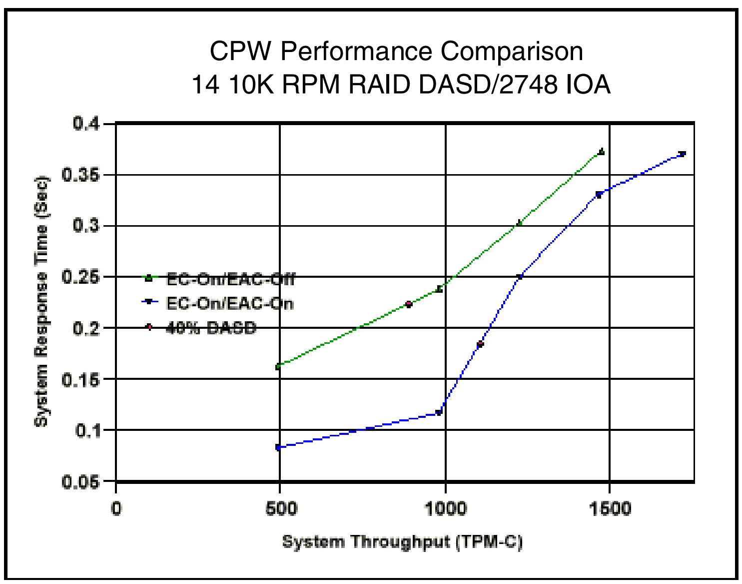


Figure 36. CPW response time: EAC with Expert Cache

With EAC, the response time remains below 0.4 second when the system throughput exceeds 1500 transactions per minute. This result indicates that EAC works well with Expert Cache.

4.2 Extended Adaptive Cache Simulator

The benefit of EAC depends greatly on the system workload characteristics and the system configuration. There is no standard justification to indicate how much you can gain from implementing EAC.

The overall effectiveness of the EAC is best understood using the EAC Simulator. The EAC Simulator is a function of the #2748 disk controller that collects disk I/O statistics for an analysis by the Performance Tool/400 program product (5769-PT1). You can generate a Performance Tool/400 report from the simulator data to help estimate the performance improvement you can potentially gain from activating EAC.

The collected I/O statistics come from the real I/O requests generated from the actual workload in your AS/400 system. The performance report is directly relevant to your system setting. You should run the Simulator for at least three hours under a heavily utilized system workload to get the best possible estimation.

The EAC Simulator operates in the #2748 disk controller itself and uses the same algorithms that manage the EAC. It is activated through AS/400 Management Central V4R4 Collection Services.

Note

For instructions on how to run the EAC Simulator and generate a report, visit the Web site `http://www.as400.ibm.com/infocenter`. Select **V4R4**, and click the **GO** button. Then look under **System Administration and Maintenance** for a topic of **Estimating benefits of Extended Adaptive Cache**. A PDF file of the instruction can also be downloaded from this Web site.

Be sure to apply service pack SF58121 (or later) to Client Access Express code before starting the Collection Services. Otherwise, it collects performance data without EAC statistics.

Emulated performance results are shown for an actual workload over time on a *per disk* basis. The generated report also gives an estimated percent of response time improvement for each disk unit.

The EAC Simulator can also be activated by the AS/400 Performance Monitor, which is a component of Performance Tools/400 product.

Activation of the EAC Simulator does not itself improve your system performance. The simulator gathers statistical information for a prediction of performance improvement that the EAC could offer if EAC is activated.

The EAC Simulator performance results are reported on a per disk unit basis within the Disk Activity section of the Component Report of Performance Tools/400. The following two fields are updated when the EAC Simulator is active:

- **EACS Read** (EAC Simulator Percent Read Hits):

  This is the percentage of the total reads that are directed to disk that would have instead been satisfied by the RCD.

- **EACS Resp** (EAC Simulator estimated percent response time improvement):

  This is an estimation of how much faster the device would run with the RCD installed. This is an approximation, since the Simulator makes basic assumptions about performance parameters that are in place when EAC is actually active. This percentage value represents the amount of reduction you can expect in the response time that is reported in the Performance Monitor

System Report for that disk unit. This projection is valid only when the System Report evaluates performance data that matches the characteristics of the performance data on which this Component Report is based.

Refer to 4.4.1, “A sample performance effect of the Read Cache Device” on page 64, for more details.

4.3 Read Cache Device

The Read Cache Device is a volatile device used to temporarily cache data for the purpose of improving performance. A maximum of one Read Cache Device attaches to the #2748 PCI RAID disk controller, and a maximum of three attach per #5065 Storage/PCI Expansion Tower.

The read cache device product feature codes are:

- #4331 for the new #5065 Storage/PCI Expansion Tower on all AS/400e 6xx, Sxx, and 7xx models, except the 600 and the S10
- #6831 for Model 170s, 600, 620, S10, S20, and 720 system units, or Model 620, S20, and 720 system unit expansions (#5064/#9364 with #9330 PCI Integrated Expansion Unit)

Extended Adaptive Cache cannot be used with compression enabled on the same #2748 PCI RAID Disk Unit Controller IOA. The #2748 IOA is shipped with compression disabled. Compression is enabled by changing a jumper position on the IOA. In order for this change to take effect, the system must be IPLed. Refer to Figure 37 on page 64 for a view of the #2748 disk controller jumper setting.

For information regarding jumpers and switches, refer to the following Web site:
`http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/INFOCENT.HTM`

Click on each of the following links: **System Administration and Maintenance-> System maintenance->Storage I/O card modes and jumpers**. The same information is provided on your AS/400 Information Center CD set, SK3T-2027. To retrieve this information, click on these same links.

Refer to Table 3 for configuration and physical placement rules for the Read Cache Device.

Table 3. Read Cache Device configuration rules

Feature code (FC)	Description/Rule
#4331	Allowed only in slots D01, D06, D11, D16, D21, D26 D31, D36, D46 of FC #5065 PCI Expansion Tower Uses one DASD slot Requires FC #2748 1.6 GB capacity
#6831	Not allowed in slot L01 of system unit Uses one DASD slot Requires FC #2748 1.6 GB capacity

4.4 The 1.6 GB Read Cache Device

When you decide to use EAC to improve the performance of your application, order the Read Cache Device (RCD), and attach it to the #2748 disk controller.

The RCD is mounted into a disk slot connected through a SCSI bus to the controller. The installation of the RCD itself activates EAC. The RCD can be added into an empty disk slot (or removed), without bringing your AS/400 system down, by using the disk concurrent maintenance feature of OS/400.

The cache function covers only the disk units attached under the same #2748 disk controller. Depending on your performance requirements, you may need multiple RCDs when multiple #2748 disk controllers exist in the same system.

Two orderable product feature codes exist for the RCD to be used in your AS/400 system:

- #4331 1.6 GB Read Cache Device installs into the #5065 Storage/PCI Expansion Tower on all AS/400e 6XX, SXX, and 7XX models (except the 600 and S10).
- #6831 1.6 GB Read Cache Device installs into model 170, 600, 620, S10, S20, and 720 system units, or model 620, S20, and 720 System Unit Expansions (#5064 or #9364 with #9330 PCI Integrated Expansion Unit).

The #2748 disk controller works in either one of the following modes:

- **Compression Mode**: This mode should only be used when you wish to run Integrated Hardware Disk Compression. In compression mode, the write cache size available in the controller is 4 MB.
- **Enhanced Mode**: This mode uses an enhanced write cache size of 26 MB. Enhanced mode also provides support for the EAC through an attachment of a RCD. Without the RCD attached, this mode enables the EAC Simulator.

Enhanced mode is the default setup. No action is required beyond installing the RCD if the card was never switched to compression mode in the past.

If the #2748 disk controller has been switched to compression mode in the past, set the operation mode of the #2748 disk controller to Enhanced Mode when installing the RCD in your AS/400 system. Operation mode is determined by the position of a jumper switch located at the back of the controller adapter. Position **C** is for Compression mode, and position **E** is for Enhanced mode. Figure 37 shows a picture of the jumper switch setting.

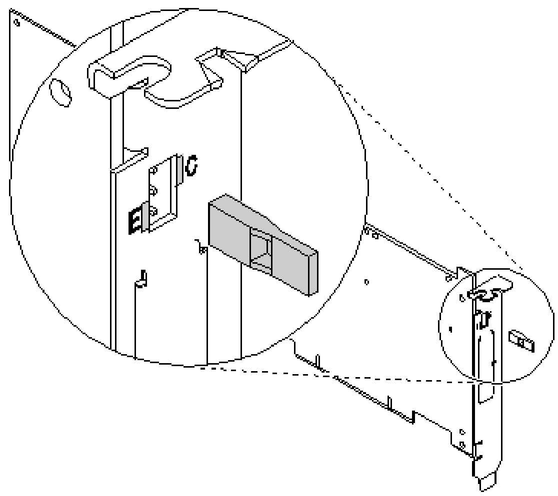


Figure 37. The #2748 disk controller: Jumper setting

A change of the jumper switch position can be done at any time. The operation mode takes effect only after a subsequent system IPL is complete.

Note

If you want to change from Compression to Enhanced mode, be sure to decompress any existing compressed disk units first.

4.4.1 A sample performance effect of the Read Cache Device

For the writing of this redbook, we tested the performance benefit of the RCD by using AS/400 Device Exerciser Plus (DEXPLUS) tools. DEXPLUS is primarily used to generate a set of I/O requests against any device in the AS/400 system. We used DEXPLUS to generate read I/O requests to all disk units in ASP 1.

We used the DEXPLUS tool on an AS/400 Server model 170, CPU feature #2385. There were eight 10,000 (10 K) RPM 8.5 GB disk units in ASP01. All of the 10 K drives were attached to one #2748 disk controller. RAID-5 was active.

We created 400 write-once-read-many disk class (*DSK) jobs that *randomly* read 2 KB data records from a 3.3 GB physical file. The Device Exerciser jobs ran in the QDEX subsystem, to which we assigned a dedicated main storage pool of 300 MB and set the maximum number of active jobs (Max Active) to 400. Each DEXPLUS job issued one read request every second. Expert Cache was turned off in the main storage pool of the QDEX subsystem.

Reminder

Expert cache is controlled by setting the Paging Option value of the subsystem pool to *FIXED (no expert cache) or *CALC (use expert cache). Use either the Work with System Status (WRKSYSSTS) command or the Change Shared Storage Pool (CHGSHRPOOL) command to change the Paging Option parameter.

We collected performance data and printed System Report-Disk Utilization and Component Report-Disk Activity reports to compare the results of running DEXPLUS with and without the RCD.

The System Report-Disk Utilization and Component Report-Disk Activity of the DEXPLUS workload *without* the RCD in the server and without the Expert Cache option activated are shown in Figure 38 and Figure 39 on page 66 respectively.

```
                                                      System Report                                   10/18/99  8:02:08

                                                     Disk Utilization                                        Page 0001
                                              EAC Simulator test NO RCD  NO XCache

Member  . . . : Q291040003 Model/Serial  . : 170/XX-XXXXX    Main storage . . : 1536.0 M   Started  . . . . : 10/18/99 04:00:02
  Library . . : EACPFR     System name . . : XXXXXXXX       Version/Release  :   4/ 4.0   Stopped  . . . . : 10/18/99 07:54:59
Partition ID  : 00          Feature Code  . : 2385

       Unit             Size   IOP   IOP          Dsk CPU  ASP   --Percent--   Op Per    K Per    - Average Time Per I/O --
 Unit  Name      Type  (M)    Util  Name          Util    ID    Full  Util    Second     I/O     Service  Wait   Response
 ----   ---------- ---- ------- ----   ------------  -------  ---   ----   ----  ---------  ---------  -------  ------
--------
 0001  DD010     6717  7,516   7.1  CMB01           7.3    01   33.5   12.2     22.85       4.0    .0053  .0035     .0088
 0002  DD008     6717  7,516   7.1  CMB01           7.3    01   33.4   12.7     23.26       4.0    .0054  .0035     .0089
 0003  DD009     6717  7,516   7.1  CMB01           7.3    01   33.5   12.8     23.08       4.0    .0055  .0037     .0092
 0004  DD006     6717  7,516   7.1  CMB01           7.3    01   33.5   12.9     23.13       4.0    .0055  .0034     .0089
 0006  DD005     6717  7,516   7.1  CMB01           7.3    01   33.4   12.4     23.36       4.0    .0053  .0033     .0086
 0008  DD007     6717  7,516   7.1  CMB01           7.3    01   33.4   12.5     23.34       4.0    .0053  .0037     .0090
 0009  DD002     6717  7,516   7.1  CMB01           7.3    01   33.5   12.9     23.63       4.0    .0054  .0037     .0091
 0010  DD004     6717  7,516   7.1  CMB01           7.3    01   33.4   12.2     23.03       4.0    .0052  .0037     .0089

Total                  68,717
Average                                                         29.3   11.2     20.63       4.0    .0054  .0035     .0089


Unit                   --  Disk arm identifier
Unit Name              --  Disk arm resource name
Type                   --  Type of disk
Size (M)               --  Disk space capacity in millions of bytes
IOP Util               --  Percentage of utilization for each Input/Output Processor
IOP Name               --  Input/Output Processor resource name
Dsk CPU Util           --  Percentage of Disk Processor Utilization
ASP ID                 --  Auxiliary Storage Pool ID
Percent Full           --  Percentage of disk space capacity in use
HPercent Util           --  Average disk operation utilization (busy)
Op per Second          --  Average number of disk operations per second
K Per I/O              --  Average number of kilobytes (1024) transferred per disk operation
Average Service Time   --  Average disk service time per I/O operation
Average Wait Time      --  Average disk wait time per I/O operation
Average Response Time  --  Average disk response time per I/O operation
```

Figure 38. System Report without RCD or Expert Cache

Component Report 10/18/99 8:02:43

Disk Activity Page 1

EAC Simulator test NO RCD NO XCache

Member . . . : Q291040003 Model/Serial . : 170/XX-XXXXX Main storage . . : 1536.0 M Started . . . . : 10/18/99 04:00:02
Library . . : EACPFR System name . . :XXXXXXXX Version/Release : 4/ 4.0 Stopped . . . . : 10/18/99 07:54:59
Partition ID : 00 Feature Code . :2385

			------------- **Average Disk Activity Per Hour** -------------						------ **Cache hit Statistics** -------					
		Srv	----------------- Disk Arm Seek Distance -----------------						Device	**Controller**	Write	**EACS**	**EACS**	-Disk Capacity	
Unit	**Util**	Time	0	1/12	1/6	1/3	2/3	>2/3	Read	**Read**	Effic	**Read**	**Resp**	MB	Percent
0001	**12.2**	.0053	**3,826**	**72,410**	541	51	221	118	6.2	**.0**	45.2	**80.1**	**64.7**	4,997	66.4
0002	**12.7**	.0054	**4,073**	**73,197**	1,378	84	249	113	5.3	**.1**	54.5	**80.1**	**65.0**	4,998	66.4
0003	**12.8**	.0055	**3,868**	**72,623**	1,584	119	267	106	5.6	**.0**	36.4	**79.9**	**65.1**	4,998	66.4
0004	**12.9**	.0055	**3,518**	**74,085**	926	184	307	105	5.4	**.0**	19.3	**79.7**	**65.1**	4,998	66.4
0006	**12.4**	.0053	**4,229**	**72,596**	1,026	99	261	111	5.8	**.4**	71.3	**79.6**	**63.5**	4,998	66.4
0008	**12.5**	.0053	**4,380**	**73,110**	1,099	103	133	67	5.6	**.1**	65.9	**79.4**	**63.8**	4,998	66.4
0009	**12.9**	.0054	**3,870**	**73,300**	1,370	131	313	120	5.5	**.5**	76.6	**79.4**	**63.6**	4,997	66.4
0010	**12.2**	.0052	**3,889**	**72,492**	1,004	26	386	153	5.8	**.1**	52.7	**80.3**	**64.5**	4,998	66.4

Column	Total	Average	
Util		11.2	
Srv Time		.0054	
Disk Arm Seek Distance			
0	**31,655**		
1/12	**583,817**		
1/6	8,932		
1/3	800		
2/3	2,141		
>2/3	898		
Cache hit Statistics			
Device Read		5.7	
Controller Read		**.1**	**<---- Cache hit rate, including RCD when present.**
Write Efficiency		60.0	
EACS Read		**79.8**	**<---- An estimation of RCD hit rate if present.**
EACS Resp		**64.4**	**<---- An estimation of disk I/O response time improvement if RCD present.**
Disk Capacity			
MB	48,567		
Percent	70.6		

Unit	-- Disk arm identifier
Util	-- Drive utilization
Srv Time	-- Average service time per request in seconds
Disk Arm Seek Distance	-- Average seek distance distributions per hour
0	-- Number of zero seeks
1/12	-- Number of seeks between 0 and 1/12 of the disk
1/6	-- Number of seeks between 1/12 and 1/6 of the disk
1/3	-- Number of seeks between 1/6 and 1/3 of the disk
2/3	-- Number of seeks between 1/3 and 2/3 of the disk
>2/3	-- Number of seeks greater than 2/ 3 of the disk
Cache hit Stastistics	--
Device Read	-- Percent of device read hits for each arm
Controller Read	-- Percent of controller cache read hits for each arm
Write Efficiency	-- Percent of efficiency of write cache
EACS Read	-- Extended Adaptive Cache Simulator percent read hits
EACS Resp	-- Extended Adaptive Cache Simulator estimated percent response time improvement
Disk Capacity	-- Average amount of disk space used or available
MB	-- Millions of bytes available on the disk
Percent	-- Percent of space available on the disk

Figure 39. Component Report without RCD or Expert Cache

Figure 40 and Figure 41 show the System Report-Disk Utilization and Component Report-Disk Activity of the DEXPLUS workload *with* the RCD in the server but without the Expert Cache option activated.

System Report 10/18/99 11:24:52

Disk Utilization Page 0001
RCD test NO Xcache

Member . . . : T5WITHRCD Model/Serial . : 170/XX-XXXXX Main storage . . : 1536.0 M Started . . . . : 10/18/99 08:21:34
Library . . : EACPFR System name . . : XXXXXXXX Version/Release : 4/ 4.0 Stopped . . . . : 10/18/99 11:20:00
Partition ID : 00 Feature Code . : 2385

Unit	Unit Name	Type	Size (M)	IOP Util	IOP Name	Dsk CPU Util	ASP ID	--Percent-- Full	Util	Op Per Second	K Per I/O	- **Average Time Per I/O** -- Service	**Wait**	**Response**
0001	DD010	6717	7,516	7.2	CMB01	7.4	01	33.6	5.4	23.07	4.0	.0023	**.0012**	**.0035**
0002	DD008	6717	7,516	7.2	CMB01	7.4	01	33.6	6.0	23.29	4.0	.0025	**.0014**	**.0039**
0003	DD009	6717	7,516	7.2	CMB01	7.4	01	33.6	6.0	23.36	4.0	.0025	**.0016**	**.0041**
0004	DD006	6717	7,516	7.2	CMB01	7.4	01	33.6	5.9	23.22	4.0	.0025	**.0015**	**.0040**
0006	DD005	6717	7,516	7.2	CMB01	7.4	01	33.6	5.7	23.38	4.0	.0024	**.0011**	**.0035**
0008	DD007	6717	7,516	7.2	CMB01	7.4	01	33.6	5.8	23.31	4.0	.0024	**.0015**	**.0039**
0009	DD002	6717	7,516	7.2	CMB01	7.4	01	33.6	5.6	23.63	4.0	.0023	**.0012**	**.0035**
0010	DD004	6717	7,516	7.2	CMB01	7.4	01	33.6	5.6	23.10	4.0	.0024	**.0014**	**.0038**
Total			68,717											
Average								29.4	5.1	20.71	4.0	.0024	**.0014**	**.0038**

Figure 40. Workload without Expert Cache and with Read Adaptive Device Utilization

Component Report 10/18/99 11:25:02

Disk Activity Page 1
RCD test NO Xcache

Member . . . : T5WITHRCD Model/Serial . : 170/XX-XXXXX Main storage . . : 1536.0 M Started . . . . : 10/18/99 08:21:34
Library . . : EACPFR System name . . :XXXXXXXX Version/Release : 4/ 4.0 Stopped . . . . : 10/18/99 11:20:00
Partition ID : 00 Feature Code . :2385

Unit	**Util**	Srv Time	**Average Disk Activity Per Hour** / Disk Arm Seek Distance 0	1/12	1/6	1/3	2/3	>2/3	**Cache hit Statistics** Device Read	**Controller Read**	Write Effic	EACS Read	EACS Resp	-Disk Capacity MB	Percent
0001	**5.4**	.0023	**1,153**	**8,071**	66	18	170	91	44.0	**80.2**	53.8	.0	.0	4,988	66.3
0002	**6.0**	.0025	**1,338**	**8,361**	220	26	250	147	43.0	**79.8**	18.2	.0	.0	4,988	66.3
0003	**6.0**	.0025	**1,436**	**8,463**	259	45	183	117	41.0	**79.9**	59.8	.0	.0	4,988	66.3
0004	**5.9**	.0025	**1,210**	**8,467**	262	71	207	136	43.3	**79.4**	19.4	.0	.0	4,988	66.3
0006	**5.7**	.0024	**1,467**	**8,579**	232	65	182	160	40.7	**79.7**	61.9	.0	.0	4,988	66.3
0008	**5.8**	.0024	**1,320**	**8,960**	216	60	209	140	39.3	**79.8**	36.7	.0	.0	4,988	66.3
0009	**5.6**	.0023	**1,273**	**8,760**	166	58	164	162	41.5	**79.8**	70.2	.0	.0	4,987	66.3
0010	**5.6**	.0024	**1,269**	**7,962**	150	31	226	170	45.0	**79.8**	34.0	.0	.0	4,988	66.3

Column	Total	Average
Util		5.1
Srv Time		.0024
Disk Arm Seek Distance		
0	**10,470**	
1/12	**67,627**	
1/6	1,576	
1/3	377	
2/3	1,594	
>2/3	1,127	
Cache hit Statistics		
Device Read		42.2
Controller Read		**79.8**
Write Efficiency		50.5
EACS Read		.0
EACS Resp		.0
Disk Capacity		
MB	48,488	
Percent	70.5	

Figure 41. Workload without Expert Cache and with Read Adaptive Device Disk Activity

Comparing the reports to determine what effects the RCD had on our 170 server, you can observe the following improvements:

- The average response time per disk I/O is reduced from 0.0089 second to 0.0038 second. This represents a 57% reduction.
- The average wait time per disk I/O is reduced from 0.0035 second to 0.0014 second, representing a 60% reduction.
- The average number of disk activities per hour is significantly reduced from 583,817 activities per hour to 67,627 activities per hour for the seek distance of 1/12. This is due to the fact that most data is available in the RCD. Therefore, fewer read I/O requests go to the disk units. Improvements also apply to the other seek distances.
- The RCD hit rate is very close to the EAC Simulator estimation of 79.8%.
- The average improvement of response time per disk I/O is 57.3%. This is fairly close to the EAC Simulator estimation of 64.4%.
- The average disk unit utilization reduced from 12% to 6%, representing a 50% decrease.

The performance results are summarized in Table 4 on page 70.

Performance comparison of EAC and RCD

We then explored how much improvement the Expert Cache had on the system performance compared to the RCD. We did this by assigning a main storage pool of 1 GB to the QDEX subsystem (up from 300 MB in the runs) where RCD is installed. The System Report-Disk Utilization and Component Report-Disk Activity of the DEXPLUS workload *without* the RCD, but with Expert Cache active, are shown in Figure 42 and Figure 43 on page 69 respectively.

System Report 10/18/99 16:55:43

Disk Utilization Page 0001

EAC Simulator test with Expert cache NO RCD

```
Member . . . : T5NRCDWXC  Model/Serial  . : 170/XX-XXXXX   Main storage . . : 1536.0 M   Started  . . . . : 10/18/99 14:11:29
  Library . . : EACPFR     System name . . : XXXXXXXX       Version/Release  :   4/ 4.0   Stopped  . . . . : 10/18/99 16:39:59
Partition ID  : 00         Feature Code  . : 2385
```

Unit	Unit Name	Type	Size (M)	IOP Util	IOP Name	Dsk CPU Util	ASP ID	--Percent-- Full	Util	Op Per Second	K Per I/O	- Average Time Per I/O -- Service	Wait	Response
0001	DD010	6717	7,516	7.4	CMB01	9.1	01	32.7	9.6	22.21	4.9	.0043	**.0028**	**.0071**
0002	DD008	6717	7,516	7.4	CMB01	9.1	01	32.7	11.2	24.91	4.5	.0044	**.0024**	**.0068**
0003	DD009	6717	7,516	7.4	CMB01	9.1	01	32.7	11.3	24.84	4.4	.0045	**.0026**	**.0071**
0004	DD006	6717	7,516	7.4	CMB01	9.1	01	32.7	11.4	25.11	4.5	.0045	**.0026**	**.0071**
0006	DD005	6717	7,516	7.4	CMB01	9.1	01	32.7	11.2	25.17	4.5	.0044	**.0026**	**.0070**
0008	DD007	6717	7,516	7.4	CMB01	9.1	01	32.7	11.3	25.00	4.4	.0045	**.0025**	**.0070**
0009	DD002	6717	7,516	7.4	CMB01	9.1	01	32.7	10.9	24.88	4.5	.0043	**.0024**	**.0067**
0010	DD004	6717	7,516	7.4	CMB01	9.1	01	32.7	10.8	24.56	4.5	.0043	**.0027**	**.0070**
Total			68,717											
Average								28.6	9.7	21.85	4.5	.0044	**.0026**	**.0070**

Figure 42. Workload with Expert Cache and without Read Adaptive Device Utilization

Component Report 10/18/99 16:56:04

Disk Activity Page 1
EAC Simulator test with Expert cache NO RCD

Member . . . : T5NRCDWXC Model/Serial . : 170/XX-XXXXX Main storage . . : 1536.0 M Started . . . . : 10/18/99 14:11:29
Library . . : EACPFR System name . . :XXXXXXXX Version/Release : 4/ 4.0 Stopped . . . . : 10/18/99 16:39:59
Partition ID : 00 Feature Code . :2385

			Average Disk Activity Per Hour						Cache hit Statistics						
		Srv	Disk Arm Seek Distance						Device	Controller	Write	**EACS**	**EACS**	-Disk Capacity	
Unit	**Util**	Time	0	1/12	1/6	1/3	2/3	>2/3	Read	Read	Effic	**Read**	**Resp**	MB	Percent
0001	**9.6**	.0043	9,928	47,156	1,305	19	2,544	1,616	8.3	.2	75.6	**78.8**	**42.8**	5,053	67.2
0002	**11.2**	.0044	12,485	54,254	1,148	3	2,266	1,915	8.7	.0	74.8	**81.3**	**47.4**	5,053	67.2
0003	**11.3**	.0045	13,578	53,885	1,214	87	2,187	1,916	7.7	.0	74.6	**81.3**	**47.7**	5,053	67.2
0004	**11.4**	.0045	12,638	55,112	1,151	16	2,350	1,917	8.7	.1	74.0	**81.2**	**47.7**	5,053	67.2
0006	**11.2**	.0044	13,117	54,077	880	26	2,306	2,033	9.0	.1	74.7	**81.6**	**47.1**	5,053	67.2
0008	**11.3**	.0045	12,789	54,639	1,054	21	2,352	1,913	8.5	.3	74.3	**81.0**	**47.6**	5,053	67.2
0009	**10.9**	.0043	12,057	53,053	1,104	19	2,248	1,934	9.5	.1	75.9	**81.1**	**46.1**	5,053	67.2
0010	**10.8**	.0043	12,780	53,321	1,049	14	2,433	1,830	8.6	.0	74.2	**81.4**	**47.2**	5,053	67.2

Column	Total	Average
Util		9.7
Srv Time		.0044
Disk Arm Seek Distance		
0	**99,374**	
1/12	**425,502**	
1/6	8,909	
1/3	209	
2/3	18,689	
>2/3	15,078	
Cache hit Statistics		
Device Read		8.6
Controller Read		.1
Write Efficiency		74.8
EACS Read		**81.0**
EACS Resp		**46.8**
Disk Capacity		
MB	49,009	
Percent	71.3	

Figure 43. Workload with Expert Cache and without Read Adaptive Device Disk Activity

Comparing the reports to determine what effects the Expert Cache (without the RCD installed) had on the 170 server, observe the following improvements:

- The average response time per disk I/O reduced from 0.0089 second to 0.0070 second, representing a reduction of 21% (0.0038 second with the RCD).
- The average wait time per disk I/O reduced from 0.0035 second to 0.0026 second. The average wait time was measured at 25% (0.0014 second with the RCD).
- The average number of disk activities per hour is slightly reduced from 583,817 activities per hour to 425,502 activities per hour for the seek distance of 1/12. This represents a 27% reduction. Compare this to 67,627 activities per hour with the RCD.
- The average disk unit utilization reduced from 12% to 11% (compared to 6% with the RCD).
- The EAC Simulator estimates a substantial 46% disk I/O response time improvement to such situation if the RCD is installed.

The performance results are summarized in Table 4.

Table 4. Read Cache Device test: Performance results

	Without RCD and without Expert Cache (300 MB subsystem pool)	With RCD but without Expert Cache (300 MB subsystem pool)	With Expert Cache but without RCD (1 GB subsystem pool)
Average disk I/O response time	0.0089 second	0.0038 second	0.0070 second
Average disk unit utilization	12%	6%	11%
Average disk activities per hour for 1/12 distance seek	583,817	67,627	425,502
Estimated disk I/O response time improvement with RCD	64%	--	46%

In summary, Expert Cache brings slight improvements to the DEXPLUS random-read intensive workload, compared to the RCD. This is because expert cache is designed to work better with sequential-read workloads than random-read workloads. The RCD can bring throughput and response time improvement to a system with a high volume of read-intensive workloads.

Visit the Web page `http://www.as400.ibm.com/beyondtech/ext_adapt_cache.htm` for further information on EAC.

Part 2. Tools and tips for AS/400 DASD storage management

Part 2 contains topics relative to the organization and management of AS/400 disk storage. File system architectures, tools to assess, monitor and manage disk units, including a process to address expected and unexpected, sudden and gradual growth of information on disk, is provided. Sample code associated with the WRKDASD tool written for this redbook is found in Appendix D, “WRKDASD sample code” on page 201.

Chapter 5. Extending the Integrated File System with UDFS

The OS/400 User-Defined File System (UDFS) is a storage mechanism that works in the same way as the Integrated File System. The UDFS provides a hierarchical directory structure, is optimized for stream file I/O, and supports case-sensitive names, links, and sockets.

While an Integrated File System directory can be created only in the system ASP, a UDFS and its directory structure can be created in any available ASP. This makes it possible for Integrated File System users to make use of the Hierarchical Storage Management (HSM) type of ASP balance function in OS/400 V4R4 to manage the Integrated File System stream files, because HSM balance works only in user ASPs. See 7.2, “The disk balance tool” on page 99, for information on ASP balance functions.

In effect, the UDFS was created in OS/400 V3R7 as an extension to the original Integrated File System introduced in OS/400 V3R1. If you are familiar with UNIX, the UDFS is similar to what is called the Mounted File System concept.

A UDFS object must always be created in the directory path of \DEV\QASPnn, where “nn” is the ASP number in which UDFS data is stored. Once created, a UDFS object *must be mounted* onto an existing Integrated File System directory structure for use. When mounted, the UDFS assumes the target directory path name. Use this name to access its data.

Use a UDFS in the same manner that you use an Integrated File System directory. *Unmount* it when you no longer need access to its data. With the file system unmounted, the data is inaccessible from the Integrated File System name space. The data remains intact in the UDFS object itself.

You can save and restore UDFS data by OS/400 SAV and RST commands in the same manner as you do with other Integrated File System components. Refer to *OS/400 Integrated File System Introduction*, SC41-5711, for more information on UDFS.

5.1 Creating and mounting a UDFS

This section describes a sample procedure to create a UDFS in ASP02 and mount it for use. Two views are presented: Operations Navigator (as the main approach) and a Command Language (CL) equivalent (for those who use a 5250 session).

Note: In a Windows 95/98 environment, an Integrated File System directory is sometimes addressed (that is, referred to) as a *folder*.

1. To create a UDFS on your AS/400 system, open Operations Navigator and expand the **File Systems -> Integrated File System -> Root -> dev** lines, as shown in Figure 44 on page 74.

© Copyright IBM Corp. 2000

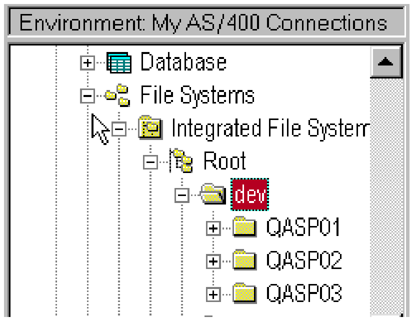


Figure 44. Expanding Integrated File System group

Under the dev folder, QASPnn folders correspond to each ASP that exists on the system. The QASPnn directories are automatically created as ASPs when added to the system. There are three ASPs in our example: ASP01, ASP02, and ASP03.

2. To create a new UDFS object in *ASP02,* right-click **QASP02** to invoke a pop-up menu. Then select **New UDFS**, as shown in Figure 45.

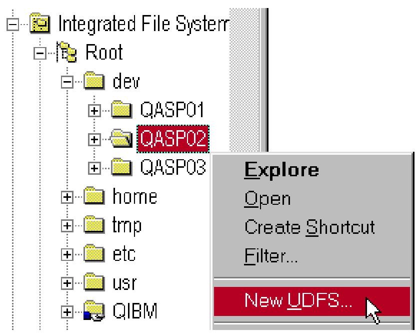


Figure 45. Creating a new User-defined File System in ASP 2

3. In the New User-Defined File System window, as shown in Figure 46, enter the name of the new UDFS and its description.

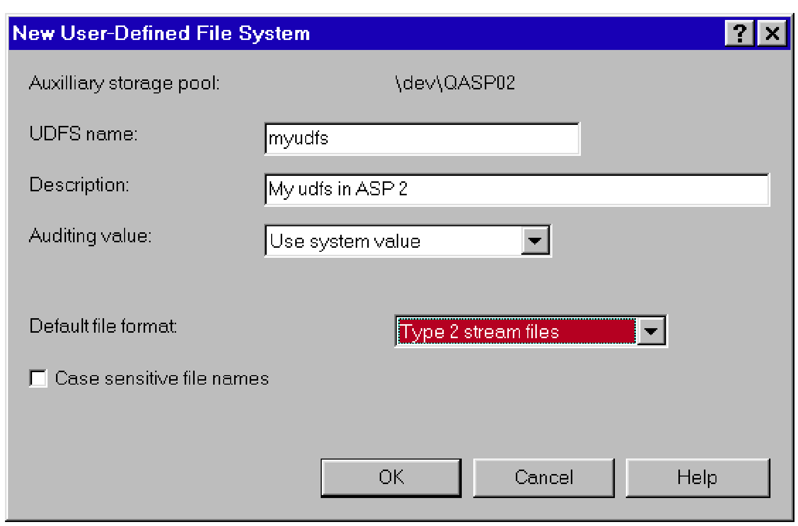


Figure 46. Specifying UDFS details

If you want all files in the UDFS to be addressed uniquely by case-sensitive names, check the **Case sensitive file names** box. For example, AbCfile and abcfile are different files if case-sensitivity is enabled. Click **OK** when the prompts are completed.

Once created, the newly created UDFS object appears on the right side of the Operations Navigator window.

A CL command equivalent to this step uses the Create UDFS (CRTUDFS) command from the OS/400 command line, as shown in Figure 47. CRTUDFS is shown with its parameters prompted.

```
                         Create User-Defined FS (CRTUDFS)

Type choices, press Enter.

User-defined file system . . . . UDFS          > '\dev\qasp02\myudfs.udfs'

Public authority for data  . . . DTAAUT          *INDIR
Public authority for object  . . OBJAUT          *INDIR
                          + for more values
Auditing value for objects . . . CRTOBJAUD       *SYSVAL

                           Additional Parameters

Case sensitivity . . . . . . . . CASE            *MONO
Default file format  . . . . . . DFTFILEFMT      *TYPE2
Text 'description' . . . . . . . TEXT          > 'My UDFS in ASP 2'
```

Figure 47. CRTUDFS command

4. To create an Integrated File System directory onto which the UDFS object will be mounted, move the mouse pointer to the **Root** file system, right-click it, and select **New Folder**. Specify `\myifsdir` as the name of the new Integrated File System directory. Click **OK** to create a new Integrated File System directory named myifsdir. This directory is used to map the file system in the user ASP to the Integrated File System root file system.

   A CL command equivalent to this step is:

   ```
   MD '\myifsdir'
   ```

   Note: Alternative CL command names to MD are MKDIR and CRTDIR.

   Now mount the UDFS to the Integrated File System directory just created.

5. Move the mouse pointer to the newly created UDFS object (myudfs.UDFS), and right-click it. Select **Mount** in the pop-up menu that appears, as shown in Figure 48 on page 76.

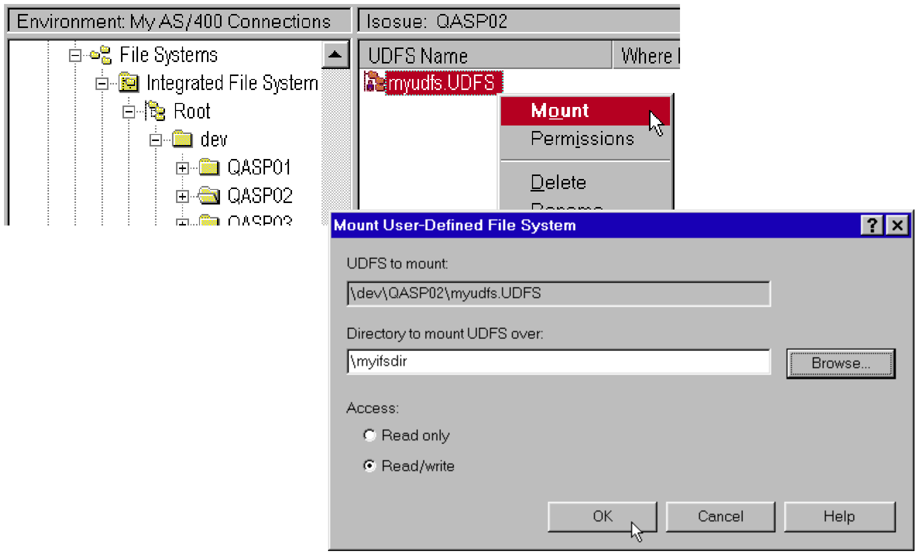


Figure 48. Mounting a UDFS onto an Integrated File System directory

In the Mount User-Defined File System window that appears, specify the Integrated File System directory path on which mount the UDFS (it is \myifsdir in our example). Alternatively, click the **Browse** button to bring up a graphical directory tree of the system and select the Integrated File System directory of your choice. Then check the access type as read only or read/write. Click **OK** to finish.

A CL command equivalent to this step uses the MOUNT command from the OS/400 command line, as shown in Figure 49, with the MOUNT command parameters prompted.

```
                         Add Mounted FS (MOUNT)

Type choices, press Enter.

Type of file system  . . . . . . TYPE         > *UDFS
File system to mount . . . . . . MFS          > '\dev\qasp02\myudfs.udfs'

Directory to mount over  . . . . MNTOVRDIR    > '\myifsdir'

Mount options  . . . . . . . . . OPTIONS        'rw'

Code page:                       CODEPAGE
  Data file code page  . . . . .                *BINARY
  Path name code page  . . . . .                *ASCII
```

Figure 49. MOUNT command

Now the myudfs UDFS has the Where Mounted field filled with \myifsdir as its mounted position. When the UDFS is unmounted, the Where Mounted field is empty.

6. Move the mouse pointer to **myudfs.UDFS** object, right-click it, and select **Properties** to display various properties of the UDFS object as shown in Figure 50.

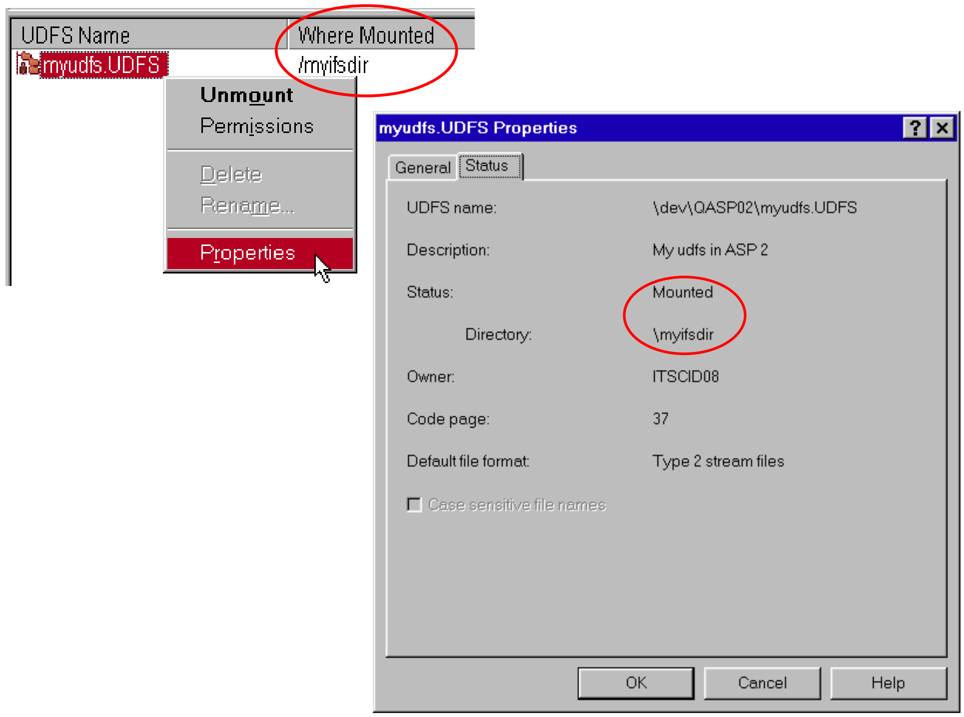


Figure 50. Displaying UDFS properties

A CL command equivalent to this step uses the Display User-Defined FS command, for example: `DSPUDFSD '\dev\qasp02\myudfs.udfs'` entered on the OS/400 command line.

As long as the UDFS object is mounted, any stream files put into the Integrated File System directory \myifsdir are stored in the ASP02 disk space of the \dev\qasp02\myudfs.udfs object.

Unmount and delete the UDFS in the same manner that you create and mount it.

In a 5250 session, the GO CMDUDFS menu provides access to help you with available commands for UDFS manipulation, as shown in Figure 51.

```
CMDUDFS                    User-Defined FS Commands

Select one of the following:

  Commands
     1. Create User-Defined FS                                  CRTUDFS
     2. Delete User-Defined FS                                  DLTUDFS
     3. Display User-Defined FS                                 DSPUDFS

  Related Command Menus
     4. Mounted File System Commands                            CMDMFS
     5. Network File System Commands                            CMDNFS
```

Figure 51. CMDUDFS menu

5.2 UDFS considerations

To make good use of UDFS, you should understand these points:

- You choose to mount the UDFS object on an *existing* Integrated File System directory. However, all current data in that Integrated File System directory, if there is any, is inaccessible as long as the UDFS is mounted onto it. The data does still exist. The Integrated File System data is accessible again after you unmount the UDFS object from the Integrated File System directory.

  In general, create an empty Integrated File System directory just for the mounting of a UDFS.

- The mounting of a UDFS object does not survive an IPL of the machine. So, in case you need a “virtually permanent” mounting of a UDFS object, include the MOUNT command in the IPL start-up program, specified in the system value QSTRUPPGM.

  Be aware of the following considerations to make a mount during the IPL successful:

  – By default, the public authority of the MOUNT command is the *EXCLUDE. Since the OS/400 start-up program named in QSTRUPPGM runs under the QPGMR user profile, add a private authority of *USE for QPGMR to the MOUNT command object. Or use a proper owner or adopted authority when you compile the start-up program.

  – A user profile needs the special authority of *IOSYSCFG to run the MOUNT command successfully. Add special authority (*SPCAUT) to the QPGMR user profile. Or, use a proper owner or adopted authority when you compile the start-up program.

- When you mount a UDFS object to an Integrated File System directory, the target Integrated File System directory must not be in use at that moment. Otherwise, you get an `Object in use` message, and the mount operation ends in failure. The same holds true for the unmount operation.

  An example of an Integrated File System directory being in use is when a PC user maps a network drive to this directory through the AS/400 NetServer function. Use the NetServer windows to end all the sessions that are using the directory to discontinue use.

  Recommendation

  Whenever possible, add the MOUNT command in the start-up program to help ensure the directory is accessible.

- Use the SAV command to save the entire UDFS object. Specify a full directory path name for the save device as shown in Figure 52.

```
                         Save Object (SAV)

Type choices, press Enter.

Device . . . . . . . . . . . . . DEV         > '\qsys.lib\tap01.devd'

                          + for more values

Objects:                         OBJ
  Name . . . . . . . . . . . . .             > '\dev\qasp02\myudfs.udfs'

  Include or omit  . . . . . . .               *INCLUDE
                          + for more values
Directory subtree  . . . . . . . SUBTREE       *ALL
Save active  . . . . . . . . . . SAVACT        *NO
Output . . . . . . . . . . . . . OUTPUT      > *print

Type of output information . . . INFTYPE       *ALL
```

Figure 52. Saving UDFS

Only an unmounted UDFS object can be saved directly. Or you can save from the Integrated File System directory while the UDFS is mounted. However, in this case, the UDFS file system information itself is not saved. This means that although you save the Integrated File System directory and its contents, you lose the identity of the source UDFS object. Therefore, a direct save of UDFS is preferred.

With this information, you are now ready to put the UDFS to its practical use for managing the disk storage of stream files.

For further information on UDFS, refer to the AS/400e Information Center at `http://www.as400.ibm.com/infocenter` or the Information Center CD, SK3T-2027, shipped with your AS/400 system.

Chapter 6. Using the QFILESVR.400 file system

The OS/400 File Server file system (QFILESVR.400) is one of the built-in file systems of the OS/400 Integrated File System structure. It is a standard component of OS/400. It is designed primarily to enable local jobs from the native OS/400 environment to share stream-mode file systems on remote AS/400 servers. Although you can also use QFILESVR.400 from a PC environment, you may not want to do so because there are other, more efficient options available, such as OS/400 NetServer or Network File System (NFS), which let the PC interact directly with the target AS/400 systems. Refer to *The AS/400 NetServer Advantage*, SG24-5196, for more information on NFS.

Applications designed with a capability to access stream files in a distributed environment (in addition to DB2/400 database files) can make use of the QFILESVR.400 file system. Two examples of such native advanced applications are Domino for AS/400 and SAP R/3 for AS/400. The Licensed Program Product numbers are 5769-DM1 and 5769-SAP, respectively.

Another example is the native OS/400 archive and retrieval applications that make use of optical storage. Objects on an optical storage are stored in a stream file format. Optical storage is accessed through the Integrated File System. Use QFILESVR.400 to share a direct-attached optical library on one AS/400 system with multiple AS/400 systems.

Sharing of QFILESVR.400 is supported in both SNA LU6.2 and TCP/IP networks. This is done by creating a *TargetSystemName* directly under the QFILESVR.400 file system. Specify a path name of \QFILESVR.400\TargetSystemName\Directory\Directory\....\Object to access objects on the target system. Each component of the path name can be up to 255 characters long. The entire path name can be up to 16 MB in length.

The target system name can be either one of the following:

- A TCP/IP host name, such as *itsosys1.ibm.com*
- An SNA LU 6.2 name such as *appn.itsosys1*

The target system name is always interpreted as a TCP/IP host name first. If it cannot be resolved to an IP address, it is interpreted as a SNA LU 6.2 name.

An SNA LU 6.2 connection is used if there is an idle session specifically established for use by the LU 6.2 available connection. When establishing an LU 6.2 connection, the QFILESVR.400 file system uses an APPC mode named *BLANK. On the target system, a job named QPWFSERV is submitted to the QSERVER subsystem to serve the connection. The user profile running this job is defined by the communications entry for the *BLANK mode.

For more information about LU 6.2 communications, see *APPC Programming*, SC41-5443. Refer to *OS/400 Integrated File System Introduction*, SC41-5711, for more information on QFILESVR.400.

© Copyright IBM Corp. 2000

6.1 Configuring QFILESVR.400 in a TCP/IP environment

To prepare for QFILESVR.400 use in a TCP/IP environment, TCP/IP must be started on the source and target system. The subsystem QSERVER must be active in *all* AS/400 systems involved in the file system sharing. A job named QPWFSERVSD is automatically activated in QSERVER for QFILESVR.400 support.

On systems prior to V4R3, if the subsystem QSERVER is started before TCP/IP, start the program QPWFSERVSD manually. For a convenient use of QFILESVR.400, modify the OS/400 start up program to start TCP/IP *before* the QSERVER subsystem is started. Communication links need to be ready among the participating systems.

To enable remote file system sharing, create a directory, *directly* under QFILESVR.400. Use the same name as the TCP/IP host name of the target system to which to connect.

Use Operations Navigator or the OS/400 Make Directory (MD) command to create a new folder named *HostName* under QFILESVR.400. An example of this command is:

```
MD '\QFILESVR.400\HostName'
```

Note: The target host name must exist either in the local AS/400 host table or in the domain name server (DNS) known to the local AS/400 system in order for the connection to be successfully established.

To access the target file system, use matching user IDs and passwords on both the local (or source) and target systems.

\QFILESVR.400*HostName* in the source system represents the entire Root file system of the target system. The amount of data you can access depends on your authority on the target system.

An example of how to use the QFILESVR.400 system to perform disk storage copy activities is in the following section.

6.2 Copying objects between systems

The CPY command allows you to copy objects within a file system or to another file system that supports the same object format. For example, to copy a stream file named clref1of4.pdf from a local directory named \usr\mydir to a remote directory named \dira on the system ITSOSYS2, the CPY command used is shown in Figure 53.

```
                          Copy Object (CPY)

Type choices, press Enter.

Object . . . . . . . . . . . . . > '\usr\mydir\clref1of4.pdf'

To directory . . . . . . . . . . > '\qfilesvr.400\itsosys2\dira'

To object  . . . . . . . . . . .   ________________________

Symbolic link  . . . . . . . . .   *NO           *NO, *YES
From Code Page . . . . . . . . .   *OBJ          1-32767, *OBJ, *PCASCII
To Code Page . . . . . . . . . .   *OBJ          1-32767, *OBJ, *CALC...
Data Format  . . . . . . . . . .   *BINARY       *BINARY, *TEXT
```

Figure 53. Copy Object display

When the command parameter To directory is specified, the target file has the same name as the source file. If you want a *different* file name on the target, use the parameter To object instead. For example, to copy a target file named xyz.pdf to the same target directory, use the CPY command as shown in Figure 54.

```
                          Copy Object (CPY)

Type choices, press Enter.

Object . . . . . . . . . . . . . > '\usr\mydir\clref1of4.pdf'

To directory . . . . . . . . . .   '.'

To object  . . . . . . . . . . . > '\qfilesvr.400\itsosys2\dira\xyz.pdf'

Symbolic link  . . . . . . . . .   *NO           *NO, *YES
From Code Page . . . . . . . . .   *OBJ          1-32767, *OBJ, *PCASCII
To Code Page . . . . . . . . . .   *OBJ          1-32767, *OBJ, *CALC...
Data Format  . . . . . . . . . .   *BINARY       *BINARY, *TEXT
```

Figure 54. Copy Object display

Use the CPY command to copy objects in the QSYS.LIB file system. In this case, the CPY command views a file object in QSYS.LIB as a directory name and its members as file names. To copy a physical file object, a file with the same record format must already exist in the target system.

For example, to copy a member named CUSTCDT of a physical file named QCUSTCDT from a local library named QIWS to a remote library named LIBA on the system ITSOSYS2, the CPY command is used as shown in the following example.

```
CPY OBJ('\qsys.lib\qiws.lib\qcustcdt.file\custcdt.mbr')
TODIR('\qfilesvr.400\itsosys2\qsys.lib\liba.lib\customer.file')
```

For the Copy command to work successfully, the file CUSTOMER in the library LIBA must exist in the ITSOSYS2 system with its record format matching that of QCUSTCDT in the source system.

When the command parameter To directory is specified, the target member has the same name as the source member. For a different target member name, use the parameter To object. For example, to create a member named XYZ in the same target physical file as in the previous example, use the CPY command as shown in the following example:

```
CPY OBJ('\qsys.lib\qiws.lib\qcustcdt.file\custcdt.mbr')
TOOBJ('\qfilesvr.400\itsosys2\qsys.lib\liba.lib\customer.file\xyz.mbr')
```

To copy multiple objects with a single command, use a wildcard (the symbol *). The following command copies all members of the LIBA/QCUSTCDT file to the target file LIBB/CUSTOMER in the system ITSOSYS2:

```
CPY OBJ('/qsys.lib/liba.lib/qcustcdt.file/*')
TODIR('/QFileSvr.400/itsosys2/qsys.lib/libb.lib/customer.file')
```

6.2.1 Sharing direct-attached optical library

The storage space of an AS/400 direct-attached optical library on the local system is accessed through the Integrated File System name space. Use the QFILESVR.400 file system to access optical storage installed on one AS/400 system, from remote AS/400 systems in the network. QFILESVR.400 enables a sharing of the optical resource.

Save and restore commands utilizing optical media are supported on a local, direct-attached optical library on AS/400 systems from OS/400 V3R7 onward. This process is designed as an archive and retrieval system rather than backup and recovery.

A save and restore interface to directly attached optical media libraries (and CD-ROMs) makes the SAVOBJ, SAVLIB, RSTOBJ, and RSTLIB commands available to use optical storage. Not all save and restore commands are supported. Some commands have parameter restrictions. For example, the OS/400 SAV and RST commands do not support QFILESVR.400.

Use the CPY command, as described in 6.2, “Copying objects between systems” on page 82, to perform archival and retrieval on optical storage from a remote system.

Two categories of APIs can be used for optical files and directories:

- Hierarchical File System (HFS) APIs
- Integrated File System support, which consists of UNIX-type APIs and the generic command interface

For the optical file system identified as the receiver of a request submitted, either to HFS or the IFS, the first portion of the path name parameter must be /QOPT. Detailed specifications about using the HFS or UNIX-type APIs can be found in *System API Reference*, SC41-5801.

6.3 QFILESVR.400 considerations

To efficiently use the QFILESVR.400 file system, understand these points:

1. The file system sharing under QFILESVR.400 does not survive a system IPL. Include the Make Directory (MD) command in the startup program for a “virtually permanent” connection if appropriate.

2. Case-sensitive names depend on the specific file system accessed on the target system. For example, case-sensitivity is not supported for object searches over the QFILESVR.400 file system.
3. The QFILESVR.400 file system periodically checks every two hours to determine if there are any connections, which are not used (for example, no files opened associated with the connection). Connections that have no activity for a two-hour period are terminated automatically.
4. The QFILESVR.400 file system does not detect loops. The following path name is an example of a loop, where “Remote1” is the local system:

   ```
   /QFileSvr.400/Remote2/QFileSvr.400/Remote1/
   QFileSvr.400/Remote2/...
   ```

   When the path name containing a loop is specified, the QFILESVR.400 file system returns an error. The error indicates that a time-out has occurred.
5. The QFILESVR.400 file system uses an existing free session to communicate over a SNA LU 6.2 link. Start the mode and establish a session between systems to successfully connect QFILESVR.400 to the remote target system.

Chapter 7. OS/400 disk storage management tools

Maintaining an efficient use of AS/400 storage is an essential task and deserving of high priority. Disk storage is an important component of any computer system as it provides active and permanent access to the data necessary to run a business. Efficient disk storage management helps maintain consistent access to data, as well as prevents filling up disk storage before due time.

There are quite a few OS/400 utilities for you as a system administrator to use to manage AS/400 disk resources in order to maintain its efficient utilization. In this chapter, we describe disk management tools based on OS/400 V4R4. Some of the tools discussed are also available in previous releases and noted accordingly.

The tools covered in this chapter are:

- Retrieve Disk Information (RTVDSKINF) and Print Disk Information (PRTDSKINF) commands
- Trace ASP Balance (TRCASPBAL), Start ASP Balance (STRASPBAL) and End ASP Balance (ENDASPBAL) commands
- Disk Balance (DSKBAL) command
- Management Central Disk Resource Utilization Monitor
- Start Disk Reorganization (STRDSKRGZ) and End Disk Reorganization (ENDDSKRGZ) commands

Find additional information on these tools at the AS/400 Beyond Technology Web page at: `http://www.as400.ibm.com/beyondtech`

7.1 Identifying disk storage consumption of AS/400 objects

One of the most demanding tasks is to maintain efficient use of AS/400 disk storage, whether the expanding amount of information stored on the AS/400 system is due to business growth, or represents information that is not used frequently (or at all). After all, some systems are installed with the planned intention to keep as much information online as practical.

Online data storage “build-up” results when users hesitate deleting information, on the chance that it will be *needed again.* The hesitation withstands whether the data is available dynamically or for manual retrieval from offline storage. Clean-up tasks can be enforced, and automated. However, disk storage needs continue to grow.

For example, copies of main storage dumps are not automatically deleted or “cleaned up”. Each one is only manually deleted when the user (customer or service center) determines that the problem associated with the dump is solved. This is done by starting the Main Storage Dump Manager service tool. First, select option **2**, Work with dump copies, and then select option **4**, Delete selected dump copies. If there is not enough space to hold a new dump copy in auxiliary storage, the error message `Insufficient storage to copy dump` is displayed at the bottom of the Main Storage Dump Manager service tool screen. Then, the copy operation is cancelled. Main storage dumps are not stored entirely on the load source. They are stored in files scattered on DASD.

© Copyright IBM Corp. 2000

Identifying large objects that may contribute to an inefficient consumption of the system disk space helps keep this growth in check. In many situations, you can make a proper decision on what action to take to ensure an efficient usage disk space if you can identify all large database objects in your system. Display their current disk space consumption status by using the Display File Description (DSPFD) command.

The DSPFD command provides a lot of information about the file, including its record length in bytes, the total number of active records, and the total number of deleted records which are not yet purged. For example, a file with a record length of 100 bytes, a total of one million (active) records, and a total of six million deleted records, means that the file is about 700 MB in size. 600 MB is occupied by the deleted records.

In a situation like this, run the Reorganize Physical File Member (RGZPFM) command over the file with too many deleted data records. RGZPFM frees up the disk space the records occupy.

You can also use the Change Physical File (CHGPF) command on the file and set the Reuse deleted record (REUSEDLT) attribute, if appropriate, to *YES. This prevents inefficient growth of the number of deleted records and thus prevents the need to run the RGZPFM command on specific files too often. Note that this action may require a change in operational or programming procedures dependent upon whether the applications using the file are designed with the REUSEDLT characteristic in mind.

For detailed information on how to use the REUSEDLT attribute of a physical file, refer to *AS/400 Database Programming,* SC41-5701.

Tip

A situation where having “reuse deleted record” set to *YES is useful is when the average number of deleted records, over a reasonable period of time (per day or week), exceeds, or is about the same as, the average number of new records added. In such a case, *YES helps prevent new space allocation of the file and thus reduces the growth rate of the file size. If the average number of deleted records is *less* than the average number of new records added, *YES may not be as useful.

Set “reuse deleted record” of a file to *NO if you need:

- To maintain *arrival sequence* of the added records.
- To implement first-in-first-out (FIFO) or last-in-first-out (LIFO) ordering for data records with duplicate key field value.
- To access the data record by the *relative record number* (RRN). This is a strategy from System/36 days. The RRN is reused when *YES is specified.

In another situation, you might use database journaling for transaction-oriented applications using commitment control. This can result in a situation where the journal receivers grow unnecessarily large. Set a limit on practical sizes for the receivers and have the system automatically delete the receivers that reach the size limit, by creating a new journal receiver with the Journal Receiver Threshold (THRESHOLD) attribute set to a proper size. Use the Change Journal (CHGJRN) command to attach the new receiver, then set the Manage receivers (MNGRCV) attribute to *SYSTEM and the Delete receivers (DLTRCV) attribute to *YES. This helps control the maximum amount of allocated disk space for the journal receivers.

Database journaling is designed for data recovery purposes. When the receivers grow too fast, use the CHGJRN command to set the Receiver size options (RCVSIZOPT) attribute of the journal to remove internal entries (*RMVINTENT). Use a minimum fixed length (*MINFIXLEN) for the journal entries to help control the growth rate of the receivers.

Refer to *AS/400 CL Reference*, SC41-5722, for more information on the CHGJRN command.

7.1.1 Disk information tool

Before you can take disk space management actions, you need to identify objects that are potential suspects for inefficient disk consumption. There are two OS/400 commands to use for this purpose:

- Retrieve Disk Information (RTVDSKINF)
- Print Disk Information (PRTDSKINF)

These commands have been available since OS/400 V3R1.

RTVDSKINF collects disk space information and stores it in the database member named QCURRENT of a database file named QAEZDISK, located in the QUSRSYS library. Every object in the system has a corresponding data record in this QAEZDISK file. Each record in QAEZDISK contains such object information as object type, size, owner, last changed date and so on.

Tip

Each time RTVDSKINF is run, the member QCURRENT is written over. To save existing information in QCURRENT for later use, rename the QAEZDISK file or duplicate the data into another file before you run RTVDSKINF.

PRTDSKINF produces a spooled print file containing a report of the disk consumption. Reports are selected based on:

- User profile (object owner) information
- Object information
- System object information
- Library information
- Folder information

The list of information in the report is sorted based on such criteria as:

- Object size in descending order
- Owner name in alphabetical order
- Object name in alphabetical order
- Last-change date in order of time
- Last-use date in order of time

Use the Submit Job (SBMJOB) command to run the RTVDSKINF and PRTDSKINF commands in batch mode. Or use the Disk Tasks menu (GO DISKTASKS) to initiate commands as shown in Figure 55.

```
DISKTASKS                      Disk Space Tasks
                                                            System: ABCDERSD
To select one of the following, type its number below and press Enter:

     1. Collect disk space information
     2. Print disk space information

    10. Work with libraries
    11. Work with folders
    12. Work with objects by owner









Type a menu option below __
```

Figure 55. DISKTASKS menu

Before you submit the RTVDSKINF command or use the DISKTASKS menu, make sure your user profile exists in OS/400 System Distribution Directory Entry. Use the Work with Directory Entries (WRKDIRE) command to add the necessary user profile or display entries. Otherwise, RTVDSKINF does not retrieve all information on system objects before ending, and PRTDSKINF does not have what it needs to produce output for the *Specific object information* portion of the report.

Tip

Do not run the RTVDSKINF command when the system is experiencing a heavy workload. The RTVDSKINF job cannot collect object information on objects in use. RTVDSKINF locks the objects while collecting information on them, which leads to file contention situations (locking and seize problems) if you run RTVDSKINF while your system is in heavy use. Off-hour operation is preferrable in most cases. Alternately, run the RTVDSKINF command after an IPL.

7.1.1.1 Disk information report using a 5250 session

Follow this sample procedure to identify large objects in your system:

1. Enroll the user running this procedure in the system directory:
   a. From a command line, type the command WRKDIRE to invoke the Work with Directory Entries screen.
   b. Type 1 in the Opt field and fill in your User ID and Address. Press Enter.
   c. Fill in the Description field and specify your User profile, which is the same one you use to sign on to the system, and the user profile entered in the system directory as in the beginning of this procedure. Other fields are

optional. Press Enter when done. Then press the F3 key to return to the OS/400 command line.

You are now ready to collect and print a report on disk space information.

2. Type the following command:

```
SBMJOB CMD(RTVDSKINF)
```

   Then press Enter. RTVDSKINF does not have any parameters, so submit it right away.

3. Use the `WRKSYSSTS` command to monitor the status of your submitted job. Wait for the job to finish. This takes anywhere from a few minutes to a few hours, depending on the disk space allocation and overall system performance. Because the collected disk information is stored in the physical file QUSRSYS/QAEZDISK, you can wait until a later time to continue this procedure.

4. When RTVDSKINF finishes, produce a report by submitting the following command:

```
SBMJOB CMD(PRTDSKINF)
```

   Then press F4 key *twice* to bring up parameters of the PRTDSKINF command as shown in Figure 56.

```
                    Print Disk Information (PRTDSKINF)

Type choices, press Enter.

Type of report . . . . . . . . . > *OBJ          *LIB, *FLR, *OWN, *OBJ, *SYS
Objects  . . . . . . . . . . . . > *ALL          Name, generic*, *ALL, *NONE
Object types . . . . . . . . . .   *ALL          *ALL, *ALRTBL, *AUTL...
              + for more values
Smallest size  . . . . . . . . .   50000         Size in 1000 bytes
Sort by  . . . . . . . . . . . .   *SIZE         *SIZE, *OWNER, *LSTCHG...
```

Figure 56. PRTDSKINF command parameters

5. Specify the Type of report that you wish to see. For example, to see a list of all objects in descending order of their size with a minimum of around 50 MB, specify:

   - `*OBJ` for Type of report.
   - `*ALL` for Objects and Object types.
   - A value of `50000` for Smallest size. This field accepts a decimal value of five positions and specifies the size in kilobytes (1000 bytes). The maximum value you can specify is 99999, which is about 100 MB.
   - `*SIZE` for Sort by.

   Press Enter when the parameters are filled in.

6. Wait for the job to finish. Then use the `WRKSPLF` command to locate and display the spool file named QUSRSYS/QPEZDISK. This report contains several disk space reports with information on:

   - System components
   - OS/400 items
   - OfficeVision/400 items

- Licensed Internal Code
- Space used by internal objects
- Specific object information
- Miscellaneous information

An example of the PRTDSKINF report is shown in Figure 57.

```
                                     Disk Space Report                                        Page  1
5769SS1 V4R4M0  990521                                                 ITSOSYS1 09/10/99  10:31:50

                                     System Information

 Information collected . . . . . . . . :   09/09/99  16:32:56   Customize options specified for this report:
                                                                  Report type . . . . . . . . . . . . . :  *OBJ
 Total disk space on system in 1,000,000                          Objects . . . . . . . . . . . . . . . :  *ALL
   bytes . . . . . . . . . . . . . . . :   9836                     Object types:
 Main storage size in megabytes . . . . :  128                        *ALL

 Machine type-model . . . . . . . . . . :  9402-400
 System serial number . . . . . . . . . :  1X-XXXXXX


                                                                  Size of smallest object . . . . . . . :  50000
                                                                  Sort by . . . . . . . . . . . . . . . :  *SIZE

                                           % of       Size in
 Description                               Disk    1,000,000 bytes

 User libraries                            11.26        1107.61
 User directories                           5.80         570.61
 Folders and documents                       .52          50.92
 QSYS                                      13.39        1316.72
 Other IBM libraries                       13.64        1341.33
 Licensed Internal Code                    11.77        1157.81
 Temporary space                            5.65         555.88
 Unused space                              35.59        3500.79
 Internal objects                            .61          59.53
 Objects not in a library                   3.00         295.46

 TOTAL                                    101.23        9956.66


                                                         % of       Size in
 Description                                             Disk    1,000,000 bytes

 OS/400 items reported:
   History files                                          .04             3.71
   Journal receivers                                      .04             4.29
   User profiles                                          .10             9.94
   Configuration information                              .01              .88
   System help (QHLPSYS)                                  .12            12.17
   Calendars                                              .01             1.18
   System directories                                     .02             2.22
   Document files                                         .02             2.08
 OfficeVision/400 items reported:
   Enrollment                                             .00              .00
   Mail files                                             .00              .00
   Text search                                            .00              .00
   Personal directories                                   .00              .00
 Licensed Internal Code:
   LIC and tables                                        9.43           927.79
   Dump space                                            1.38           135.27
   Error logs                                             .01              .94
   Trace tables                                           .33            32.79
   VLIC logs                                              .55            54.53
   VLIC control blocks                                    .07             6.49

 * This list points out items of interest and does not contain all
 objects on the system.  For more information, see AS/400 System
 Operation V3R6, SC41-4203.
```

Figure 57. PRTDSKINF Disk Space Report (Part 1)

```
                                            Disk Space Report                                                          Page    2
5769SS1 V4R4M0  990521                                                                                  ITSOSYS1 09/10/99  10:31:50

                                                 System Information*

                                                                          % of        Size in
 Description                                                              Disk     1,000,000 bytes

 Space used by internal objects:
   Additional configuration information                                    .00                .11
   Communications                                                          .07               6.71
   Data base                                                               .02               1.52
   Manage jobs                                                             .31              30.84
   Monitor performance                                                     .00                .00
   Folders                                                                 .00                .44
   Mail                                                                    .00                .00
   OS/400                                                                  .06               6.23
   S/36 Environment                                                        .00                .00
   Security                                                                .00                .11
   SNADS and object distribution                                           .01                .75
   Spool                                                                   .06               6.39
   Cross component                                                         .01               1.26
   Product installation                                                    .00                .09
   Recovery of information                                                 .02               1.98
   Obsolete                                                                .00                .00
   Internal system journals                                                .03               3.10
 Miscellaneous items reported:
   Printer output & Spooling (QSPL)                                        .49              48.06
   Replaced object library (QRPLOBJ)                                       .00                .34
   Storage affected by RCLSTG                                              .00                .03
   Reclaim storage library (QRCL)                                          .00                .00

* This list points out items of interest and does not contain all
objects on the system.  For more information, see AS/400 System
 Operation V3R6, SC41-4203.

                                              Specific Object Information

                                              % of        Size in  Last      Last
Object      Library    Type      Owner       Disk     1000 bytes  Change    Use       Description

QAPMJOBS    QPFRDATA   *FILE     QPGMR       3.76       369586.2  08/03/99  08/03/99  Job related performance data
ISAJ000217  QGPL       *JRNRCV   DHQB        1.07       105545.7  02/25/99
ISAJ000225  QGPL       *JRNRCV   DHQB         .90        88137.7  09/09/99
ISAJ000220  QGPL       *JRNRCV   DHQB         .90        88133.6  06/07/99
QTQIINIT    QSYS       *SRVPGM   QSYS         .82        80285.7  09/02/99

TOTAL                                                   731688.9

                                 * * * * *   E N D   O F   L I S T I N G   * * * * *
```

Figure 58. PRTDSKINF Disk Space Report (Part 2)

The last portion shown on page 4 of the report is titled Specific Object Information. This section contains a list of objects in descending order of their size as specified in the `PRTDSKINF` command. In this example, we specified 50 MB as the lower limit of object size. All the objects displayed in this report are larger than 50 MB in size. You can see one large file and three large journal receivers in this portion of the sample report.

Other useful information is also presented. For example, on page 1 of the report, you see a line indicating `Objects not in a library`. This primarily includes objects that are not fully created, or not fully deleted, and will be deleted from the system when the Reclaim Storage (RCLSTG) command is run. It also includes job message queues. Job message queues are traditional system objects, but do not reside in libraries.

You can also identify if additional disk space can be gained after running RCLSTG. Look on page 4 for the line indicating `Storage affected by RCLSTG` under Miscellaneous items report.

With the information provided by the report, you can take an informed action in managing disk space utilization.

7.1.1.2 Disk information report using AS/400 Operations Navigator

On V4R4 systems, the operations described in the preceding section are performed from Operations Navigator, if you prefer to do so.

On the PC side, use the Operations Navigator program code that comes with the shipped OS/400 V4R4 system, or from the Client Access Express V4R4 install CD. Run the `STRTCPSVR SERVER(*MGTC)` command to start the AS/400 server component of Management Central before using Operations Navigator from the PC.

Use the Run Command function of the Management Central component of the Operations Navigator to submit an OS/400 command. Before using the Run Command function, create an *endpoint system* entry for the AS/400 system against which you run the command. See 7.3, “Management Central disk resource utilization monitor” on page 111, for information on how to create an endpoint system using Management Central. When the endpoint system is created, complete the following steps to run RTVDSKINF and PRTDSKINF:

1. Open Operation Navigator and enroll your user profile into the system Distribution Directory. Expand the target AS/400 system icon to view the functions available.

   Expand the **User and Groups** icon, and then click **All Users** (Figure 59). All user profiles in the system are displayed on the right side of the window.

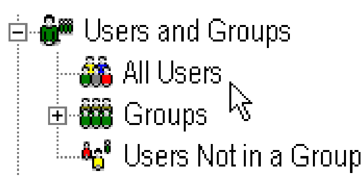


Figure 59. Operations Navigator: Users and Groups

2. Double-click your user profile to bring up the Properties window. Then click the **Personal** button (Figure 60).

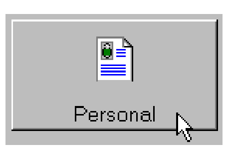


Figure 60. Personal button in the User Profile’s Properties window

3. Click the **Mail** tab to bring up the System Distribution Directory Entries information (Figure 61). All fields are already filled with default information for your user profile. Make any necessary changes. Then click **OK**.

   If you are already enrolled in the Distribution Directory, the fields User ID and Address are displayed in *gray background* instead of white. Make changes if required. Otherwise, click **OK** to return to the User ID Properties window.

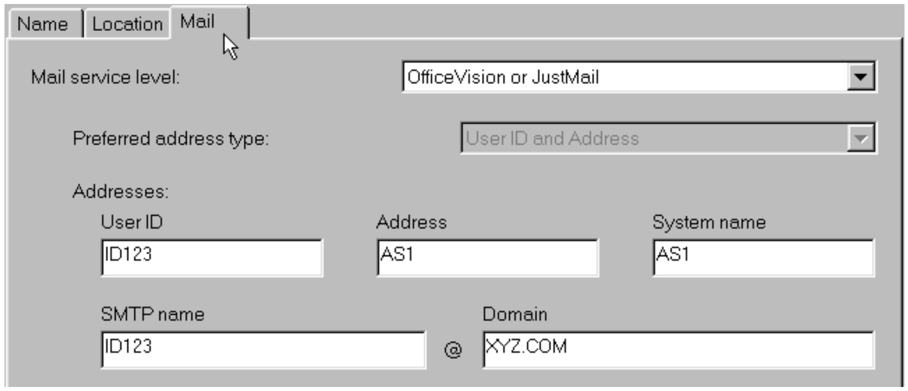


Figure 61. Personal window: Mail tab

4. Click **OK** in the User ID Properties window to finish the enrollment.
5. On the left side of the Operation Navigator window, right click the target AS/400 endpoint system name to bring up the pop-up menu.

   Note: You may need to expand the Management Central and AS/400 Endpoint Systems lines before you can see the system name. If you cannot see your system here, create it. See 7.3, "Management Central disk resource utilization monitor" on page 111, for steps on how to create an endpoint system using Management Central. Return to this step when finished.
6. Highlight the menu item **Run command...**, and click once to bring up the Run Command window (Figure 62).

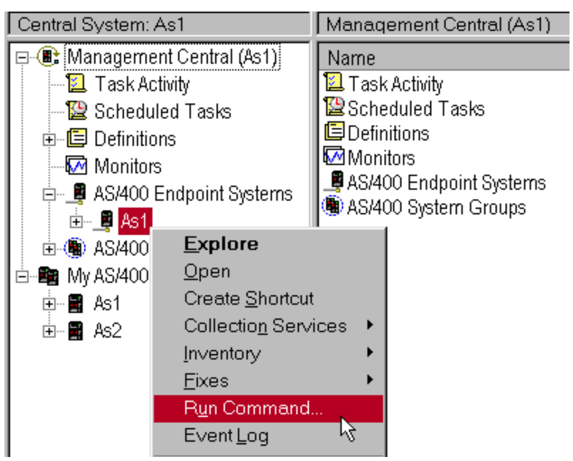


Figure 62. Invoking the Run Command function

7. Type the command `RTVDSKINF` (Figure 63 on page 96). Click the **Check Syntax** button to verify if the command format typed is correct. Click **OK** to submit the command to the AS/400 system immediately.

   Note: In contrast to a 5250 session environment, you do not use the Submit Job (SBMJOB) command to submit the command in a Management Central environment. The command you submit under Management Central runs under its own environment in the QSYSWRK subsystem. Use the `SBMJOB` command if you prefer to run the job in other subsystems. However, when not running in QSYSWRK, you are not able to monitor the job status from Management Central - Task Activity.

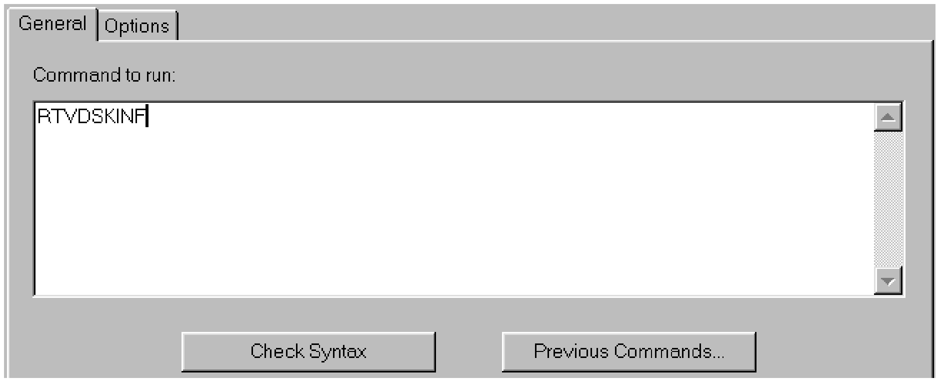


Figure 63. Run Command window

8. Monitor the status of the submitted job to see when it finishes. Do this by clicking the **Task Activity** icon under **Management Central**. All the tasks initiated by Management Central are displayed on the right side of the window. A status of Started indicates a submitted task is in progress.

   Right-click the task icon, and select **Status** to see the detail of the status. This brings up another Run Command Status window that displays a few details of the task as shown in Figure 64.

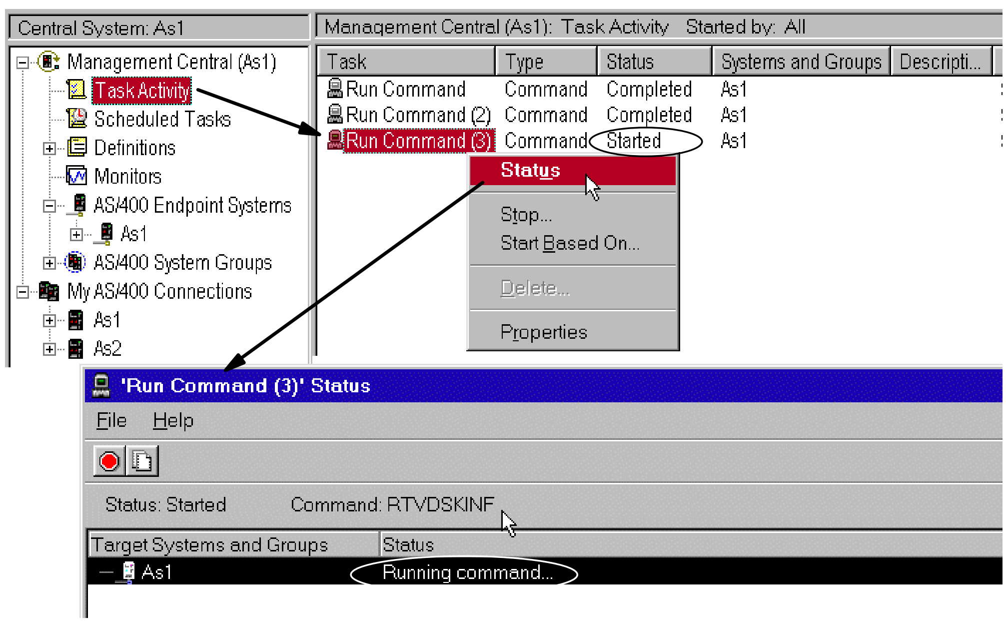


Figure 64. Management Central: Task activity

9. Wait for the job to finish (as indicated when the Status field changes from Started to Completed). This takes anywhere from a few minutes to a few hours depending on the disk space allocation and system performance. Because the collected disk information is stored in the physical file QUSRSYS/QAEZDISK, you can wait until a later time to continue.

   While you wait for the task to finish, you can close the Operations Navigator and turn off your PC. Reopen it later to display the status of the task.

10.When the RTVDSKINF task finishes, produce a report by submitting the PRTDSKINF command. Use the same Run Command function as described in the preceding steps, that is:

```
PRTDSKINF RPTTYPE(*OBJ) OBJ(*ALL) OBJTYPE(*ALL) MINSIZE(50000) SORT(*SIZE)
```

Note

Although there is a Check Syntax button available for use, there is no prompting function (F4) for a list of command parameters in the Operations Navigator environment, compared to 5250 session environment. Enter the command in a keyword format with all the required parameter keywords and values specified.

11.Wait for the job to finish. Right-click the completed task in the Task Activity list. Select **Status** to invoke the Run Command Status window as shown in Figure 65.

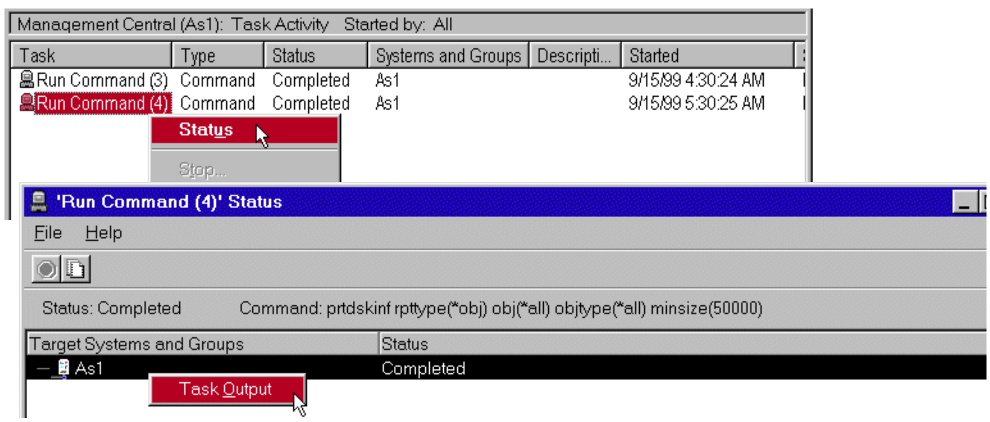


Figure 65. Management Central: Displaying the spooled output of a completed task

In this window, right-click the system name, and select **Task Output** to bring up the Printer Output - (System name) window.

12.In the Printer Output window, right-click the spooled file name listed under Output Name field. Select **Open** to invoke the AFP Viewer to display the content of the spool file.

The same reports are produced as discussed in 7.1.1.1, “Disk information report using a 5250 session” on page 90.

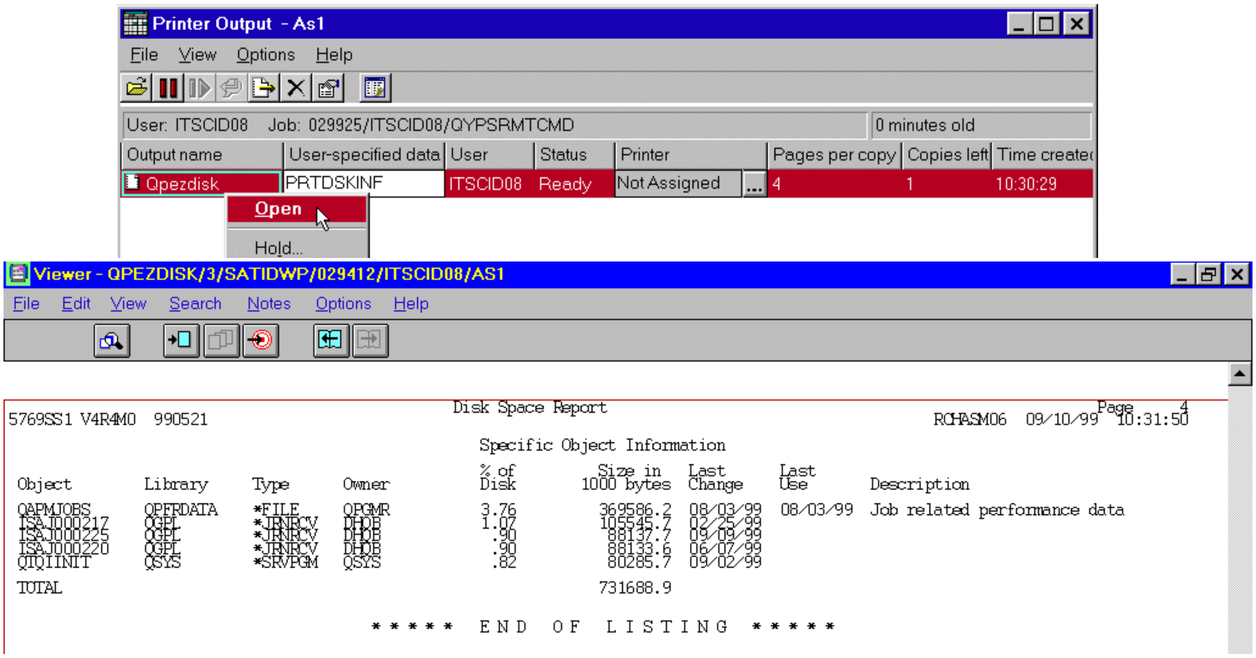


Figure 66. Viewing a spooled file with AFP Viewer

The last portion of the report is titled Specific Object Information, as shown on page 4 in our example in Figure 66. This section lists the objects in descending order of their size, as specified in the PRTDSKINF command. In this example, we specified 50 Mbyte as the lower limit of the object size. So, all the objects displayed in this report are larger than 50 MB in size.

Other useful information is also presented in this report. For example, on page 1 of the sample report, there is a line indicating `Objects not in a library`. This value primarily includes objects that are not fully created or not fully deleted. They would be deleted from the system if RCLSTG is executed.

The report also includes job message queues. Job message queues are standard system objects but do not reside in libraries. Look for a line indicating `Storage affected by RCLSTG` on the Miscellaneous items report section. This information identifies if additional disk space could be gained by running RCLSTG. Refer to page 3 in the sample report.

Refer to *Management Central: A Smart Way to Manage AS/400 Systems*, SG24-5407, for detailed information on how to use many functions delivered in Management Central.

7.1.2 Creating a customized disk information report

QUSRSYS/QAEZDISK contains information on objects as of the time the last RTVDSKINF command is run. Additional information, different formatting, and alternate sort sequences is available beyond what PRTDSKINF provides using data from the QAZDISK file.

To produce a customized disk information report, use a query tool of your choice with the QAEZDISK file as input. The record format of the file QAEZDISK is shown in Figure 67.

Data Field	Field Type	Column Length	Heading
DIOBLI	CHAR	10	Object Library
DIOBNM	CHAR	12	Object Name
DIPRFL	CHAR	12	Parent Folder
DIOBTP	CHAR	7	Object Type
DIOBAT	CHAR	9	Object Attribute
DIOBSZ	PACKED	15 0	**Object Size**
DIOBTX	CHAR	50	Text Description
DICCEN	CHAR	1	Last Changed Century
DICDAT	CHAR	6	Last Changed Date
DICTIM	CHAR	6	Last Changed Time
DIOBUS	CHAR	1	Object Use Indicator
DIUCEN	CHAR	1	Last Used Century
DIUDAT	CHAR	6	Last Used Date
DIUTIM	CHAR	6	Last Used Time
DISTIN	CHAR	1	Status Indicator
DIOBOW	CHAR	10	Object Owner
DIOBAS	CHAR	2	Object ASP
DIFLPT	CHAR	63	Folder Path

Figure 67. Record format of QAEZDISK file

The field named DIOBSZ contains the object size in bytes. The object owner is in the DIOBOW field and the object type is in DIOBTP field. The field DIOBAT contains the object attribute, which further qualifies the object type. For example, a program object has an object attribute that clarifies what language it is compiled from (such as RPG, C, CBL, or CLP). Some object types do not have a further qualifier.

To generate a report on *how much disk space is consumed by RPG program objects which are owned by a particular user*, run RTVDSKINF. When it finishes, run a query equivalent to the following SQL statement (or use the SQL statement itself):

```
SELECT SUM(DIOBSZ) FROM QUSRSYS.QAEZDISK WHERE DIOBOW='a_user_profile' AND
DIOBTP='PGM' AND DIOBAT='RPG'
```

The result of this SELECT statement is the total size in bytes of the RPG objects owned by a specific user profile.

To list the file types available in QAEZDISK, run the following SQL statement:

```
SELECT DISTINCT(DIOBTP) FROM QUSRSYS.QAEZDISK
```

Use the following statement to see the available object attributes:

```
SELECT DISTINCT(DIOBAT) FROM QUSRSYS.QAEZDISK
```

The example described in this section shows the flexibility available when customized reports are needed.

7.2 The disk balance tool

Maintaining a balanced utilization of disk storage, both in terms of space consumption and data access performance, helps your system run smoothly. It also helps reduce the chance of a system interruption due to disk storage problems.

This section discusses the OS/400 V4R4 disk balance tool which can be used to balance information on the disk storage system. The disk balance tool is available on V4R4 systems.

Note: There is limited support for disk balance tools delivered as PTFs for V4R3, V4R2 and V4R1, as identified in 7.2.6, “DSKBAL command” on page 109.

The disk balance tool is useful in the following situations:

- When you add new disk units into an Auxiliary Storage Pool (ASP) and want to spread existing data onto the newly added units for an even space utilization on all disk units in the ASP.

  Prior to V4R1, to achieve a disk capacity balance, you save and restore the entire ASP. If this balance involves the system ASP, an entire system save and restore is necessary. This is not an efficient method in most customer settings. Therefore, in most cases, newly added disk units are left empty and OS/400 populates new data onto them as time goes by. This situation can cause a performance problem since most of the new data is allocated onto the new units, resulting in an uneven amount of I/O requests to the disk units in the same ASP.

- You may suspect that a particular group of frequently accessed database files are not evenly allocated on all disk units in the ASP. This causes a performance bottleneck because certain disk units have too many I/O requests compared to the rest of the disk units in the ASP. Use the Work with Disk Status (WRKDSKSTS) command, or collect performance data for an analysis of whether disk units have a balanced amount of I/O requests.

  For systems prior to V4R4, it is normally recommended that you save these objects onto tape media, rename them on the disk storage, and restore the original objects back. Delete the renamed objects if the restore completes normally. OS/400 tries to allocate the objects evenly to all disk units in the ASP, if possible, at restore time. However, this might be impractical because, in many cases, it takes much effort to identify the group of frequently accessed objects in the first place.

- Hardware disk compression lets you choose to compress disk units in a *User ASP* on a disk arm basis. An estimated two to three times the disk space is gained, dependent upon the characteristics of the data being compressed. An access time penalty of 0 to 30% is possible. Therefore, enable compression when disk storage capacity (or efficiency) is of higher priority than performance.

  Disk compression is available in V4R3 or later with hardware that supports compression.

  If compressed disk units are mixed with uncompressed ones in the same user ASP, you can place rarely accessed objects onto compressed the units. This “preserves” the uncompressed, better-performing disk units for objects that are accessed more frequently.

7.2.1 Disk balance type

The scenarios described beginning in 7.2, “The disk balance tool” on page 99, are addressed by the ASP Balance commands in V4R4 using the Trace ASP Balance (TRCASPBAL), Start ASP Balance (STRASPBAL), and End ASP Balance (ENDASPBAL) commands.

OS/400 V4R4 provides three types of disk balance processes: capacity, usage, and HSM balance. Each of these balance types is invoked by the Start ASP Balance (STRASPBAL) command.

- **Capacity balancing**: Capacity balancing is when the data on all disk units within the ASP is balanced so that each unit has an equal distribution of percentage used and percentage unused space, as pictured in Figure 68.

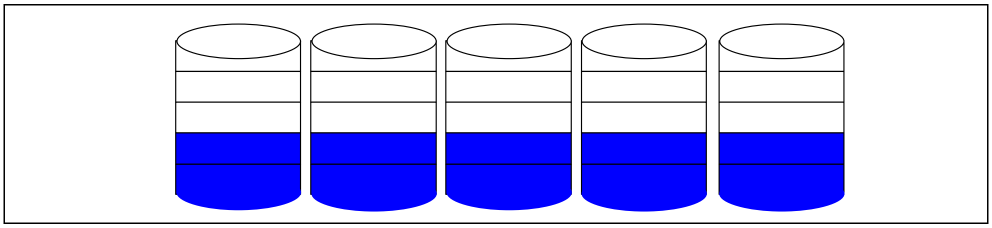


Figure 68. Capacity balance: The objective

An even distribution of data can help reduce the chance of a disk unit overflow, assuming total disk storage is monitored for threshold.

Capacity balancing is useful when *new* units are added to an ASP. Instead of having several units with the majority of the data and several new ones with a much lesser proportion of data, data is spread evenly across all disk units before using them in your normal system environment.

A conceptual view of the capacity balancing process is shown in Figure 69.

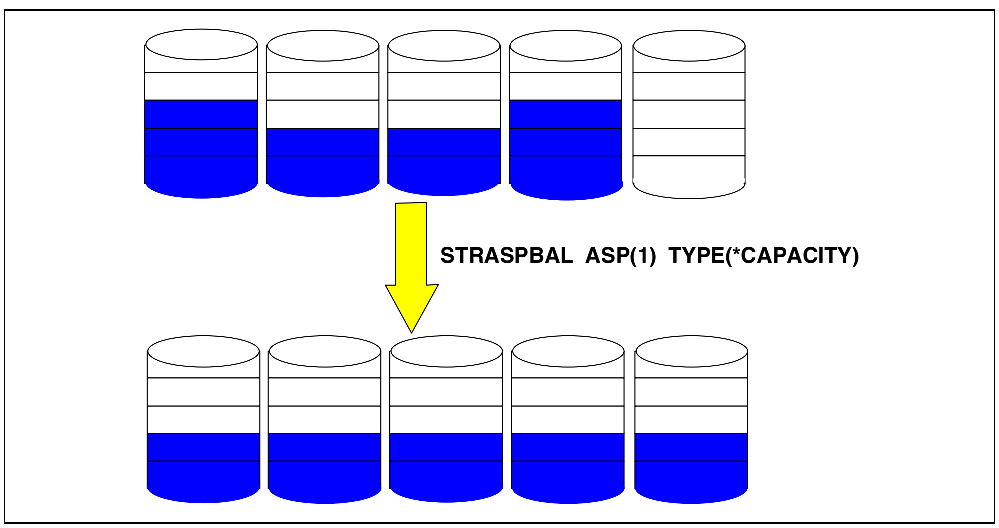


Figure 69. Capacity balance process: A conceptual view

Capacity balancing is started from System Service Tools (SST) or Dedicated Service Tools (DST). Use the Add units to ASPs and balance data menu option.

Note: When started from this menu option, if an IPL is performed before the capacity balancing process completes, balancing restarts automatically during the next IPL.

- **Usage balancing**: Usage balancing is when the “low-use” pieces of data on all disk units in a specified ASP are redistributed. This helps prevent the accumulation of high-use data on too few disk units, which in turn improves data access performance. “High-use” data is not moved.

A conceptual view of the objective of usage balancing is shown in Figure 70.

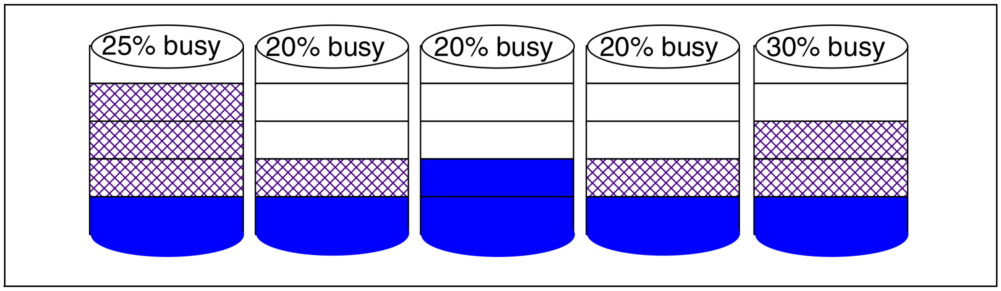


Figure 70. Usage balance: The objective

Based on the traced disk usage statistics, the usage balancing process reallocates the low-use data in such a way that it influences future data allocation requests on disk units in the ASP. This is done with an attempt to identify whether there are disk units that are statistically *over*utilized and *under*utilized. If so, the system tries to move existing low-use data from underutilized disk units onto overutilized disk units.

Note: To minimize a performance impact, *USAGE balancing does not move hot data.

Usage balancing prevents the system from allocating further data on the overutilized units and thus prevents additional utilization. This in turn leaves a larger amount of empty space on the underutilized disk units, which leads to the system allocating new data onto them. Finally, the underutilized units become better utilized from the presence of new high-use data.

At the completion of the usage balancing process, all disk units in your system may not have equal percentage of used space because capacity balancing is not the focus of usage balancing.

A conceptual view of the usage balancing process is shown in Figure 71.

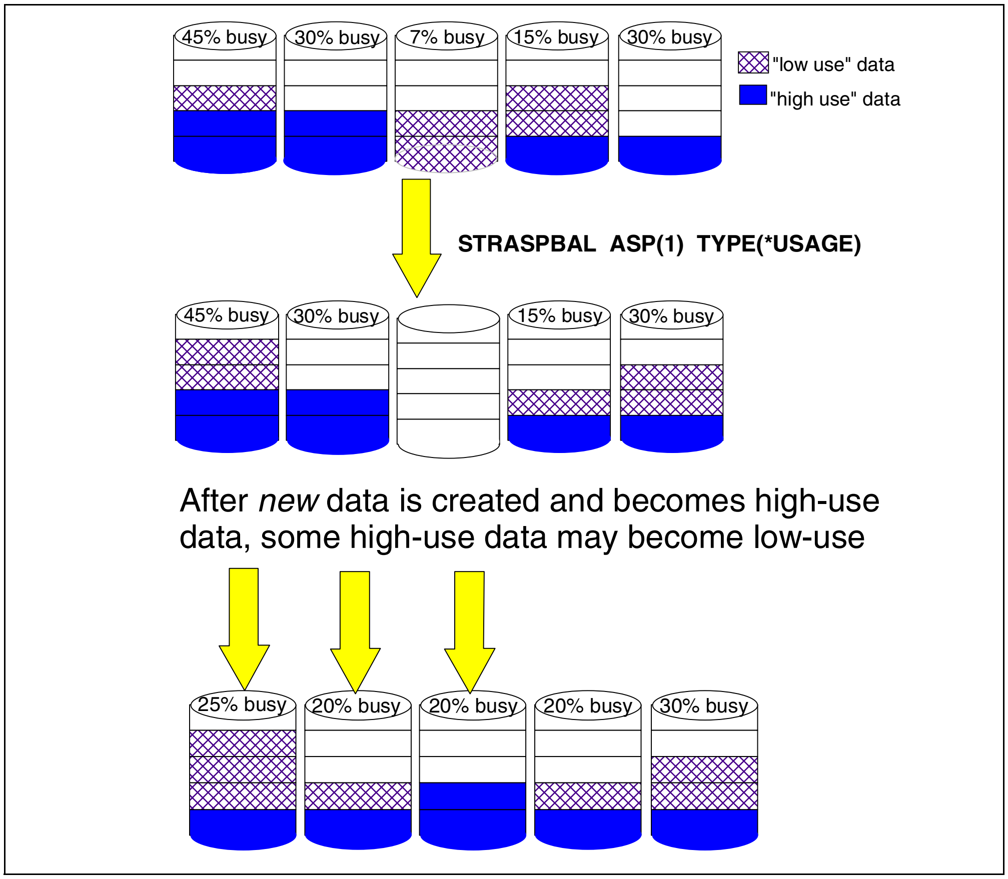


Figure 71. Usage balance: A conceptual view

- **Hierarchical Storage Management (HSM) balancing**: Hierarchical Storage Management balancing requires there be *compressed and uncompressed disk units in the same user ASP*. The system ASP is not eligible for compression. As such, HSM balancing moves high used data to faster (uncompressed) disk units, and lesser used data to slower (compressed) units.

  A conceptual view of the HSM balancing process is shown in Figure 72 on page 104.

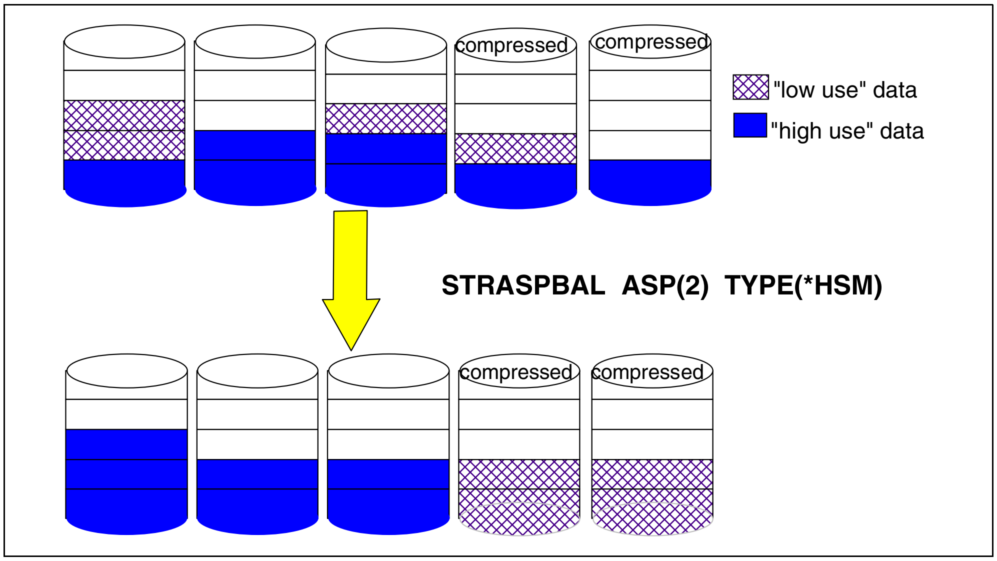


Figure 72. HSM balance: A conceptual view

With HSM balancing, the system determines the amount and location of the high-use and low-use data on the disk units. The system then redistributes the disk unit data to prevent accumulation of the high-use data on too few disk units. With HSM balancing, the user ASP should contain a sufficient amount of uncompressed disk space to hold all the high-use data that HSM balancing moves onto. Otherwise, the goal might not be achieved.

Refer to the Web page `http://www.as400.ibm.com/beyondtech/hsm.htm` and the redbook *Complementing AS/400 Storage Management Using Hierarchical Storage Management APIs*, SG24-4450, for more information on HSM usage.

Note

All three types of ASP balancing in V4R4 have restrictions on what object types or locations can be moved. Situations or object types which are restricted from movement by a balancing function include:

- The first gigabyte of the load source unit
- Storage Management directories area
- Journals and journal receivers
- Temporary objects
- Objects that are currently being used, or pinned in main storage (for caching purpose as an example)

This means that high-use data which falls into these categories *remains where it is on disk storage* when the ASP balance operation is in action.

Reduce the amount of unmovable objects to a minimum by running the ASP balance function during off-hours or in a restricted state. A restricted state is initiated with the `ENDSBS *ALL *IMMED` command.

7.2.2 Recommendation for using balance commands

For an effective use of disk balancing, follow these recommendations:

- The disk balance function is not intended to be a one-time operation, but rather as an element in a series of operations to achieve balance over time.
- Collect usage statistics over a reasonable time period under proper system workload.
- Start usage and HSM balancing *only after* the TRCASPBAL process finishes.

Overhead of balance commands

Trace ASP Balance (TRCASPBAL) tracing is separate from that done by Start Performance Monitor (STRPFRMON). TRCASPBAL maintains an IO count that maps a strip. Each disk unit is divided into many strips. This table of IO counts is small and resides in mainstore. This trace is part of the IO path. It adds very few instructions to the IO path and does not contribute much to the overall overhead of an IO. Tests with the trace on in the various benchmarks in the Lab showed no impact.

The STRASPBAL is where you may notice an impact. Therefore, it should be run in low activity periods.

7.2.3 TRCASPBAL command

The Trace ASP Balance (TRCASPBAL) command collects statistics that can later be used by *USAGE and *HSM balance processes to rearrange the disk allocation of data. This helps maintain efficient disk utilization.

During the trace period, the statistics of disk usage are collected in the form of:

- I/O requests count of all data stripes (1 MB in size) on all disk units in a specified ASP
- A utilization level of each disk unit in a specified ASP

This activity helps the system to identify high-use and low-use pieces of data, as well as to note how busy each disk unit is.

The TRCASPBAL command parameters are shown in Figure 73.

```
                         Trace ASP Balance (TRCASPBAL)

Type choices, press Enter.

Auxiliary storage pool ID  . . . > *ALL          1-16, *ALL
               + for more values
Trace option setting . . . . . . > *ON           *ON, *OFF, *CLEAR
Time limit . . . . . . . . . . .   *NOMAX        1-9999 minutes, *NOMAX
```

Figure 73. TRCASPBAL command

Tracing can be started for a specific ASP or for multiple ASPs. It can also be started for a specific length of time by specifying a value for the Time limit, or stopped at any time, by specifying TRCASPBAL SET(*OFF), and then restarted at a later time.

The collected statistics are *cumulative*. If the trace is started, ended at a later time, and then restarted without clearing the statistics, the second collection of statistics are added to the first group.

Be careful when using TRCASPBAL. Do not mix the trace data gathered from a very low system workload period with data from a very high system workload period. This generates disk usage statistics which, when used by a usage balance process to select which piece of data to move, produces less efficient disk balancing.

Use TRCASPBAL SET(*CLEAR) to clear the irrelevant trace data before starting a new trace for the next usage or HSM balancing process. Also clear the traced statistics if usage or HSM balance functions are partially completed (that is, they do not reach completion). Start a new trace.

The TRCASPBAL process does not run as a batch job. It runs as low level tasks below the AS/400 Machine Interface. Use the Work System Activity (WRKSYSACT) command to monitor tasks with the names of SMIOCOLLECT and SMTRCTASK. WRKSYSACT is available when the Performance Tools/400 Licensed Program Product (5769-PT1) is installed.

Messages are sent to the system history log (QHST) when the trace function starts, ends, and when trace statistics are cleared, as shown in Figure 74.

```
Message ID . . . . . . :   CPI1476       Severity . . . . . . . :   10
Message type . . . . . :   Escape
Date sent  . . . . . . :   10/07/99      Time sent  . . . . . . :   09:00:00

Message . . . . :   ASP tracing successfully started for ASP 1.
Cause . . . . . :   The ASP tracing function successfully started for ASP 1.
  The tracing function will run until the time limit expires or the function
  is stopped by the user.


Message ID . . . . . . :   CPI1477       Severity . . . . . . . :   10
Message type . . . . . :   Escape
Date sent  . . . . . . :   10/07/99      Time sent  . . . . . . :   09:50:00

Message . . . . :   ASP tracing for ASP 1 ended.
Cause . . . . . :   The ASP tracing function for ASP 1 ended.  The function
  had an ending code of  1.  For more information about the ending codes and
  possible recoveries see the Hierarchical Storage Management guide. The
  ending codes and their meanings are as follows:
    1 - The time limit specified for the tracing has expired.
    2 - The user requested the tracing be ended.
    3 - The tracing function could not be started.
    4 - The tracing function could not be started on an ASP that consists of
  only a single disk unit.
    5 - The tracing function ended unexpectedly.


 Message ID . . . . . . . . . :   CPI1478
 Message file . . . . . . . . :   QCPFMSG
   Library  . . . . . . . . . :     QSYS

 Message . . . . :   ASP tracing data successfully cleared for ASP 1.
 Cause . . . . . :   ASP tracing data has been successfully cleared for ASP 1.
```

Figure 74. Trace ASP Balance messages in the OS/400 history log

7.2.4 STRASPBAL and ENDASPBAL commands

After disk usage statistics are collected for a reasonable amount of time during a period of proper system utilization, the ASP is then balanced using the Start ASP Balance (STRASPBAL) command. Specify the parameter TYPE with a value of *USAGE or *HSM. After the balance function completes, the statistics are automatically cleared.

STRASPBAL TYPE(*CAPACITY) can run at any time because it does not need disk usage statistics for its operation. The STRASPBAL command parameters are shown in Figure 75.

```
                         Start ASP Balance (STRASPBAL)

Type choices, press Enter.

Auxiliary storage pool ID  . . . > *ALL          1-16, *ALL
              + for more values
Balance type . . . . . . . . . . > *CAPACITY     *CAPACITY, *USAGE, *HSM
Time limit . . . . . . . . . . . > *NOMAX        1-9999 minutes, *NOMAX
```

Figure 75. STRASPBAL command

Recommendation

Run the usage or HSM balance functions *soon after* the TRCASPBAL has finished collecting statistics. This ensures that the statistics apply directly to the data access characteristics of the system workload. Running the balance processes after leaving the statistics for too long may not be applicable to your system *if there are substantial changes* in the data access characteristics during this period.

In general, run STRASPBAL in the off-hours of the same day that you run TRCASPBAL. If the statistics become useless because the data access characteristic of your workload has changed in a noticeable way, clear them by running TRCASPBAL SET(*CLEAR).

If you run the usage or HSM balance process but cannot see any tangible results when it finishes, this can mean that the trace statistics do not contain any substantially observable data for the balance process to take action. For example, your system may not have high-use data at all, or the workload is too light during the trace period. Clear the trace and run it again during a more representative time period.

ASP balance processes run as low level tasks, below the AS/400 Machine Interface. The task names are SMDASDUTILTASK, SMBALUNIT, and SMEQnnnn, where *nnnn* is the ASP number the task is working on. Messages are sent to the system history log (QHST) when the ASP balance process starts and ends, as shown in Figure 76 on page 108.

```
Message ID . . . . . . :   CPI1474       Severity . . . . . . . :   10
Message type . . . . . :   Escape
Date sent  . . . . . . :   10/07/99      Time sent  . . . . . . :   09:08:27

Message . . . . :   ASP balancing successfully started for ASP 1.
Cause . . . . . :   The ASP balancing function of type *CAPACITY successfully
  started for ASP 1. The balance function will run until the time limit
  expires, the ASP has been completely balanced, or the function is stopped by
  the user.


Message ID . . . . . . :   CPI1475       Severity . . . . . . . :   10
Message type . . . . . :   Escape
Date sent  . . . . . . :   10/07/99      Time sent  . . . . . . :   09:45:21

Message . . . . :   ASP balancing for ASP 1 ended.
Cause . . . . . :   The ASP balancing function of type *CAPACITY for ASP 1
  ended.  The function had an ending code of  1.  For more information about
  each ending code and possible recoveries see the Hierarchical Storage
  Management guide. The ending codes and their meanings are as follows:
    1 - The ASP is completely balanced.
    2 - The time limit specified for the balancing has expired.
    3 - The user requested the balancing be ended.
    4 - The balance function could not be started.
    5 - The balance function could not be started on an ASP that consists of
  only a single disk unit.
    6 - The balance function ended unexpectedly.
   7 - The balance function could not be started on an ASP with no trace
  information.
    8 - The balance function could not be started on an ASP that does not
  contain the correct mixture of high-performance and low-performance disk
  units(for example compressed disk units).
```

Figure 76. ASP Balance messages in OS/400 history log

Specify a time limit for the ASP balance function to run, or let it run to completion. If the ASP balance function runs for a certain period of time and then stops, it continues from where it left off when restarted at a later time. This allows the balancing to run outside of normal office hours, and over a period of several days, to prevent a negative system performance impact during normal office hours.

Use the End ASP Balance (ENDASPBAL) command to end the ASP balancing function before the specified time limit expires, or when *NOMAX is used for the "Time limit" attribute.

By default, you must have *ALLOBJ special authority to use these balance commands.

Find additional information on ASP balancing in *Hierarchical Storage Management*, SC41-5351.

7.2.5 Integrated File System and HSM Balance

The ASP balance tool works with disk storage on an ASP basis. This means that all kinds of data located in the ASP (which include Integrated File System's various file systems, such as the QSYS and QDLS file systems) gain benefits from the balance tool. However, the HSM balance function works only with *user ASPs*, because the system ASP does not support disk compression. Since *all* *stream files in the Integrated File System are stored in the system ASP only*, move them to a user ASP for the HSM balance to include this data.

To put stream files housed in the Integrated File System into a user ASP, use OS/400 User-Defined File System (UDFS). Refer to Chapter 5, “Extending the Integrated File System with UDFS” on page 73, for more information on UDFS.

Note: The Make Directory (MD) command does not work to place Integrated File System information in a user ASP because MD creates directories in the system ASP only.

7.2.6 DSKBAL command

OS/400 V4R3, V4R2 and V4R1 do not have the TRCASPBAL nor STRASPBAL commands. Capacity balancing is available using the Disk Balance (DSKBAL) command. Usage and HSM balancing are not options.

DSKBAL is packaged as Program Temporary Fixes (PTFs). The OS/400 PTFs are:

- SF48378 for V4R1
- SF48479 for V4R2
- SF49675 for V4R3

Associated microcode PTFs are:

- MF18863 and MF17616 for V4R1
- MF19026 and MF18676 for V4R2
- MF19032 for V4R3

These PTFs should be included in the latest cumulative PTFs packages of each release.

The disk capacity balance function using the DSKBAL command is performed during an IPL (both normal and abnormal IPLs), right after storage management directory recovery. The DSKBAL command is shown in Figure 77.

```
                                 Disk Balance (DSKBAL)

 Type choices, press Enter.

 Number of minutes 0-9999 . . . .   *CHECK        0-9999, *STOP, *CHECK, *SAME
 ASP to balance . . . . . . . . .   *ALL          1-16, *ALL
 Balance how often:
   Every IPL, or once . . . . . .   *EVERYIPL     *ONCE, *EVERYIPL
   Minutes each sucessive IPL . .   30            0-9999
```

Figure 77. DSKBAL command parameters

The disk capacity balancing function is restricted to run *at IPL time only*. You can specify that it run only once at the following IPL, or run at every IPL until the balancing is complete.

Specify a time duration for the function to run in minutes. While the function runs at IPL, a System Reference Code (SRC) of C6004257 is displayed on the control panel. The code runs until the percent used on the drives specified is equal, or the time specified has elapsed.

When the code terminates, it creates an LIC log entry VL100020C5. The entry contains the control block for the command and the following message, which summarizes the execution:

- SUCCESSFUL — We completed in less time than was specified.
- TIME EXPIRED — We terminated because we had run for the length of time specified.
- MOVE EXTENTS ERROR — We received this message.

Stop the DSKBAL function by typing:

```
DSKBAL BALTIM(*STOP)
```

Recommended run times

The time necessary to move existing data depends on a number of factors. The most critical factor is the system configuration, including the type and size of existing DASD, the number of IOPs, etc. The current data allocation sizes also affect the time necessary to move the data. Given this, the results of the following formula should be used only as a guideline to how long it will take to balance the drives.

First, it is necessary to determine how much data will be moved. For the ASP to be balanced, multiply the size of the ASP (before the new DASD has been added) by the percent used. This is the amount of allocated storage. Divide this number by the size of the ASP after the new DASD is added. This is the percent used that the ASP will be after the new DASD is added. Multiply this number by the size of the new DASD. This is the amount of data to be moved.

In IBM tests, it was found that it took one minute to move 10 MB of data. Dividing your last result above (assuming the amount of data to move was in MB) by ten yields the approximate number of minutes it will take to balance the ASP.

Consider this example. The system is one ASP system that appears as shown in Figure 78 after the new DASD has been added.

```
                         Display Disk Configuration Capacity

                                                       ----Protected---  ---Unprotected---
ASP   Unit  Type  Model Threshold Overflow   Size     %Used      Size    %Used
 1     1    6606   030     95%       No        0       0.00%      5062    54.59%
                                               0       0.00%      1967    68.45
```

Figure 78. Display Disk Configuration Capacity

IBM recommends that you let the DSKBAL command run as long as you are able. If it has not finished, create “dummy” files to even out the percent used of the new DASD. When there is time to run the command again, delete the dummy files and run the DSKBAL command as long as possible.

7.3 Management Central disk resource utilization monitor

Management Central provides a real-time system resource monitor function to keep track of several disk-related resources, including the management of disk space. Management Central provides details on the resource you choose to monitor. This section discusses the disk resource utilization monitor function of Management Central:

Note: The functions we explain apply to OS/400 V4R3 or later. On the PC side, use Operations Navigator from the Client Access for Windows 95/NT V3R2 install CD, or a newer release. On a V4R3 system, call the program QSYS/QYPSSTRS to start the server for Management Central before proceeding. On a V4R4 system, use the Start TCP Server (STRTCPSVR) command to start the server component of Management Central. Specify `*MGTC` for the SERVER parameter, as:

```
STRTCPSVR SERVER(*MGTC)
```

Follow these steps to set up the disk resource utilization monitor:

1. Invoke the Operations Navigator and supply sign-on information if necessary.
2. Click the plus sign (+) on the left of the line to expand the **Management Central** line. The expanded option is shown in Figure 79.

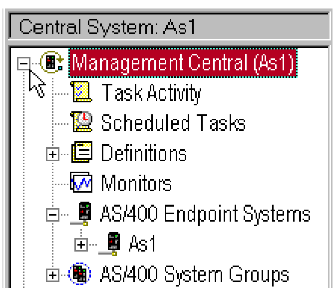


Figure 79. Expanded Management Central function group

The system to be monitored must be defined as an endpoint system before you can create a monitor profile for it.

> **Note**
>
> The endpoint system must have a communication link to the central system and must run OS/400 V4R3 or later. For systems with V4R2 or earlier, install the PRPQ named Performance Investigator for OS/400 (product number 5799-RFD) to use Management Central functions.

3. If there is no endpoint system created for your target system, right-click the **AS/400 Endpoint Systems** line to invoke a pop-up menu. Select **New Endpoint System...**. The New Endpoint System window appears (Figure 80 on page 112).

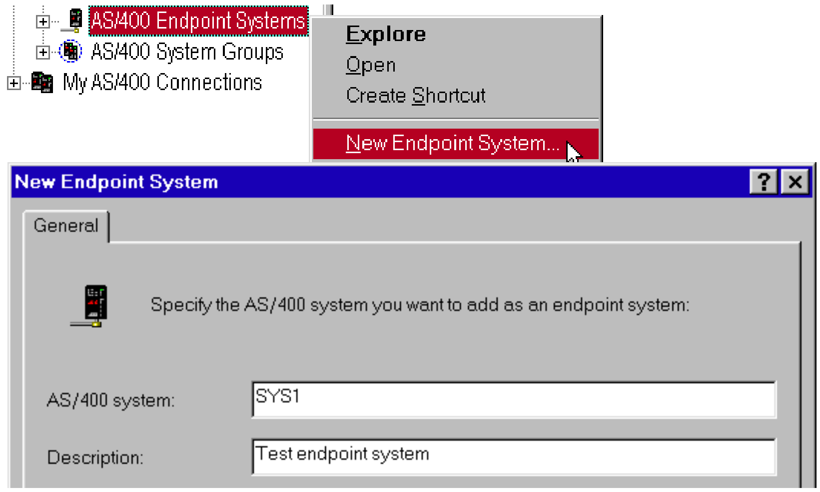


Figure 80. Creating a new endpoint system

4. Specify the name of the intended system in the New Endpoint System window. Also specify a brief description to describe this system as an endpoint system in a list of systems that exist. Click **OK** to finish creating a new endpoint system. Notice that the line AS/400 Endpoint Systems can be expanded with the newly added system name shown underneath.

Note

When you click **OK** in this step, Management Central checks for the existence of the endpoint system name specified. It searches the \WINDOWS\HOSTS file in your PC or in the Domain Name Server (DNS) specified in Windows TCP/IP configuration. If it cannot locate the system, a notification message is sent.

Make sure the endpoint system name exists in either the PC HOSTS file or the DNS.

5. Move the mouse pointer to the **Monitors** icon, and right-click. Select **New Monitor...** to invoke the New Monitor window as shown in Figure 81.

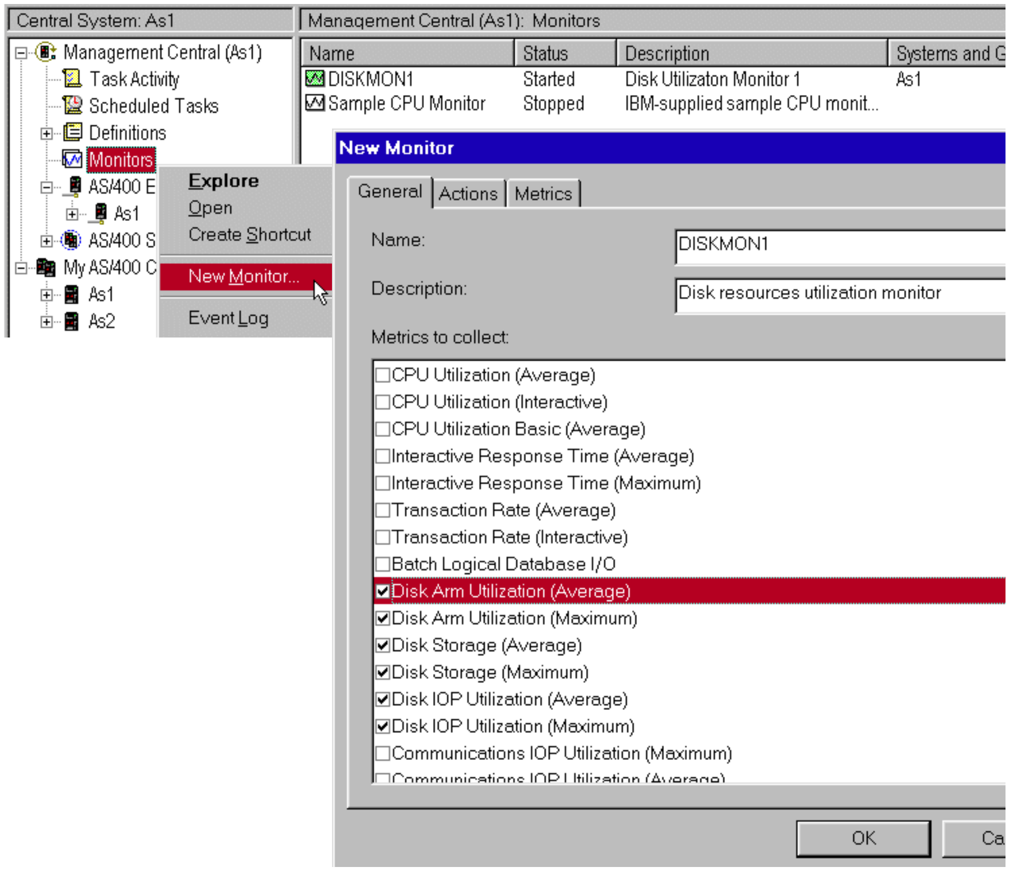


Figure 81. Creating a new monitor profile

6. On the **General** tab of the New Monitor window, specify the name for the new monitor profile to be created and type in the description.

   The Metrics to collect list box contains many resource utilization categories available for selection. There are six categories relating to disk utilization:

   - Disk Arm Utilization (Average)
   - Disk Arm Utilization (Maximum)
   - Disk Storage (Average)
   - Disk Storage (Maximum)
   - Disk IOP Utilization (Average)
   - Disk IOP Utilization (Maximum)

   Click the check boxes in front of the items to select the items to monitor.

   To see the overall utilization of the disk resources over a period of time, choose the *Average* category. This category is used to verify a uniform distribution to all individual components of the category.

   To identify the highest utilization of an individual component at a specific moment, select the *Maximum* category. The objective is to see the Average and the Maximum at an equal level. This indicates a consistent utilization of the resource throughout. A *large difference between the average and the maximum* can indicate an inconsistent utilization over time, or among the individual components. The larger the difference, the more the inconsistency.

7. On the **Actions** tab, select **threshold trigger** and **threshold reset** actions.

8. Select the **Metrics** tab (Figure 82) to specify parameters to control the resource utilization categories selected in the General tab.

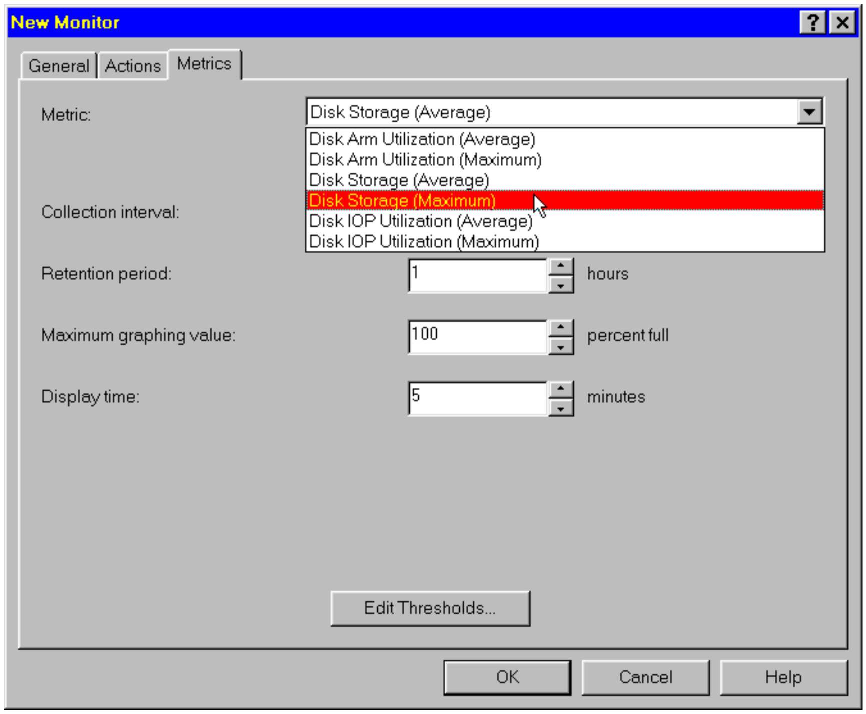


Figure 82. New Monitor window: Metrics tab

Matrix options include:

- **Collection interval**: Use the collection interval to control how often statistics are collected.
- **Retention period**: Use the retention period parameter to determine how long to keep the collected utilization statistics in the central system. This value determines how much space is allocated in the *disk storage of the central system* to hold the statistics. The longer the period, the larger the disk allocation.
- **Maximum graphing value**: The maximum graphing value controls the vertical dimension of the graph window. It is measured as a range of percentage of statistics seen at any time.
- **Display time**: The display time parameter controls the horizontal dimension of the graph window. It displays the range of statistics seen at any time.

9. To edit the threshold trigger for the reset actions taken when utilization reaches a certain level, click **Edit Thresholds** (as shown in Figure 83). Specify the threshold level and action to be taken. Click **OK** when done.

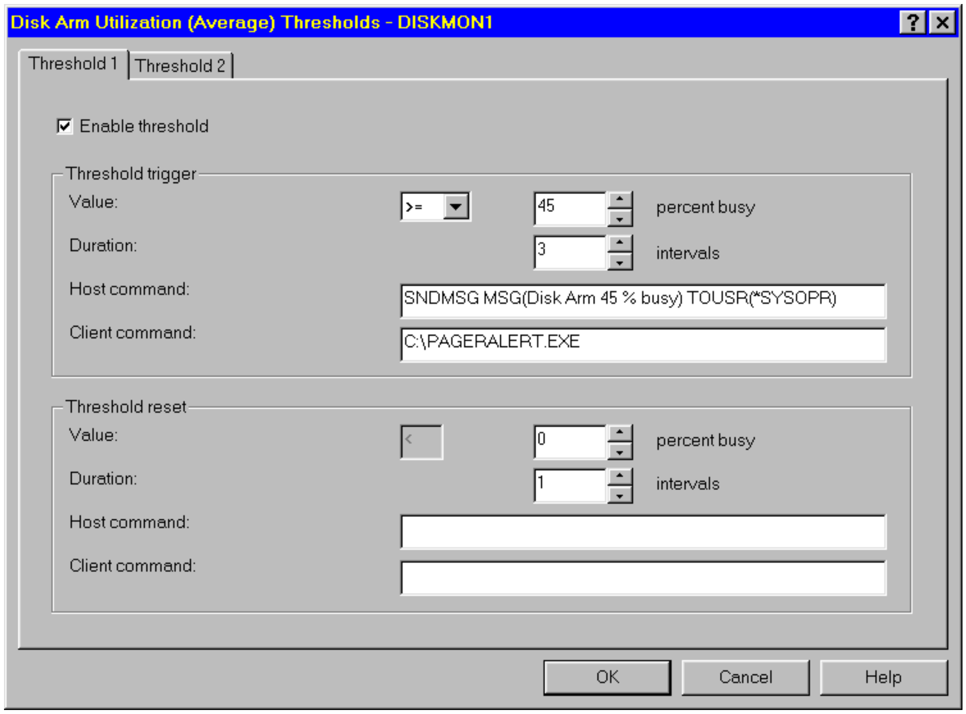


Figure 83. Edit Thresholds window

10.In the New Monitor windows, click **OK** to create a new monitor profile. The new monitor profile is the name which you specified in Figure 81 on page 113. It appears in the right side of the window.

11.Move the mouse pointer to the new monitor profile just created and right-click to bring up the pop-up menu. Select **Start** to begin real-time monitoring of the selected resource categories (Figure 84). You are prompted to select the endpoint system to be monitored.

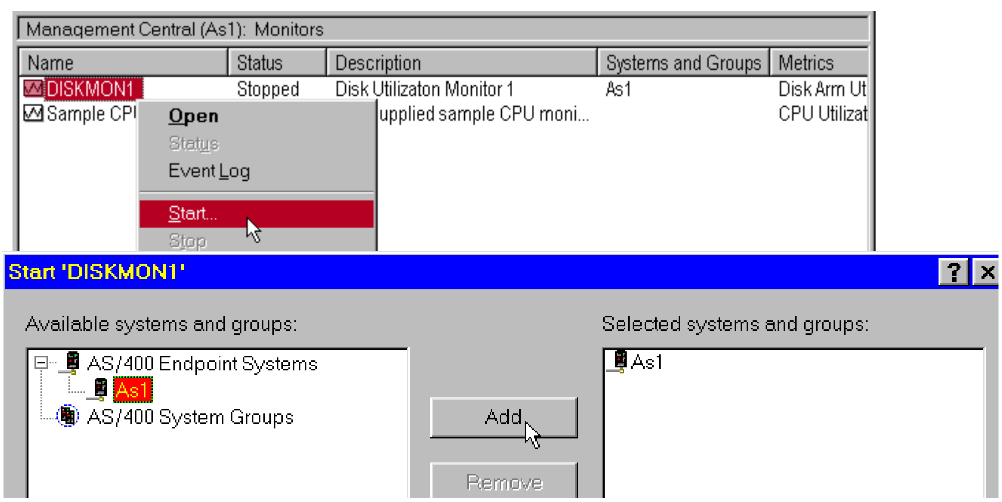


Figure 84. Starting resource utilization monitor

12.Select the system, and click **Add**. Click **OK** when done.

13.Press F5 key to refresh the display. Notice that the status of the monitor profile changes from stopped to starting and then to started.

14.To display the real-time resource utilization monitor window, move the mouse pointer to the monitor profile name and right-click to bring up the pop-up menu. Select **Open**. You see information windows similar to those shown in Figure 85.

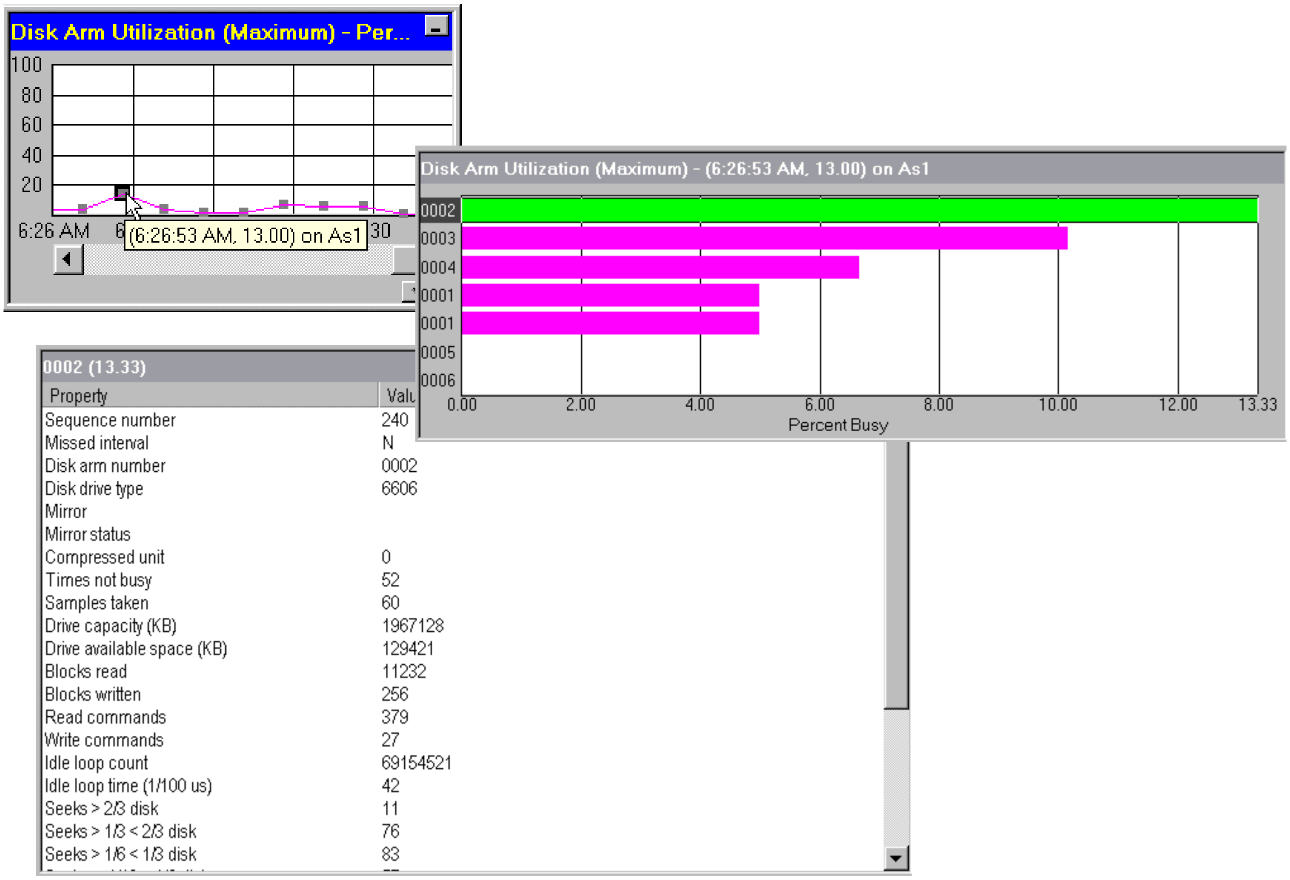


Figure 85. A real-time resource utilization monitor windows

The information window displays three levels of sampled data:

- The first level of data is on the left side of the window. Shown as a line graph, it depicts the average or maximum utilization collected over time.
- Move the mouse pointer to a point of interest in the first level data. Click to see the second level of detail on the top half of the right side of the window. Shown as a bar graph, it displays the utilization of each individual component in descending order.
- Click an individual bar of interest. The third level of detail shows in the bottom half of the window. This table lists the properties of the individual component collected at that specific point in time.

  Move the mouse pointer to the property line of interest. Right-click, and select **What's this** to display an explanation on what that property is.

In the second level information window (as shown in Figure 85), you can see there are a total of seven disk units in our sample system. There are two units with the same disk arm number of 0001. These are in the mirrored pair of the Load Source Unit (LSU). The LSU assumes disk number 0001 in every system.

At that particular point in time, unit number 0002 has the highest percent busy of about 13%. Unit numbers 0003, 0004, and 0001 appear in descending order of utilization. Disk units 0005 and 0006 appear to be idle.

The third level information is about disk number 0002 in particular because it is the most utilized unit at that moment (13.3%). In our example, it has read 11,232 blocks, written 256 blocks, executed 379 read commands, 27 write commands, and so on.

Use this third-level information to analyze performance-related information or the availability-related status of each of the individual components. This helps the administrator identify if further action should be taken to rectify any issues uncovered from this report.

15. Close the real-time monitor window. To display it again later, keep the monitor running.

    Note: If you leave the monitor running continuously, be aware that the amount of AS/400 disk space used by the monitor depends on the length of time you specify as retention period as shown in Figure 82 on page 114.

16. To end the monitor, right-click the monitor profile icon as shown in Figure 84 on page 115, and select **Stop**.

Management Central is a user-friendly, graphical interface to help monitor disk resources utilization. Refer to *Management Central: A Smart Way to Manage AS/400 Systems*, SG24-5407, for detailed information on how to use many functions delivered in Management Central.

7.4 Disk reorganization tool

To consolidate the unused space into one physical area, use the Start Disk Reorganization (STRDSKRGZ) and End Disk Reorganization (ENDDSKRGZ) commands. They handle the task to rearrange unused blocks of disk space so that the empty chunks of disk space are *consolidated into one physical contiguous area*. It does not attempt to “defrag” objects in the used space of disk storage.

Note

There is a tool shipped as a PTF that does the equivalent of the STRDSKRGZ command. The difference is that the action occurs during the next IPL. The command is called DEFRAG.

Refer to Figure 86 on page 118 for a conceptual view of the disk reorganization function.

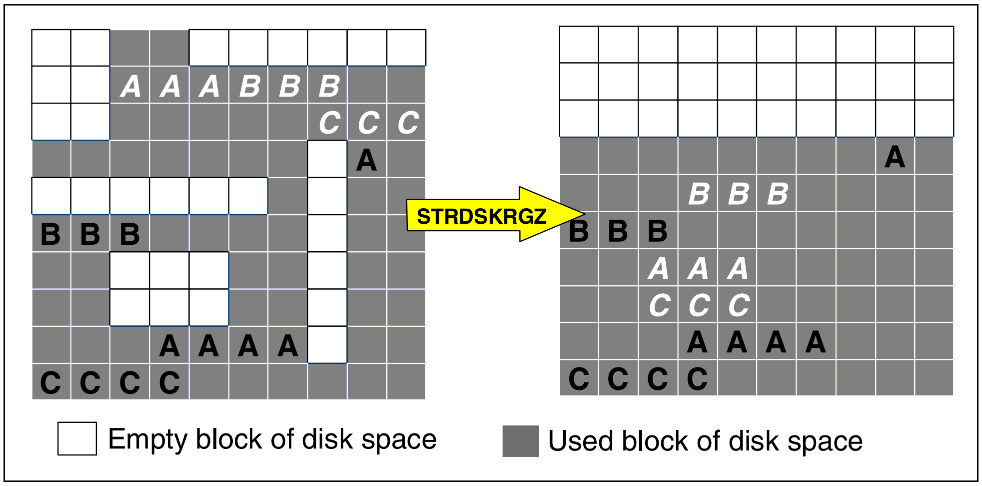


Figure 86. Disk reorganization: A conceptual view

This tool helps accommodate the future allocation of large objects on the disk space such that the allocated objects are not scattered into multiple small chunks of disk space. As such, a negative performance impact for data accesses is lessened.

An example where disk reorganization is useful is for an Electronic Documents Management application, such as Imageplus WAF/400. This type of application migrates *rarely-used* electronic documents from disk to optical storage, freeing up space for new documents. Typical electronic documents are 5 to 100 KB in size. When a large number of these documents are removed from disk storage, a lot of small empty disk chunks are left scattered throughout the ASP. Gathering the empty disk chunks into one area improves performance and storage efficiency.

Using the Hierarchical Storage Management (HSM) APIs (available in OS/400 in V4R3 or later), automatic migration or archival of objects can leave free space scattered on the disk units. The disk reorganization tool is useful to consolidate the free space into one area for this application, as well.

Refer to `http://www.as400.ibm.com/beyondtech/hsm.htm` on the Web and the redbook *Complementing AS/400 Storage Management Using Hierarchical Storage Management APIs*, SG24-4450, for further information on HSM and these APIs.

7.4.1 Using the STRDSKRGZ command

The STRDSKRGZ command has two parameters:

- **Time limit**: Use the time limit parameter to specify how long the disk reorganization task runs. Specify `*NOMAX` to run disk reorganization as long as it needs to reorganize the disk units. Disk reorganization stops when the specified units are reorganized, until the specified time limit is reached, or until an ENDDSKRGZ is issued.
- **ASP**: Use the ASP to specify the ASP to reorganize. Specify `*ALL` to reorganization all ASPs in the system.

STRDSKRGZ can run during a normal system operation, however, off-hour operation is preferred.

The reorganization process runs as a low level background task. It is named SMDEFRAGTA by System Licensed Internal Code (SLIC).

Use the `ENDDSKRGZ` command to stop the disk reorganization task at any time before its run-time limit is reached. An all-object special authority is necessary on the user profile to run this command. Specify SPCAUT(*ALLOBJ) on the user profile running the job.

Messages CPI1472 and CPI1473 are sent to the system history log (QHST) to indicate the start and end of the reorganization function for each ASP. Refer to Figure 87 for an example.

```
 Message ID . . . . . . :   CPI1473       Severity . . . . . . . :   00
 Message type . . . . . :   Information
 Date sent  . . . . . . :   10/07/99      Time sent  . . . . . . :   21:16:41

 Message . . . . :   Disk reorganization started for all configured ASP's.
 Cause . . . . . :   Disk reorganization has been started for all ASP's
   configured on the system. It will run until the time limit has expired, the
   ASP has been completely reorganized, or the function is stopped by the user.


Message ID . . . . . . :   CPI1472       Severity . . . . . . . :   00
 Message type . . . . . :   Escape
 Date sent  . . . . . . :   10/07/99      Time sent  . . . . . . :   21:16:43

 Message . . . . :   Disk reorganization for ASP 1 ended.
 Cause . . . . . :   The disk reorganization function for ASP 1 has been ended
    The function had ending code 5. The ending codes and their meanings are as
   follows:
     1 - The ASP is completely reorganized.
     2 - The time specified for reorganization has expired.
     3 - The user requested that reorganization be ended.
     4 - The reorganization  function could not be started.
     5 - The reorganization function cannot be started on an ASP that consists
         of only one unit.
     6 - The reorganization function ended unexpectedly.
```

Figure 87. Disk reorganization messages in OS/400 history log

Note: The disk reorganization commands are available from OS/400 V4R2 onward.

7.5 Automating disk management commands

There are job scheduling system utilities to automate the process of disk management, and schedule the tasks to run in an off-peak period to lessen the impact to system performance.

To set up a scheduled (daily, weekly or monthly) ASP balance or disk reorganization process to run during after-production time, several utilities are available. These include:

- The Work With Job Schedule Entries (WRKJOBSCDE) command. Use WRKJOBSCDE to set up daily, weekly, or monthly schedules for a single AS/400 system.
- Management Central - Command Definition. Management Central is used to schedule running *the same* processes for *multiple* AS/400 endpoint systems at the same time. To support this function, all endpoint systems must have communication links to the same central system. Daily, weekly, or monthly schedules are supported.
- Use Job Scheduler for AS/400 (5769-JS1) Licensed Program Product for more complex scheduling (for example, to schedule an event that is contingent upon the completion of another event).

Refer to *AS/400 Work Management Guide,* SC41-5306, for additional information on WRKJOBSCDE.

Refer to the redbook *Management Central - A Smart Way to Manage AS/400 Systems*, SG24-5407, for more information on Management Central.

Refer to *AS/400 Job Scheduler,* SC41-5300, for additional information on the Job Scheduler licensed program.

7.6 Web sites on the Internet for system management

The following Web sites are good sources of updated information on AS/400 system management.

- AS/400 Information Center: `http://www.as400.ibm.com/infocenter`
- AS/400 Technical Studio: `http://www.as400.ibm.com/techstudio`
- AS/400 Beyond Technology Page: `http://www.as400.ibm.com/beyondtech`

Bookmark these sites for easier access, using your Web browser.

Chapter 8. Runaway DASD

As modern application software packages become more and more sophisticated, they demand more and more processor power and disk space to support advanced functionality. Recent studies show that a typical AS/400 customer increases their DASD consumption by 30% per year. Therefore, a good DASD management strategy forms an important part of the overall AS/400 system management plan to protect a growing company's asset.

IBM recommends that you maintain DASD utilization at less than 70 to 75% for optimal performance. However, from time to time many systems exceed this guideline.

Examples of when this occurs are:

- A new version of the application software package consumes more disk space than the prior release.
- The business experiences an unexpected upturn in sales.
- Budget constraints prevent the installation of additional disk resource to handle expected growth.
- There is improper planning or the lack of a good DASD maintenance or management strategy.

Whatever the reason for the increase in DASD usage, whether it is business related, planned, or unplanned, it is important that you are aware that this growth impacts the performance of your system. This chapter outlines tools available to analyze the cause for the growth.

Note

The information described in this chapter is based on OS/400 Version 4 Release 4 of OS/400 and is by no means exhaustive.

8.1 Disk and growth terminology

We start with some commonly used terminology used to describe disk topics and the management of disk growth.

- **Controlled growth**

  The most common cause of DASD consumption problems is natural business growth. An increase in business generates more transactions, which in turn require more disk space to store them. This is referred to as "controlled growth".

- **Uncontrolled growth**

  There are times when jobs consume more DASD than would usually be expected. For example, an application software bug causes a program to add a lot more records to a database file than it should. This growth is referred to as "uncontrolled growth".

© Copyright IBM Corp. 2000

- **Runaway DASD**

  In extreme cases, “Uncontrolled Growth” consumes DASD at such a rate that, unless stopped, causes an application or system outage. This uncontrolled growth is referred to as “Runaway DASD”.

8.2 Storage concepts

To understand why an AS/400 system can run out of storage, it is necessary to be familiar with some of the basic concepts of storage usage on the AS/400 system relative to data types and how they are stored:

- Data on the AS/400 system is stored either on hard disk (DASD) or in real memory.
- The AS/400 system uses a storage management architecture called Single Level Storage (SLS). SLS treats all addressable memory (either hard disk storage or real memory) as one addressable space. This allows great flexibility for application programs and users, and also increases the effective address space on the system.
- To survive across an IPL, data needs to be on DASD.
- There are two types of data from the point of view of storage on the AS/400 system: permanent and temporary.
  - Permanent is data that survives across an IPL if it is on disk (for example, database files).
  - Temporary is data that does not survive across an IPL. An example is the work space for a job that is terminated by an IPL. Temporary objects are not expected to be re-used.
- DASD on the AS/400 system can be divided into groups called Auxiliary Storage Pools (ASPs). There is always one ASP, known as the system ASP. There can be up to 16 defined ASPs: 15 known as user ASPs and the one system ASP.
- A user ASP has a finite amount of storage available to it, defined by the amount of DASD configured in it. When the amount of data in the ASP reaches 100% of the capacity available, new data directed to that ASP is re-directed to the system ASP instead. This data is called Overflow data.
- The % system ASP used field in the top right corner of the WRKSYSSTS screen (Figure 88) reflects the total of Permanent and Temporary storage used in the system ASP. If this value reaches 100%, the system stops, as the system ASP has nowhere to overflow.

```
                         Work with System Status x            ITSOSYS1
                                                      09/24/99  09:55:13
% CPU used . . . . . . . :       99.3    Auxiliary storage:
Elapsed time . . . . . . :   00:00:01      System ASP . . . . . . :     6.88 G
Jobs in system . . . . . :        423      % system ASP used  . . . : 96.6918
% addresses used:                          Total  . . . . . . . . :     9.83 G
  Permanent  . . . . . . :       .010      Current unprotect used :      487 M
  Temporary  . . . . . . :       .010      Maximum unprotect  . . :      494 M

Type changes (if allowed), press Enter.

System    Pool    Reserved    Max   -----DB-----  ---Non-DB---
 Pool   Size (M)  Size (M)  Active  Fault  Pages  Fault  Pages
   1      50.00     24.70   +++++     .0     .0   14.0   19.1
   2      61.25       .01      30     .0     .0     .7    5.1
   3        .75       .00       2     .0     .0     .0     .0
   4      16.00       .00       6     .0     .0    6.6    7.3

                                                                     Bottom
Command
===>
F3=Exit   F4=Prompt           F5=Refresh   F9=Retrieve   F10=Restart
F11=Display transition data   F12=Cancel   F24=More keys
```

Figure 88. Work with System Status display

How the 100% used mark is reached

DASD consumption problems due to a permanent storage issue include situations such as:

- Too much data is restored to the system from tape or optical
- Too much data is received over a communication link
- The overflow of a user ASP continues unchecked

DASD consumption problems due to a temporary storage issue includes situations such as when a large query continues to create more and more temporary storage to accommodate work files.

Note

One side effect of DASD utilization nearing the 100% mark is that System Network Architecture Distribution Services (SNADS) ceases to function. Messages and mail no longer flow in or out of the system.

SNADS ends because we can no longer be sure that there is sufficient storage available to store incoming data. This is one reason IBM recommends DASD consumption remains below a 70 to 75% threshold.

8.3 Detecting DASD consumption problems

There are several ways the system operator detects that the system has a disk consumption problem. The following sections cover a number of these indicators, including:

- An auxiliary storage threshold is reached.
- A rapid increase in disk storage is observed.

- Your system is slowing down.
- Your system has stopped

An example of each scenario is given and recovery procedures are discussed in order to help you determine what action to take next.

8.3.1 An auxiliary storage threshold has been reached

Most often the way the system operator is first made aware there is a disk consumption problem on the system is when an auxiliary storage threshold has been reached. When an auxiliary storage threshold is reached, the system sends a CPI099C warning message, indicating that a critical storage lower limit has been reached. See Figure 89 for an example.

CPI099C is sent to the System Operator message queue. The message acts as a “trigger” to the System Operator to take a close look at auxiliary storage usage. The system does not automatically address the concern.

```
                           Additional Message Information

Message ID . . . . . . :   CPI099C       Severity . . . . . . . :   90
Message type . . . . . :   Information
Date sent  . . . . . . :   09/17/99      Time Sent  . . . . . . :   10:20:15

Message . . . . :   Critical storage lower limit reached.
Cause . . . . . :   The amount of storage used in the system auxiliary storage
  pool has reached the critical lower limit value. The system will now take
  the action specified in the QSTGLOWACN system value: &5. The possible
  actions are:
    *MSG -- No further action is taken.
    *CRITMSG -- Message CPI099B is sent to the user specified by the
  CRITMSGUSR service attribute.
    *REGFAC -- A job is submitted to run the exit programs registered for the
  QIBM_QWC_QSTGLOWACN exit point.
    *ENDSYS -- The system is ended to the restricted state.
    *PWRDWNSYS -- The system is powered down immediately and restarted.
                                                                       More...
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details   F12=Cancel
F21=Select Assistance Level
```

Figure 89. System operator message

All AS/400 systems have a threshold that can be set for each ASP user and system. The auxiliary storage threshold for an ASP determines when the system warns you that the space allocated for the ASP has reached the point where action needs to be taken to keep disk usage at a level less than (or equal to) a user-defined limit.

The default value for an ASP threshold is 90%.

Note

The threshold value of the system ASP is changed by changing the Auxiliary Storage Lower Limit system value (QSTGLOWLMT). Use Dedicated Service Tools (DST) or System Service Tools (SST) to change the threshold for each of the fifteen user ASPs.

For more information on changing auxiliary storage thresholds, refer to 8.2.2, “How to Change the Storage Threshold for an Auxiliary Storage Pool”, in *AS/400 Backup and Recovery,* SC41-5304.

When the auxiliary storage threshold is reached, the system takes the action specified in the Auxiliary Storage Lower Limit Action system value (QSTGLOWACN). The alternatives are:

- ***MSG**: To send message CPI099C to QSYSMSG (if it has been defined) and the QSYSOPR message queue.
- ***CRITMSG**: To send critical message CPI099B to the user specified in the service attribute to receive critical messages. CPI099B indicates that a critical storage condition exists.
- ***REGFAC**: To submit a job to call the exit programs registered for the QIBM_QWC_QSTGLOWACN exit point.
- ***ENDSYS**: To end the system to a restricted state.
- ***PWRDWNSYS**: To power down the system immediately (a *IMMED option) and restart it.

Review these options and ensure that QSTGLOWACN contains the most appropriate setting for your installation.

Note: CPI099C is sent for all actions defined for QSTGLOWACN.

What to do next

If exceeding the threshold is not a surprise (you are monitoring disk capacity and have seen a gradual increase due to natural business growth, for example), the action to take depends on the urgency:

1. If the increase in storage does not require immediate attention, use the DISKTASKS menu detailed in the *System Operation Guide*, SC41-4203, to determine storage usage.

   Use the Disk Information Tools to manage disk capacity until a disk upgrade can be installed. Refer to 8.4.1, “Using disk information tools” on page 142, for a description of tools available.
2. If storage use requires prompt attention, check active jobs for any specific restore jobs or communication jobs receiving a large amount of data onto the system.
3. Check for queries consuming lots of temporary storage.

   To determine the growth of temporary storage, check the *Temp Storage* field under Run Attributes when working with such a job. See Figure 90 on page 126 for an example.

```
                         Display Job Run Attributes
                                                             System:   ITSOSYS1
Job:   QPADEV0003     User:   ITSCID20       Number:   032001

Run priority . . . . . . . . . . . . . . . . . . :   50
Time slice in milliseconds . . . . . . . . . . . :   5000
Eligible for purge . . . . . . . . . . . . . . . :   *YES
Default wait time in seconds . . . . . . . . . . :   30
Maximum CPU time in milliseconds . . . . . . . . :   *NOMAX
  CPU time used . . . . . . . . . . . . . . . . . :     9803
Maximum temporary storage in megabytes . . . . . :   *NOMAX
  Temporary storage used . . . . . . . . . . . . :     999580
Maximum threads . . . . . . . . . . . . . . . . . :   *NOMAX
  Threads . . . . . . . . . . . . . . . . . . . . :     1







Press Enter to continue.

F3=Exit   F5=Refresh   F9=Change job   F12=Cancel   F16=Job menu
```

Figure 90. Run attributes display

If you cannot isolate an offending job, then you may have a Runaway DASD problem.

Consider ending the system to a restricted state to control additional storage growth and to allow time for problem analysis.

Refer to 8.4, "Analyzing DASD consumption problems" on page 142, for other tools to help you find the source of the problem.

8.3.2 A rapid increase in disk storage has been observed

Disk storage on the AS/400 system is usually monitored by:

- Using the WRKSYSACT display
- Using the WRKSYSSTS display
- Using Management Central
- Using a third-party disk monitoring system

The following sections describe how to use the first three of these disk storage monitoring methods and suggest additional follow-up actions to take.

Note

If there is an error log posted to the Licensed Internal Code log related to a rapid increase in disk storage, the associated dump (from the VLOG entry) may provide further information.

8.3.2.1 Using the WRKSYSACT display

With Performance Tools for AS/400 (5769-PT1), "View 4" lists, at V4R4, the allocated and deallocated storage assigned to a job or task. if the "run away

DASD” job is a long running job, you can identify it from the WRKSYSACT display. For an example, see Figure 95 on page 131.

8.3.2.2 Using the WRKSYSSTS display

The Work with System Status display (as shown in Figure 91) shows a set of statistics depicting the current status of the system. One field displayed is the percentage of the system auxiliary storage pool (ASP) currently in use.

The statistical information is gathered during an identified time interval (shown as the elapsed time). The data can be updated by extending the measurement time interval (use F5), or it can be reset and a new time interval started to gather a new set of data (use F10).

```
                          Work with System Status                ITSOSYS1
                                                       09/24/99  09:55:13
% CPU used . . . . . . . :       99.3    Auxiliary storage:
Elapsed time . . . . . . :   00:00:01      System ASP . . . . . . :     6.88 G
Jobs in system . . . . . :        423      % system ASP used  . . :  _96.6918
% addresses used:                          Total  . . . . . . . . :     9.83 G
  Permanent  . . . . . . :       .010      Current unprotect used :      487 M
  Temporary  . . . . . . :       .010      Maximum unprotect  . . :      494 M

Type changes (if allowed), press Enter.

System     Pool    Reserved    Max   -----DB-----  ---Non-DB---
 Pool    Size (M)  Size (M)  Active  Fault  Pages  Fault  Pages
   1       50.00     24.70    +++++     .0     .0   14.0   19.1
   2       61.25       .01       30     .0     .0     .7    5.1
   3         .75       .00        2     .0     .0     .0     .0
   4       16.00       .00        6     .0     .0    6.6    7.3

                                                                       Bottom
Command
===>
F3=Exit   F4=Prompt            F5=Refresh   F9=Retrieve   F10=Restart
F11=Display transition data    F12=Cancel   F24=More keys
Critical storage condition reached.
```

Figure 91. Work with System Status display

8.3.2.3 Using Management Central

Management Central is a suite of graphical systems management functions. Management Central is an optionally-installable component of Operations Navigator, the strategic graphical user interface for IBM AS/400 systems. These functions are available with OS/400 Version 4 Release 3 (V4R3) and later releases.

Note

To get information about Operations Navigator, including how to set up Management Central, refer to the AS/400e Information Center.

The Information Center provides a path to AS/400 technical information using the power of browsers and online technology. You can access numerous pages of information within the Information Center, as well as link to a wide variety of Internet sites with crucial technical content.

If you prefer not to rely on the Internet, access the Information Center from the CD-ROM (publication number SK3T-2027 for the English version). It arrives with your system software. The Information Center is located on the Web at: `http://www.as400.ibm.com/infocenter`

One function within Management Central is the ability to monitor real-time system performance using detailed graphs to visualize what is going on with your system.

This DASD monitor function can be used to display Disk Storage Information, as illustrated in Figure 92.

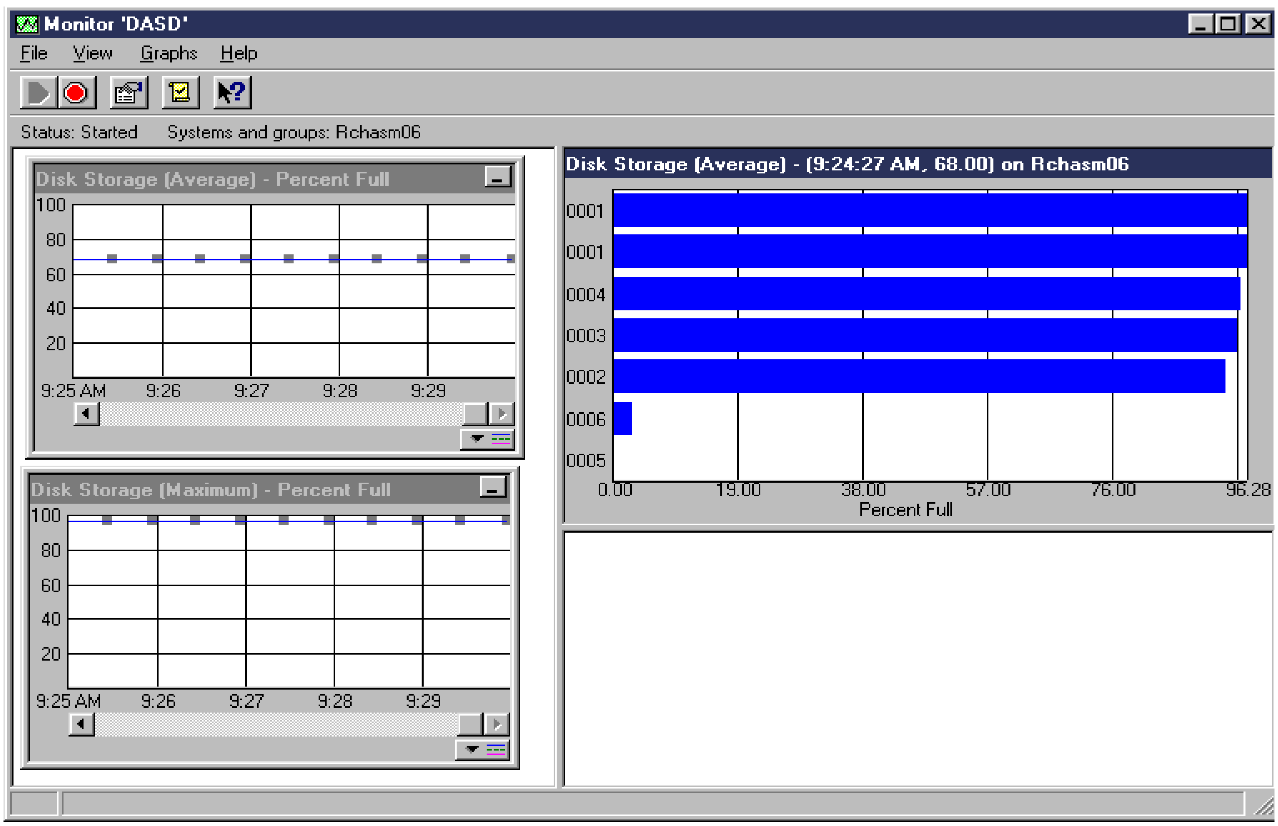


Figure 92. Management Central DASD Monitor

The Monitor DASD screen shows the average and maximum percentage of storage full on all disk arms during the collection interval.

Clicking on an individual time interval displays a chart for that interval. This detailed chart shows the percentage of storage full on each disk arm. Use this display to see how full your system is and whether all of the disk arms are evenly loaded.

8.3.3 What to do next

When using one of these disk storage threshold monitoring methods and a rapid increase in the percentage of disk storage used is noted, here are further suggested actions to take and possible causes:

1. Check the disk status display using the `WRKDSKSTS` command to confirm that the problem is system-wide. If the percent used of the disk units is unevenly distributed (as illustrated in Figure 93), and if you have multiple ASPs, a large data file or library may have been copied to the wrong ASP by mistake.

```
                              Work with Disk Status                     ITSOSYS1
                                                              09/24/99  09:28:49
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  6606   1967  96.6      .0        .0     .0     .0     .0     .0      0
   1  6606   1967  96.6      .0        .0     .0     .0     .0     .0      0
   2  6606   1967  96.6      .0        .0     .0     .0     .0     .0      0
   3  6606   1475  96.8      .0        .0     .0     .0     .0     .0      0
   4  6606   1475  96.7      .0        .0     .0     .0     .0     .0      0
   6  6606   1475   2.8      .0        .0     .0     .0     .0     .0      0
   5  6606   1475    .2      .0        .0     .0     .0     .0     .0      0







                                                                          Bottom
Command
===>
F3=Exit   F5=Refresh   F12=Cancel   F24=More keys
```

Figure 93. Work with Disk Status display

2. Use the System Service Tool (`STRSST`) command to verify your ASP configuration and usage, as illustrated in Figure 94 on page 130. Investigate that all user libraries are where they should be.

```
                        Display Disk Configuration Capacity

                                                      ----Protected--- ---Unprotected--
 ASP  Unit   Type Model  Threshold  Overflow              Size % Used        Size % Used
   1                           90%    No                 6884 96.94%           0  0.00%
        1    6606  030                                   1967 97.48%           0  0.00%
        1    6606  030                                   1967 97.48%           0  0.00%
        2    6606  070                                   1967 96.65%           0  0.00%
        3    6606  074                                   1475 96.81%           0  0.00%
        4    6606  074                                   1475 96.72%           0  0.00%
   2                           90%    No                 1475  2.82%           0  0.00%
        6    6606  074                                   1475  2.82%           0  0.00%
   3                           90%    No                 1475  0.21%           0  0.00%
        5    6606  074                                   1475  0.21%           0  0.00%






 Press Enter to continue.

 F3=Exit          F5=Refresh        F10=Display disk unit details
 F11=Display disk configuration protection  F12=Cancel
```

Figure 94. Display Disk Configuration Capacity

3. If you do not have multiple ASPs, or have performed the previous steps and still have a problem, then you may have a Runaway DASD problem. Refer to 8.4, “Analyzing DASD consumption problems” on page 142, for other tools to help you find the source of the problem.
4. Add non-configured disk units to the system ASP, or remove a unit from a user ASP. Then, add it to the system ASP.

8.3.4 Your system is slowing down

As the system starts to reach the high nineties of percentage disk utilization (especially system ASP usage), overall system performance can begin to suffer. The system also runs slower when the system ASP is highly fragmented (that is, free space is not in a contiguous area, but rather scattered across the ASP). Accounting for disk space occurs within the job, thread, and task in which task segments are to be created, extended, truncated, or destroyed.

With the WRKSYSACT command (which is part of the Performance Tools Licensed Program Product, 5769-PT1), you can display the storage allocated and allocated on a job or task basis. Figure 95 shows an example.

```
                          Work with System Activity                  ITSOSYS1
                                                            09/28/99  09:49:49

 Automatic refresh in seconds  . . . . . . . . . . . . . . . . .     5
 Elapsed time . . . . :   00:00:05       Average CPU util . . :   1.7
 Number of CPUs . . . :   2              Maximum CPU util . . :   1.8
                                         Minimum CPU util . . :   1.6

 Type options, press Enter.
   1=Monitor job   5=Work with job

     Job or                                        ---------Storage-----------
Opt  Task        User        Number   Thread        Allocated      Deallocated
      QPADEV0003 SUSAN2      100526   00000227          26              0
      QPADEV0011  TVB        100214   00000038          03              0
      ITRTR00012                                         0              0
      SMPOL001                                           0              0
      VIMTS002                                           0              0




                                                                          Bottom
 F3=Exit   F10=Update list   F11=View 2   F12=Cancel   F19=Automatic refresh
 F24=More keys
```

Figure 95. Work with System Activity display

Note

Many storage management tasks execute at the highest priority. Higher priority tasks consume more CPU.

8.3.4.1 What to do next

As your system noticeably slows down, you may not have time to use Disk Tools to determine storage usage. Try one or more of the following suggested actions:

- Check active jobs for restore or communication jobs that may be writing a large amount of data onto the disk.
- Check for queries that may be consuming lots of temporary storage. To check for temporary storage consumption, check the *Temp Storage* field under Run Attributes when working with such a job. See Figure 90 on page 126 for an example.

If you cannot isolate an offending job, then you may have a Runaway DASD problem. In this case, you should:

- Consider ending the system to restricted state for further problem analysis.
- Refer to 8.4, “Analyzing DASD consumption problems” on page 142, for other tools to help you find further information about the problem.

8.3.4.2 Resource contention and performance of system

When a resource is not available for your job, or when multiple users request the same resource concurrently, OS/400 manages the resource to preserve the integrity of the data. Contentions can occur when a system limit is reached.

To get around some limits that are job related, one solution is to have the job re-submit itself when it reaches the maximum. Another solution is to start the job with a different user profile and switch to the desired profile. The problem with this second approach is that the objects (for example, spooled file listing) are not as easily accessible as when they are stored under the current job. In the case of a print job, the application must direct the output to a specific output queue and monitor the queue itself, or use a data queue (if the spooled files are in a ready state) to monitor the output queue.

Note: Use security auditing of the spooled files to identify the spooled file generated.

On systems with a very active spool environment, a locking situation can be caused by a contention bottleneck on the output queue, the spool control block of the job, or the print spooled file internal indexes. A locking bottleneck can occur due to multiple jobs running either a WRKSPLF *ALL or WRKSPLF command on a user with many thousands of spooled files, and the default wait time on the class description expires. A message is issued indicating the necessary resource is not available (CPF3330) and that the lock operation for the object was not satisfied in the specified time interval (MCH5802).

To manage this event, you can perform the following steps:

1. Increase the default wait time in the class description.

2. Make sure latest PTFs for OS/400 programs managing spooled files are applied, specifically SF58933 for V4R4 and SF58419 for V4R3.

3. Refrain from doing a WRKSPLF *ALL as much as possible.

4. Spread the spooled files across multiple output queues and access the spooled files through the WRKOUTQ interface rather than WRKSPLF.

Note: There is a maximum of 9,999 spooled files per job on any V4R4 system.

Refer to *The System Administrator's Companion to AS/400 Availability and Recovery*, SG24-2161, for an appendix listing the system limits for V4R2 systems. V4R4 system limits are available at `http://www.redbooks.ibm.com` as an download for SG24-2161 under the Additional Material option.

8.3.5 Your system has stopped

When the system reaches 100% threshold of disk capacity, it ceases to function and posts what is termed a System Reference Code (SRC) on the front panel of the processor. The code displayed depends on the operation the system attempts first once all storage is used up.

For example, if the system tries to extend a directory, `SRC B600 0219` is posted on a non-RISC system or `SRC B600 0650`, and others, on a RISC system.

Note: Some of these RISC SRC codes can occur for non-storage problems as well. For example, if the system tries to allocate an Invocation Work Area (IWA), an `SRC B600 0615` is posted. In rare circumstances, an `SRC B900 3FFF` is posted, indicating a function check in the system arbitor job (QSYSARB). This SRC occurs when the system tries to perform database operations, error logging, and so on, with no storage available to complete the operation.

Once the system stops with any of these SRCs, you are forced to perform an IPL to recover.

8.3.5.1 Recovering your system

Follow these steps to recover your system:

1. Put the system in manual mode and power it off.
2. Return the system to normal mode and power it back on.
3. IPL the machine to a restricted state by taking a “Y” for the Start System to a restricted state option on the “IPL Options Display”. Refer to Figure 105 on page 140 for an example.

   This starts only the console and prevents other jobs from starting that may consume what little free space available.
4. If your system fails to IPL, refer to 8.3.5.3, “What to do if your system fails to IPL” on page 138.
5. If the IPL completes and the system is returned to a command line, issue a WRKSYSSTS command to determine the current storage level (that is, % system ASP used). Refer to Figure 91 on page 127 for an example.

If the storage level has dropped below the lower limit threshold

After the IPL, if the storage level drops below the lower limit threshold, there was a temporary storage problem. Enough storage was recovered during the IPL to keep below threshold. The immediate problem of storage consumption has been resolved.

You now have to “play detective” to determine the cause of the disk consumption, as not a lot of clues are available after an IPL. Suggested actions include:

1. Analyze history information for evidence of unusual storage growth.
2. Check history files for critical storage condition errors, as described in 8.3.5.2, “Analyzing historical information” on page 134.
3. If you cannot find an obvious cause, allow the users back on to the system. Closely monitor active jobs for high disk I/O or high DASD consumption. Use the tools described in 8.4, “Analyzing DASD consumption problems” on page 142.

If the storage level is still higher than the Storage Lower Limit

If there is a permanent storage problem, not much storage is recovered during the IPL. You need to determine what is consuming storage. Refer to the Disk Tools described in 8.4.1, “Using disk information tools” on page 142.

RTVDSKINF gives you information about all of the files on the system sorted in descending size order. It also provides you with the date last changed of the file and the user profile that created it.

Using this information and your application knowledge, you should be able to identify any unusually large files and investigate further the programs that created these files.

Note: If you choose to use the RTVDSKINF tool, make sure you have enough room for its temporary and spool files (usually less than 50,000 bytes).

8.3.5.2 Analyzing historical information

To analyze the history files, follow these steps:

1. Check QHST for all incidences of message CPF0907: `Serious storage condition may exist. Press Help.`

   This is done by entering the command `DSPLOG` on the command line and pressing F4. Change the beginning date to *BEGIN and press F10 for additional parameters. See Figure 96 for an example.

```
                              Display Log (DSPLOG)

 Type choices, press Enter.

 Log . . . . . . . . . . . . . .   QHST          QHST
 Time period for log output:
   Start time and date:
   Beginning time . . . . . . . .   *AVAIL        Time, *AVAIL
   Beginning date . . . . . . . .   *begin        Date, *CURRENT, *BEGIN
   End time and date:
   Ending time  . . . . . . . . .   *AVAIL        Time, *AVAIL
   Ending date  . . . . . . . . .   *CURRENT      Date, *CURRENT, *END
 Output . . . . . . . . . . . . .   *             *, *PRINT, *PRTWRAP










                                                                       Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display        F24=More keys
```

Figure 96. Display Log Command (Part 1)

   Next, page down to display the additional parameters, and enter CPF0907 in the message identifier field, as shown in Figure 97.

```
                          Display Log (DSPLOG)

Type choices, press Enter.


                          Additional Parameters

Jobs to display  . . . . . . . .   *NONE         Name, *NONE
  User . . . . . . . . . . . . .                 Name
  Number . . . . . . . . . . . .                 000000-999999
               + for more values


Message identifier . . . . . . .   cpf0907       Name, *ALL
               + for more values







                                                                   Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 97. Display Log command (Part 2)

Press the Enter key. A list of all CPF0907 messages in the history log is displayed, as shown in Figure 98.

```
                         Display History Log Contents

Serious storage condition may exist. Press HELP.
Serious storage condition may exist. Press HELP.
Serious storage condition may exist. Press HELP.
Serious storage condition may exist. Press HELP.
Serious storage condition may exist. Press HELP.















                                                                   Bottom
Press Enter to continue.

F3=Exit   F10=Display all   F12=Cancel
```

Figure 98. Display History Log Contents

Searching on the CPF0907 message allows the determination of how long the problem has taken place. To find out when the first message occurred, display additional message information by putting the cursor under the first message (as shown in Figure 98) and press F1 for help (Figure 99 on page 136).

```
                      Additional Message Information

Message ID . . . . . . :   CPF0907       Severity . . . . . . . :   80
Message type . . . . . :   Inquiry
Date sent  . . . . . . :   09/28/99      Time sent  . . . . . . :   10:35:32

Message . . . . :   Serious storage condition may exist. Press HELP.
Cause . . . . . :   The amount of available auxiliary storage in the system
  auxiliary storage pool has reached the threshold value. This is a
  potentially serious system condition. The auxiliary storage capacity is
  77,301,000 bytes.  The auxiliary storage used is 70,807,716 bytes.  The
  thresholdis90.00percent.Theauxiliarystorageavailableis9.40percent.
Recovery  . . . :   Use the WRKSYSSTS command to monitor the amount of storage
  used. The use of storage can be reduced by the following actions: Save
  objects by specifying STG(*FREE); delete any unused objects.  Save the old
  log versions of QHST that are not currently used and then delete them;
  either write or delete spooled files on the system. Failure to reduce the
  storage usage may lead to a situation requiring initialization of auxiliary
  storage and loss of user data.  The System Service Tools function can be
  used to display and change the threshold value. See the Backup and Recovery
  book, SC41-5304, for more information.

Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details   F12=Cancel
F21=Select assistance level
```

Figure 99. Additional message information

Once you know *when* the problem started, list the jobs that may have been responsible. Get a list of the active jobs before the first message and compare it to the list of those still active at the time the system stopped due to the 100% auxiliary storage threshold being reached.

Repeat the above procedure, but adjust the ending date and time to the date and time the first CPF0907 message occurred.

Tip

Search on messages CPF1124 and CPF1164. This gives you the starting and ending times of all jobs prior to the first CPF0907 message.

Match pairs of messages with the same job number. The unmatched start messages give you the active jobs at the time of the first CPF0907 message.

Refer to Figure 100 for an example.

```
                        Display History Log Contents

Job 031744/ALISON/QTTFT00897 ended on 09/28/99 at 06:05:33; 1 seconds used; end
Job 031743/ANNE/QTTFT01884 ended on 09/28/99 at 06:35:34; 1 seconds used; end c
Job 031752/GEN/QTTFT02017 started on 09/28/99 at 06:35:33 in subsystem QSYS
Job 031706/SUE/QTTFT04471 ended on 09/28/99 at 07:05:34; 2 seconds used; end co
Job 031753/CHAPPO/QTTFT00916 started on 09/28/99 at 07:35:34 in subsystem QSYS
Job 031754/JAMES/QTTFT00927 started on 09/28/99 at 07:35:34 in subsystem QSYS
Job 031750/ROB/QTTFT00888 ended on 09/28/99 at 08:05:34; 1 seconds used; end co
Job 031752/GEN/QTTFT02017 ended on 09/28/99 at 08:35:34; 1 seconds used; end co
Job 031757/JOANNE/QPADEV0008 started on 09/28/99 at 09:29:05 in subsystem QINTER
Job 031746/SELINA/QTFTP01245 ended on 09/28/99 at 09:33:39; 4 seconds used; end
Job 031753/CHAPPO/QTTFT00916 ended on 09/28/99 at 09:35:34; 1 seconds used; end
Job 031754/JAMES/QTTFT00927 ended on 09/28/99 at 10:05:34; 1 seconds used; end
Job 031759/MANDY/QPADEV0004 started on 09/28/99 at 10:08:56 in subsystem QINTER
                                                                         Bottom




Press Enter to continue.

F3=Exit   F10=Display all   F12=Cancel
```

Figure 100. Display History Log Contents

2. Check the job logs of these jobs for any clues. Look for a large number of files, large number of members within files, growth within the file or member, or journals being created. Consider running each job again individually and closely monitor for disk I/Os and temporary storage consumption.
3. If storage starts to climb, use the WRKACTJOB command and type `5` to work with the suspected job, followed by `3` for Run Attributes. The amount of temporary auxiliary storage (in K) that is currently allocated to this job is displayed. Refer to Figure 101 on page 138 for an example.

```
                         Display Job Run Attributes
                                                          System:   ITSOSYS1

 Job:   QPADEV0004     User:   ITSCID20       Number:   031759

 Run priority  . . . . . . . . . . . . . . . . . . . :   20
 Time slice in milliseconds  . . . . . . . . . . . . :   2000
 Eligible for purge  . . . . . . . . . . . . . . . . :   *YES
 Default wait time in seconds  . . . . . . . . . . . :   30
 Maximum CPU time in milliseconds  . . . . . . . . . :   *NOMAX
   CPU time used . . . . . . . . . . . . . . . . . . :     47579
 Maximum temporary storage in megabytes  . . . . . . :   *NOMAX
   Temporary storage used  . . . . . . . . . . . . . :     292
 Maximum threads . . . . . . . . . . . . . . . . . . :   *NOMAX
   Threads . . . . . . . . . . . . . . . . . . . . . :     1







 Press Enter to continue.

 F3=Exit   F5=Refresh   F9=Change job   F12=Cancel   F16=Job menu
```

Figure 101. Display Job Run Attributes

The F5 key refreshes the display and allows you to check if the temporary storage continues to increase.

4. If you are unable to determine the problem using these methods, use one of the more detailed analysis methods described in 8.4, “Analyzing DASD consumption problems” on page 142.

8.3.5.3 What to do if your system fails to IPL

After you have performed the steps described in 8.3.5.1, “Recovering your system” on page 133, if the system fails to IPL to a command line, but stops with an SRC code again instead, then there is a permanent storage capacity problem. Space has to be freed up during the IPL.

To do this, start another IPL by putting the system back in Manual mode and powering the system off. Power on with the system still in Manual mode.

Note: Manual mode is not a typical nor recommended daily action. Refer to *AS/400 System Operation,* SC41-4203, for instructions on how to place the system into manual mode and power off.

Complete the following steps to IPL the system to a restricted state:

1. At the first screen, IPL or Install the System as shown in Figure 102. Enter option `1` (Perform an IPL).

```
                         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
            5. Save Licensed Internal Code
```

Figure 102. IPL screen

2. When the Signon screen is displayed (as shown in Figure 103), sign on with a user profile with *SECOFR authority, for example QSECOFR.

```
                              Sign On
                                           System  . . . . . :   ITSOSYS1
                                           Subsystem . . . . :   QINTER
                                           Display . . . . . :   QPADEV0004

            User  . . . . . . . . . . . . . .
            Password  . . . . . . . . . . . .
            Program/procedure . . . . . . . .
            Menu  . . . . . . . . . . . . . .
            Current library . . . . . . . . .

                                       (C) COPYRIGHT IBM CORP. 1980, 1999.
```

Figure 103. Signon screen

3. On the Select Products to Work with PTFs screen (Figure 104 on page 140), press F3 (Exit) to continue.

```
                          Select Product to Work with PTFs

        Position to . . . . . . .____________ Product


        Type options, press Enter.  Press F21 to select all.
          1=Select

                     Product
        Opt  Product  Option  Release
         _   5769999  *BASE   V4R3M0
         _   5769SS1  *BASE   V4R3M0







 F3=Exit
```

Figure 104. Select Products display

4. The next screen is called IPL Options. It allows the clearing of permanent objects such as job queues, incomplete job logs and output queues. Specify a Y option for any of these that you wish to clear. Figure 105 shows an example.

```
                                     IPL Options


      Type choices, press Enter.

      System date  . . . . . . . . . . . . . .   07 / 26 / 88
      System time  . . . . . . . . . . . . . .   12 : 00 : 00
      Clear job queues . . . . . . . . . . . .   N
      Clear output queues  . . . . . . . . . .   N
      Clear incomplete job logs  . . . . . . .   N
      Start print writers  . . . . . . . . . .   Y
      Start system to restricted state . .   N


      Set major system options . . . . . . . .   Y
      Define or change system at IPL . . . . .   Y



 Press Enter to continue.

 F3=Exit
```

Figure 105. IPL Options display

Important

A Y for output queues clears all spooled files. This can potentially free up a lot of disk space as spool file entries can consume a lot of storage. Be sure this is what you want to do before selecting this option!

5. Bring the system up in Restricted State by selecting a Y for the Start system to restricted state option. This starts only the console and stops other jobs starting that may use up what little free space is available. Once the system is back up and you have ensured there is enough free space, you can restart the rest of the system.
6. After you select the desired options, press Enter.
7. You may get one more screen called Edit Rebuild of Access Paths (Figure 106). Press Enter.

```
                    Edit Rebuild of Access Paths                    ITSOSYS1
                                                           05/12/90 13:49:34

IPL threshold . . . . . . . . . . . . . 50   0-99

  Type sequence, press Enter.
  Sequence:  1-99, *OPN, *HLD

                    ------------Access Paths----------  Unique  Rebuild
  Seq   Status      File        Library     Member      Keyed   Time
  25__  IPL         QAPZSYM2    QSYS        QAPZSYM2    NO      00:00:01










 Press Enter to continue.

F3=Exit
```

Figure 106. Rebuild Access Path display

8. When you get the main menu screen, put the system back into Normal mode. Follow the diagnosis routines listed in 8.3.5.2, “Analyzing historical information” on page 134.
9. If the system fails to IPL using the preceding method, another recovery option is available if a spare disk unit is available. Add the spare non-configured disk unit to the ASP requiring space. Alternately, remove units from one user ASP and add it to another ASP.

   Note: This assumes information from the moved unit is either saved or can be cleared prior to the move.

   Call your service provider for more in-depth recovery processes.

8.4 Analyzing DASD consumption problems

There are many tools available on the AS/400 system to help you manage disk utilization once you know you have a DASD consumption problem. This section describes tools available to find out more information about the specific problem you have.

The tools discussed are:

- Disk information tools
- Performance tools
- Application Programming Interfaces (APIs)
- Performance Explorer (PEX)
- Enhanced PEX Trace

8.4.1 Using disk information tools

The disk information tools (RTVDSKINF and PRTDSKINF) provide static sizing information on:

- Libraries (as shown in Figure 107)
- Database files (as shown in Figure 108)
- Folders (as shown in Figure 109)
- System Objects (as shown in Figure 112)

Sample reports are shown in the following figures.

```
                                           Disk Space Report                                              Page     4
5769SS1 V4R4M0  990521                                                                             ITSOSYS1 09/28/99
12:05:13


                                           Library Information

                             % of          Size in     Last        Last
 Library       Owner         Disk        1000 bytes   Change      Use         Description

 R3PRDDATA     PRDOWNER      11.24        12556632.1   09/24/99    08/25/99    COLLECTION - created by SQL
 R3DEVDATA     DEVOWNER      11.13        12426780.7   09/24/99    09/08/99    COLLECTION - created by SQL
 R3DEVOFR      DEVOWNER       3.57         3986702.3   07/26/99    08/06/99    Library for *savf
 QGPL          QSYS           1.87         2086912.0   09/27/99    09/27/99
 QSYS          QSYS           1.79         1996321.3   09/27/99    09/27/99    System Library
 QPEXDATA      QSYS           1.10         1232277.5   09/14/99    09/22/99    Performance Explorer Data Library
 QUSRADSM      QSYS           1.00         1118879.7   04/22/99    04/22/99    adsm400 licpgm5769sv3
 *NOLIBRARY                    .74          827768.8
 QSTURT24      ASHLEY          .59          659501.1   09/15/99    09/15/99
 QSTURT8       ALLISON         .46          516313.1   09/13/99    09/22/99
 QSTURT6       ADRIANNE        .39          433287.2   08/04/99    08/25/99
 R3DEVOPT      R3OWNER         .39          431468.5   08/12/99    09/08/99
 QSPL          QSPL            .38          423206.9   09/27/99                Spool Library
 WXYZS         WXYZS           .37          417390.6   04/26/99    04/26/99    Temp Library
 QUSRSYS       QSYS            .31          343560.2   09/27/99    09/27/99

TOTAL                                    43288994.3
                                        * * * * *   E N D   O F   L I S T I N G   * * * * *
```

Figure 107. Disk Space Report: Library Information

```
                                Disk Space Report                                        Page     4
5769SS1 V4R4M0  990521                                                              ITSOSYS1 09/28/99
13:29:41
                                Specific Object Information

                                                          % of        Size in   Last      Last
 Object        Library     Type       Owner               Disk       1000 bytes Change    Use        Description

 R3DEVDATA     R3DEVOFR    *FILE      DEVOWNER            3.57       3986702.3  07/26/99  08/06/99   Library R3DEVDATA
 BACKUP01      QUSRADSM    *FILE      WXYZS                .94       1049645.1  04/21/99  04/22/99
 QAYPEMBRKT    QPEXDATA    *FILE      QSYS                 .59        661262.3  09/14/99  09/14/99   PEX MI Bracketing Event
 D010S         R3DEVDATA   *FILE      DEVOWNER             .52        585314.3  07/29/99             Defaults changed by R3
 D010S         R3PRDDATA   *FILE      PRDOWNER             .52        582168.6  07/14/99  08/25/99   Defaults changed by R3
 QAYPETIDX     QPEXDATA    *FILE      QSYS                 .46        515506.2  09/14/99  09/14/99   PEX Trace Index Data
 QAPMDMPT      LOCK        *FILE      QPGMR                .40        448352.3  09/13/99  09/22/99   Performance trace dump
 DOKCLU        R3DEVDATA   *FILE      TERRY                .40        445853.7  07/29/99  09/08/99   Defaults changed by R3
 DOKCLU        R3PRDDATA   *FILE      SUSAN                .40        445853.7  07/14/99  08/25/99   Defaults changed by R3
 DD03L         R3DEVDATA   *FILE      SCOTT                .32        353570.8  07/19/99  07/22/99   Defaults changed by R3
 DD03L         R3PRDDATA   *FILE      STACY                .32        353570.8  07/14/99  08/03/99   Defaults changed by R3
 ACMEDS1       JJLIBLE     *FILE      STEVE                .27        302268.4  08/23/99  08/23/99
 BACK          JJLIBXE     *FILE      MIKET                .27        302268.4  08/23/99  08/23/99
 FSIOP         WXYZS       *FILE      TODD                 .27        301482.0  04/22/99  04/22/99
 S5769AS1PA    QGPL        *FILE      QSYI                 .26        295714.8  09/21/99  09/21/99

 TOTAL                                                              19836485.7
                                 * * * * *   E N D   O F   L I S T I N G   * * * * *
```

Figure 108. Disk Space Report: Specific object Information

```
                                Disk Space Report                              Page     4
5769SS1 V4R4M0  990521                                                ITSOSYS1  09/28/99  13:52:25

                                    Folder Information

                              % of        Size in    Last
 Folder         Owner         Disk       1000 bytes  Change     Folder Path

 DLL            QSECOFR        .01         14351.4   09/23/99   QFPBFLR.001/QFPBSYS2/OS2
 BBBDF          QUMB           .01         12361.7   09/23/99   QUMBREPO
 QSVNOS2        QSYS           .01          8883.2   09/23/99
 QFPBSYS1       QSECOFR        .00          5550.6   09/23/99   QFPBFLR.001
 QFPNTWE        QSYS           .00          4818.4   09/23/99
 UNITABLE       QSYS           .00          4075.5   09/23/99   QBKBOOKS
 OS2            QSECOFR        .00          3411.5   09/23/99   QFPBFLR.001/QFPBSYS2
 QFPBSYS2       QSECOFR        .00          2594.8   09/23/99   QFPBFLR.001
 BBBFF          ITSCID19       .00          2190.3   09/26/99   QUMBREPO
 PSFONTS        QSECOFR        .00          1835.0   09/23/99   QFPBFLR.001/QFPBSYS1
 QFPNTWI        QSECOFR        .00          1606.7   09/23/99   QFPBFLR.001/QFPBSYS4
 DLL            QSECOFR        .00          1116.2   09/23/99   QFPBFLR.001/QFPBSYS2/IBMCOM

 TOTAL                                     62795.3
                                 * * * * *   E N D   O F   L I S T I N G   * * * * *
```

Figure 109. Disk Space Report: Folder Information

```
                              Disk Space Report                                            Page     1
5769SS1 V4R4M0  990521                                                  ITSOSYS1 09/28/99 14:17:18

                              System Information

 Information collected  . . . . . . . . . :   09/27/99  07:39:05    Machine type-model . . . . . . . . . . . :
9406-720
                                                                System serial number . . . . . . . . . . :   10-99999
 Total disk space on system in 1,000,000
   bytes  . . . . . . . . . . . . . . . . :   111669                Customize options specified for this report:
 Main storage size in megabytes . . . . . :   4096                    Report type  . . . . . . . . . . . . . :   *SYS

                                              % of        Size in
 Description                                  Disk     1,000,000 bytes
 User libraries                              34.62          38664.88
 User directories                             7.78           8689.22
 Folders and documents                         .07             73.27
 QSYS                                         1.80           2008.18
 Other IBM libraries                          4.63           5165.49
 Licensed Internal Code                       2.36           2639.89
 Temporary space                              1.58           1765.14
 Unused space                                46.09          51472.91
 Internal objects                              .15            168.44
 Objects not in a library                      .74            827.77
 TOTAL                                       99.82         111475.19
```

Figure 110. Disk Space Report: System Information (Part 1 of 3)

```
                              Disk Space Report                                            Page     2
5769SS1 V4R4M0  990521                                                  ITSOSYS1 09/28/99 14:17:18

                              System Information

                                                 % of        Size in
 Description                                           Disk     1,000,000 bytes
 OS/400 items reported:
   History files                                        .02             25.56
   Journal receivers                                    .01             11.72
   User profiles                                        .03             30.27
   Configuration information                            .00              2.61
   System help (QHLPSYS)                                .03             33.78
   Calendars                                            .00              1.32
   System directories                                   .00              2.35
   Document files                                       .00              3.08
     OfficeVision/400 items reported:
       Enrollment                                       .00               .11
       Mail files                                       .00               .34
       Text search                                      .00               .97
       Personal directories                             .00               .48
     Licensed Internal Code:
       LIC and tables                                   .88            977.24
       Dump space                                       .23            252.71
       Error logs                                       .00              3.24
       Trace tables                                    1.06           1189.15
       VLIC logs                                        .18            201.33
       VLIC control blocks                              .01             16.22

     * This list points out items of interest and does not contain all
     objects on the system.  For more information, see AS/400 System
     Operation V3R6, SC41-4203.
```

Figure 111. Disk Space Report: System Information (Part 2 of 3)

```
                                    Disk Space Report                                                    Page     3
5769SS1 V4R4M0  990521                                                                          ITSOSYS1 09/28/99
14:17:18

                                    System Information


                                                          % of        Size in
    Description                                           Disk     1,000,000 bytes
    Space used by internal objects:
      Additional configuration information                 .00              .13
      Communications                                       .01             6.77
      Data base                                            .00              .16
      Manage jobs                                          .10           111.89
      Monitor performance                                  .00              .00
      Folders                                              .00              .51
      Mail                                                 .00              .02
      OS/400                                               .01             9.60
      S/36 Environment                                     .00              .07
      Security                                             .00             2.15
      SNADS and object distribution                        .00             1.35
      Spool                                                .02            19.35
      Cross component                                      .01             7.77
      Product installation                                 .00              .09
      Recovery of information                              .00              .82
      Obsolete                                             .00              .00
      Internal system journals                             .01             7.76
    Miscellaneous items reported:
      Printer output & Spooling (QSPL)                     .38           423.43
      Replaced object library (QRPLOBJ)                    .00              .95
      Storage affected by RCLSTG                           .00              .00
      Reclaim storage library (QRCL)                       .04            40.18

    * This list points out items of interest and does not contain all
    objects on the system.  For more information, see AS/400 System
    Operation V3R6, SC41-4203.
                                          * * * * *   E N D   O F   L I S T I N G   * * * * *
```

Figure 112. Disk Space Report: System Information (Part 3 of 3)

This information is useful in identifying the biggest objects on the system.

Investigate each object further to see if its size can be reduced by:

- Deleting it, if it is obsolete.
- Archiving it to tape or compressed disk, if it is rarely used. Then delete it.
- Reorganizing it, if it has a large number of deleted records.

Refer to Chapter 7, “OS/400 disk storage management tools” on page 87, for a detailed description of how to run the RTVDSKINF and PRTDSKINF tools.

8.4.2 Using performance tools

The IBM Performance Tools for AS/400 Licensed Program Product (5769-PT1) is a set of commands and programs that provides system managers and operators tools for analysis, modelling and reporting of AS/400 systems performance levels.

Two specific areas in which the Performance Tools Licensed Program Product helps to identify DASD-related information are:

- The Work with System Activity (WRKSYSACT) command
- Performance reports

8.4.2.1 The Work with System Activity (WRKSYSACT) command

The WRKSYSACT command allows you to interactively work with the jobs and tasks currently running in the system.

The initial view (Figure 113) shows the total number of synchronous and asynchronous physical disk input/output (I/O) operations performed by the job or task during the elapsed time.

Note

These displays do not have a refresh key, so the information you see is not cumulative. They contain information on what has happened on the system during the elapsed time interval only. They do not contain the total I/O counts for the job.

```
                         Work with System Activity                ITSOSYS1
                                                          10/13/99  14:17:40
Automatic refresh in seconds  . . . . . . . . . . . . . . . . .    60
Elapsed time . . . . :   00:01:49       Overall CPU util . . :   40.6




Type options, press Enter.
  1=Monitor job   5=Work with job
                                                          Total  Total
     Job or                                          CPU  Sync   Async   PAG
Opt  Task        User        Number  Thread    Pty  Util   I/O    I/O  Fault
     QDFTJOBD    ITSCID20    009533  0000001E   50  31.9    180    413      0
     QDFTJOBD    ITSCID69    009534  00000022   10   4.9   3182     23      0
     CFINT01                                     0   2.5      0      0      0
     QPADEV000C  ITSCID20    009531  00000020   20    .7     84     21      0
     SMPOL001                                  100    .1      0    243      0
     IPRTR00013                                  0    .0      0      0      0
     QBATCH      QSYS        008780  00000001    0    .0     19      5      0
     RMTMSAFETA                                  0    .0      0      0      0
                                                                      More...
F3=Exit   F10=Update list   F11=View 2   F12=Cancel   F19=Automatic refresh
F24=More keys
```

Figure 113. The Work with System Activity display: View 1

The Total Sync I/O value is the sum of the synchronous database and non-database reads and writes. This is shown in view 2. See Figure 114 for an example.

```
                        Work with System Activity                  ITSOSYS1
                                                          10/13/99  14:17:40
Automatic refresh in seconds  . . . . . . . . . . . . . . . . .     60
Elapsed time . . . . :   00:01:49        Overall CPU util . . :   40.6



 Type options, press Enter.
   1=Monitor job   5=Work with job
                                                   --------Synchronous---------
      Job or                                         DB      DB   Non-DB  Non-DB
 Opt  Task         User        Number   Thread     Read   Write    Read    Write
      QDFTJOBD     ITSCID20    009533   0000001E      0     125       2       53
      QDFTJOBD     ITSCID20    009534   00000022      2       0    3174        6
      CFINT01                                         0       0       0        0
      QPADEV000C   ITSCID20    009531   00000020      0      11      18       55
      SMPOL001                                        0       0       0        0
      IPRTR00013                                      0       0       0        0
      QBATCH       QSYS        008780   00000001      0       0      17        2
      RMTMSAFETA                                      0       0       0        0
                                                                          More...
 F3=Exit   F10=Update list   F11=View 3   F12=Cancel   F19=Automatic refresh
 F24=More keys
```

Figure 114. The Work with System Activity display: View 2

The Total Async I/O value is the sum of the asynchronous database and non-database reads and writes. This is shown in view 3. See Figure 115 for an example.

```
                        Work with System Activity                  ITSOSYS1
                                                          10/13/99  14:17:40
Automatic refresh in seconds  . . . . . . . . . . . . . . . . .     60
Elapsed time . . . . :   00:01:49        Overall CPU util . . :   40.6



 Type options, press Enter.
   1=Monitor job   5=Work with job
                                                   --------Asynchronous--------
      Job or                                         DB      DB   Non-DB  Non-DB
 Opt  Task         User        Number   Thread     Read   Write    Read    Write
      QDFTJOBD     ITSCID20    009533   0000001E      0     409       0        4
      QDFTJOBD     ITSCID20    009534   00000022      0       0      21        2
      CFINT01                                         0       0       0        0
      QPADEV000C   ITSCID20    009531   00000020      0       2       2       17
      SMPOL001                                        0       7       0      236
      IPRTR00013                                      0       0       0        0
      QBATCH       QSYS        008780   00000001      0       0       0        5
      RMTMSAFETA                                      0       0       0        0
                                                                          More...
 F3=Exit   F10=Update list   F11=View 4   F12=Cancel   F19=Automatic refresh
 F24=More keys
```

Figure 115. The Work with System Activity display: View 3

The display can be sorted over these fields so that the jobs with the highest I/O counts during the elapsed time are easily identified.

8.4.2.2 Performance reports

There are a number of reports available in Performance Tools that show detailed information about disk utilization and disk capacity. An example is the Disk Activity section of the Component Report.

This report gives the average disk activity per hour and the disk capacity for each disk. Refer to Figure 116 for an example.

```
                                                     Component Report                                   11/11/98  8:20:28
                                                      Disk Activity                                             Page   16

                                                  STD I, B, then java/threads
 Member  . . . : Q983150757 Model/Serial  . : 510-2144/10-99999   Main storage . . :  512.0  M  Started . . . . :  11/11/98 07:58:01
   Library . . : PFRRES42   System name . . :         ABSYSTEM   Version/Release  :    4/ 2.0  Stopped . . . . :  11/11/98 08:18:43

                    ---------------- Average Disk Activity Per Hour ---------------- ----- Cache hit Statistics ----
            Srv    ------------------ Disk Arm Seek Distance ---------------------- Device   Controler    Write    -Disk Capacity-
 Unit   Util  Time         0       1/12       1/6        1/3        2/3      >2/3  Read       Read      Efficiency  MB    Percent
 ----   ---- -----  ---------  ---------  ---------  ---------  ---------  --------- -------   ---------   ---------- ------- -------
 0001 A  1.8 .0012      2,367      1,435        406         75        574         95     .0          .0           .0     446   22.6
 0001 B  1.5 .0009      2,375      1,897        507         95        243          2     .0          .0           .0     446   22.6
 0003     .4 .0002      3,156        870        435      1,131      1,618        217   17.0          .1         33.5     374   25.3
 0004     .9 .0003      3,457      1,395        690        983      1,757        403   30.7          .0         15.9     378   25.6
 0005     .8 .0004      3,272        887        510        936      1,746        362   18.6          .0         14.9     374   25.3
 0006     .8 .0003      2,897      1,099        365      1,084      1,435        229   40.2          .1         39.4     381   25.8
 0007    2.0 .0002      6,329      2,294      1,006      1,450      2,660        835   57.1          .1         59.4     814   25.8
 0008     .6 .0002      4,911      1,372      1,053      1,316      1,734        481   20.5          .1         49.8     803   25.5
 0009    5.0 .0002      7,484      4,650      1,357      2,149      2,233        501   73.9          .0         81.8     814   25.8
 0010     .5 .0002      4,832      2,039        760      1,433      1,644        304   29.0          .1          2.2     811   25.7

              Column                      Total             Average
           --------------------------  ----------------    ----------------
           Util                                                       1.4
           Srv Time                                                 .0003
           Disk Arm Seek Distance
              0                                 41,085
              1/12                              17,941
              1/6                                7,092
              1/3                               10,657
              2/3                               15,650
              >2/3                               3,434
           Cache hit Statistics
               Device Read                                           52.8
               Controller Read                                         .0
               Write Efficiency                                      59.5
           Disk Capacity
               MB                                5,641
               Percent                            25.1

 Unit                   --  Disk arm identifier
 Util                   --  Drive utilization
 Srv Time               --  Average service time per request in seconds
 Disk Arm Seek Distance --  Average seek distance distributions per hour
     0                  --  Number of zero seeks
     1/12               --  Number of seeks between 0 and 1/12 of the disk
     1/6                --  Number of seeks between 1/12 and 1/6 of the disk
     1/3                --  Number of seeks between 1/6 and 1/3 of the disk
     2/3                --  Number of seeks between 1/3 and 2/3 of the disk
     >2/3               --  Number of seeks greater than 2/ 3 of the disk
 Cache hit Stastistics  --
     Device Read        --  Percent of Device Read Hits  for each arm
     Controller Read    --  Percent of Controller Cache  Read Hits for each arm
     Write Efficiency   --  Percent of Efficiency of write Cache
 Disk Capacity          --  Average amount of disk space used or available
     MB                 --  Millions of bytes available on the disk
     Percent            --  Percent of space available on the disk
```

Figure 116. Disk Activity sample report

While this information is useful in analyzing performance-related disk problems such as disk access time or disk IOP throughput, it does not provide detailed information about real time DASD consumption.

8.4.3 Using APIs

For users who do not have Performance Tools, and therefore do not have access to the WRKSYSACT command, we provide information to create a command using APIs to show DASD I/O rates for jobs on the system. For the purposes of this redbook, we named it Work with DASD (WRKDASD).

The Work with DASD command displays auxiliary I/O requests, database I/O requests, and temporary storage used for each job selected.

This command allows you to identify the jobs on the system that have high I/O rates.

Note: For the purposes of this redbook, we assume that those jobs with high I/O rates are the most likely ones to cause DASD consumption problems.

Refer to Figure 117 for an example.

```
                               Work with DASD

 Type options, press Enter.
   5=Work with Open Files   6=Work with Spool Files   7=Work with Job Locks

                                                         Aux I/O  DB I/O  Temp
 Opt Job        User       Number Program     Subsystem  Request Request Storage
     QINTER     QSYS       008774             QINTER        1692               1
     QPADEV000B ITSCID20   009526 WRKSYSACT  QINTER        1593               3
     QPADEV000D ITSCID20   009541 WRKDASD    QINTER         989        2      3
     QPADEV000C ITSCID20   009542 NEP        QINTER         279    25991      2









                                                                        Bottom

 F3=Exit     F5=Refresh     F6=Sort by DB I/O   F7=Sort by Temp Stor
 F8=Sort by Aux I/O   F12=Cancel
```

Figure 117. Work with DASD display

The WRKDASD command can be obtained using one of the following methods:

- Order, load, and apply PTF Sxxxxxx. Follow the instructions in the cover letter to activate the command.
- Download it from the Web at: `http://www.as400service.ibm.com`
- Create it yourself by using the source code and programming instructions contained in Appendix D, “WRKDASD sample code” on page 201.

Note

This code is not officially supported by IBM and is provided on an “as is” basis.

Once you have the command installed on your system, make sure that libraries QGY and WRKDASD are in your library list. If not, use `ADDLIBLE QGY` and `ADDLIBLE WRKDASD` to add them to your library list.

To run the command, type `WRKDASD` and press F4 to prompt. Refer to Figure 118 for an example.

```
                        Work with DASD (WRKDASD)

 Type choices, press Enter.

 Output . . . . . . . . . . . . .   *              *, *PRINT
 Subsystem Name . . . . . . . . .   *ALL           Name, *ALL




                                                                Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 118. Work with DASD command prompt

You can display the information dynamically on the screen or produce a printout. Jobs can be selected by individual subsystem or *ALL.

Note

Select individual subsytems until you become familiar with the WRKDASD command. Selecting all jobs on the system can take a few minutes to calculate the cumulative database I/O figures. This is due to the fact that there is no API available at the time of publication of this redbook to identify which database files are currently in use by a particular job. It is expected that this limitation will be overcome in future releases of OS/400, and, therefore, the performance of this command will be significantly improved.

Make your selections, and then press Enter. A Work with DASD screen appears, as shown in Figure 119. We have used Output = * and Subsystem Name = QINTER for the following examples.

```
                              Work with DASD

Type options, press Enter.
  5=Work with Open Files   6=Work with Spool Files   7=Work with Job Locks

                                                         Aux I/O  DB I/O  Temp
Opt Job        User       Number Program    Subsystem  Request Request Storage
    QINTER     QSYS       008774            QINTER        1692               1
    QPADEV000B ITSCID20   009526 WRKSYSACT  QINTER        1593               3
    QPADEV000D ITSCID20   009541 WRKDASD    QINTER         989        2      3
    QPADEV000C ITSCID20   009542 NEP        QINTER         279    25991      2

                                                                        Bottom

F3=Exit     F5=Refresh     F6=Sort by DB I/O   F7=Sort by Temp Stor
F8=Sort by Aux I/O   F12=Cancel
```

Figure 119. Work with DASD display: Sorted by Auxiliary I/O

The default sort criteria for the WRKDASD command is Auxiliary I/O. Use F6 to change to Database I/O (as shown in Figure 120) or Temporary Storage or use F7 (as shown in Figure 121 on page 152).

```
                              Work with DASD

Type options, press Enter.
  5=Work with Open Files   6=Work with Spool Files   7=Work with Job Locks

                                                         Aux I/O  DB I/O  Temp
Opt Job        User       Number Program    Subsystem  Request Request Storage
    QPADEV000C ITSCID20   009542 NEP        QINTER         279    25991      2
    QPADEV000D ITSCID20   009541 WRKDASD    QINTER         989        2      3
    QINTER     QSYS       008774            QINTER        1692               1
    QPADEV000B ITSCID20   009526 WRKSYSACT  QINTER        1593               3

                                                                        Bottom

F3=Exit     F5=Refresh     F6=Sort by DB I/O   F7=Sort by Temp Stor
F8=Sort by Aux I/O   F12=Cancel
```

Figure 120. Work with DASD display: Sorted by Database I/O

```
                              Work with DASD

Type options, press Enter.
  5=Work with Open Files   6=Work with Spool Files   7=Work with Job Locks

                                                     Aux I/O  DB I/O  Temp
 Opt Job        User       Number Program    Subsystem Request Request Storage
     QPADEV000B ITSCID20   009526 WRKSYSACT  QINTER       1593               3
     QPADEV000D ITSCID20   009541 WRKDASD    QINTER        989       2       3
     QPADEV000C ITSCID20   009542 NEP        QINTER        279   25991       2
     QINTER     QSYS       008774            QINTER       1692               1









                                                                        Bottom

 F3=Exit     F5=Refresh     F6=Sort by DB I/O   F7=Sort by Temp Stor
 F8=Sort by Aux I/O   F12=Cancel
```

Figure 121. Work with DASD display: Sorted by Temporary Storage

The WRKDASD command contains a number of options that provide further information about each job to help with problem analysis.

Option 5 provides you with a list of open database files for the selected job and the database I/O count for each file. To access this information, enter option 5 from the main WRKDASD display, as shown in Figure 122. Press Enter.

These two screens help you determine which of the files used by the job are generating high I/O rates

```
                              Work with DASD

Type options, press Enter.
  5=Work with Open Files   6=Work with Spool Files   7=Work with Job Locks

                                                     Aux I/O  DB I/O  Temp
Opt Job        User       Number Program    Subsystem Request Request Storage
    QINTER     QSYS       008774            QINTER       1692               1
    QPADEV000B ITSCID20   009526 WRKSYSACT  QINTER       1593               3
    QPADEV000D ITSCID20   009541 WRKDASD    QINTER        989       2       3
 5  QPADEV000C ITSCID20   009542 NEP        QINTER        279   25991       2









                                                                       Bottom

F3=Exit     F5=Refresh     F6=Sort by DB I/O   F7=Sort by Temp Stor
F8=Sort by Aux I/O   F12=Cancel
```

Figure 122. Work with DASD display: Work with Open Files

Press F11 to display I/O details, as shown in Figure 123.

```
                             Display Open Files

Job . . :   QPADEV000C     User . . :   ITSCID20       Number . . . :   009542
Number of open data paths . . . . . . . . . . :      2


                      Member/    Record     File   I/O   ----Open--- Relative
File       Library    Device     Format     Type  Count  Opt Shr-Nbr  Record
QDUI80     QSYS       QPADEV000C USRRCD     DSP      13  IO  NO
DATAFILE   BBLIB      DATAFILE   DATAREC    PHY   25991  IO  NO         13000










Press Enter to continue.

F3=Exit   F5=Refresh   F11=Display scoping data   F12=Cancel   F16=Job menu
```

Figure 123. Display Open Files

Option 6 provides a list of spooled files for the selected job, and the total pages generated for each spooled file. To access this information, enter option `6` from the main WRKDASD display as shown in Figure 124. Press Enter.

```
                              Work with DASD

 Type options, press Enter.
   5=Work with Open Files   6=Work with Spool Files   7=Work with Job Locks

                                                       Aux I/O  DB I/O  Temp
 Opt Job        User       Number Program    Subsystem Request Request Storage
     QINTER     QSYS       008774            QINTER       1692               1
     QPADEV000B ITSCID20   009526 WRKSYSACT  QINTER       1593               3
     QPADEV000D ITSCID20   009541 WRKDASD    QINTER        989       2       3
  6  QPADEV000C ITSCID20   009542 NEP        QINTER        279   25991       2










                                                                        Bottom

 F3=Exit     F5=Refresh     F6=Sort by DB I/O   F7=Sort by Temp Stor
 F8=Sort by Aux I/O   F12=Cancel
```

Figure 124. Work with DASD display: Work with Spool Files

As shown in the example in Figure 125 on page 154, this screen helps you determine if a job is generating large numbers of pages or large spooled files.

```
                         Work with Job Spooled Files

Job:   QPADEV000D     User:   ITSCID20       Number:   009541

Type options, press Enter.
  1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
  8=Attributes        9=Work with printing status

                 Device or                        Total   Current
Opt  File        Queue       User Data   Status   Pages     Page   Copies
     QSYSPRT                              FIN
     QSYSPRT     QPRINT      WRKDASD      OPN       659                1
     QPDSPJOB                             FIN
     QPDSPJOB                             FIN
     QPDSPJOB                             FIN
     QPDSPJOB                             FIN
     QPDSPJOB                             FIN
     QPDSPJOB                             FIN
     QPDSPJOB                             FIN
                                                                     More...
Parameters for options 1, 2, 3 or command
===>
F3=Exit   F10=View 3   F11=View 2   F12=Cancel   F22=Printers   F24=More keys
```

Figure 125. Work with Job Spooled Files display

Option 7 provides you with a list of job locks for the selected job, and the type and status of each lock. To access this information, enter option `7` from the main WRKDASD display, as shown in Figure 126. Press Enter.

```
                              Work with DASD

Type options, press Enter.
  5=Work with Open Files   6=Work with Spool Files   7=Work with Job Locks

                                                      Aux I/O  DB I/O  Temp
Opt Job        User       Number Program    Subsystem Request Request Storage
    QINTER     QSYS       008774            QINTER       1692               1
    QPADEV000B ITSCID20   009526 WRKSYSACT  QINTER       1593               3
    QPADEV000D ITSCID20   009541 WRKDASD    QINTER        989       2       3
 7  QPADEV000C ITSCID20   009542 NEP        QINTER        279   25991       2










                                                                       Bottom

F3=Exit     F5=Refresh     F6=Sort by DB I/O   F7=Sort by Temp Stor
F8=Sort by Aux I/O   F12=Cancel
```

Figure 126. Work with DASD display: Work with Job Locks

Refer to Figure 127 for an example.

```
                        Work with Job Locks
                                                      System:   ITSOSYS1
Job:   QPADEV000C     User:   ITSCID20      Number:   009542

Job status:   ACTIVE

Type options, press Enter.
  5=Work with job member locks   8=Work with object locks

                             Object                     Member
Opt  Object      Library     Type        Lock     Status  Locks
     BBLIB       QSYS        *LIB        *SHRRD    HELD
     DATAFILE    BBLIB       *FILE-PHY   *SHRRD    HELD    YES
     ITSCID20    QSYS        *USRPRF     *SHRRD    HELD
     MAIN        QSYS        *MENU       *SHRNUP   HELD
                                         *SHRNUP   HELD
     QDUI80      QSYS        *FILE-DSP   *SHRNUP   HELD
     QGPL        QSYS        *LIB        *SHRRD    HELD
     QGSPWRKF    QSYS        *PNLGRP     *SHRNUP   HELD
     QHLPSYS     QSYS        *LIB        *SHRRD    HELD
                                                               More...
F3=Exit   F5=Refresh   F10=Display job record locks   F11=Display thread data
F12=Cancel
```

Figure 127. Work with Job Locks display

Printing

Entering `*PRINT` for the output option on the WRKDASD prompt screen (Figure 128) produces a detailed report, as shown in Figure 129 and Figure 130 on page 157.

```
                        Work with DASD (WRKDASD)

Type choices, press Enter.

Output . . . . . . . . . . . . .   *            *, *PRINT
Subsystem Name . . . . . . . . .   *PRINT       Name, *ALL

















                                                                Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 128. Work with DASD command prompt

Note that the print option can also be run in batch or be scheduled to run on a regular basis. The jobs are listed in descending order of Auxiliary I/O Requests.

Work with DASD						Date 10/14/99 Time 10:27:03 Page 1	
Job Name	User Name	Job Number	Program	Subsystem	Aux I/O Requests	DB I/O Requests	Temporary Storage
QSPLMAINT	QSYS	008736			56,912	0	0
SCPF	QSYS	000000			45,816	0	18
QIJSSCD	QIJS	008797	QIJSCMON	QSYSWRK	6,024	0	1
QPWFSERVSO	QUSER	009468		QSERVER	4,544	0	2
QDBSRVXR	QSYS	008741			3,104	381	0
QZDASOINIT	QUSER	008830		QSERVER	2,962	0	53
QDBSRVXR2	QSYS	008745			2,933	879	0
QSYSSCD	QPGMR	008772	QEZSCNEP	QCTL	2,492	0	0
QFILESYS1	QSYS	008740			2,116	0	0
QSYSCOMM1	QSYS	008746			1,961	0	0
QZDAINIT	QUSER	008782		QSERVER	1,864	0	20
QSYSARB	QSYS	008721			1,803	0	1
QJOBSCD	QSYS	008737			1,768	0	1
QINTER	QSYS	008774		QINTER	1,726	0	1
QTVDEVICE	QTCP	008814	QTVDEVMG	QSYSWRK	1,408	0	1
QSYSWRK	QSYS	008752		QSYSWRK	1,369	0	2
QSERVER	QSYS	008767		QSERVER	1,196	0	2
QCTL	QSYS	008751		QCTL	1,153	0	1
QALERT	QSYS	008738			1,142	0	1
QTMSNMP	QTCP	008811	QTOSMAIN	QSYSWRK	1,135	0	2
QTCPIP	QTCP	008799		QSYSWRK	1,113	0	2
QSNMPSA	QTCP	008820	QNMSARTR	QSYSWRK	1,090	1	2
QTVDEVICE	QTCP	008813	QTVDEVMG	QSYSWRK	877	0	1
QNPSERVR	QUSER	008794		QCMN	845	0	3
QBATCH	QSYS	008780		QBATCH	822	0	1
QUSRWRK	QSYS	008768		QUSRWRK	807	0	1
QPFRADJ	QSYS	008735			767	0	0
QPWFSERVSD	QUSER	008801		QSERVER	754	0	1
QDBSRV01	QSYS	008728			633	0	0
QCMN	QSYS	008788		QCMN	494	0	1
QZSOSMAPD	QUSER	008834		QSYSWRK	476	0	2
QSERVER	QPGMR	008783		QSERVER	441	0	2
QLUS	QSYS	008726			426	0	1
QZBSEVTM	QUSER	008761	QZBSEVTM	QSYSWRK	389	0	1
QTVDEVICE	QTCP	008816	QTVDEVMG	QSYSWRK	375	0	1
QZSOSGND	QUSER	008829		QSYSWRK	373	0	2
QNPSERVS	QUSER	008823		QUSRWRK	371	0	3
QCMNARB02	QSYS	008748			355	0	0
QZLSSERVER	QPGMR	008808		QSERVER	349	0	3
QPADEV000B	ITSCID20	009561	WRKDASD	QINTER	306	95	2
QPASVRP	QSYS	008765	QPASVRP	QSYSWRK	301	0	0
QSYSARB3	QSYS	008723			298	0	9
QSYSARB5	QSYS	008725			296	0	0
QTPOP01198	QTCP	008835		QSYSWRK	268	0	1
QZDASSINIT	QUSER	008832		QSERVER	254	0	20
QZSCSRVR	QUSER	008796		QCMN	243	0	2
QCMNARB03	QSYS	008749			230	0	0
QDBSRV02	QSYS	008729			226	0	1
QPWFSERVS2	QUSER	008771		QSERVER	219	0	2
QCMNARB01	QSYS	008747			214	0	1
QZRCSRVSD	QUSER	008827		QSYSWRK	201	0	1
QNEOSOEM	QUSER	008777	QNEOSOEM	QSYSWRK	184	0	1
DAILYQUERY	WITTSM	007401	DAILYQUERY	QBATCH	182	0	1

Figure 129. Work with DASD report (Part 1 of 2)

QDBSRV04	QSYS	008731			177	0	0
QZSCSRVS	QUSER	008818		QUSRWRK	170	0	2
QTMSNMPRCV	QTCP	008817	QTOSRCVR	QSYSWRK	170	0	2
QSYSARB4	QSYS	008724			170	0	0
QTFTP01394	QTCP	009557		QSYSWRK	163	0	3
QPWFSERVSS	QUSER	008773		QSERVER	162	0	2
QPWFSERV	QUSER	008778		QSERVER	157	0	2
QDBSRV03	QSYS	008730			155	0	1
QDBSRV05	QSYS	008732			150	0	0
QMSF	QMSF	008798		QSYSWRK	147	0	0
QTVTELNET	QTCP	008812		QSYSWRK	146	0	2
QSPL	QSYS	008766		QSPL	141	0	1
QZSCSRVSD	QUSER	008819		QSYSWRK	129	0	1
QNEOSOEM	QUSER	008760	QNEOSOEM	QSYSWRK	122	0	2
QZRCSRVR	QUSER	008795		QCMN	120	0	1
DAILYQUERY	WITTSM	007397	DAILYQUERY	QBATCH	118	0	1
QNEOSOEM	QUSER	008779	QNEOSOEM	QSYSWRK	111	0	1
QSYSARB2	QSYS	008722			109	0	0
QPWFSERV	QUSER	008775		QSERVER	103	0	2
QNMAPINGD	QUSER	008790		QCMN	99	0	1
QZDASRVSD	QUSER	008833		QSERVER	97	0	1
QPASVRS	QSYS	008784	QPASVRS	QSYSWRK	97	0	0
QNMAREXECD	QUSER	008791		QCMN	88	0	2
QZHQSRVD	QUSER	008822		QSYSWRK	86	0	1
QPWFSERV	QUSER	008776		QSERVER	69	0	2
QTCPMONITR	QTCP	008802	QTOCMON	QSYSWRK	64	0	1
QNPSERVD	QUSER	008824		QSYSWRK	62	0	1
QTFTP01505	QTCP	009558		QSYSWRK	62	0	3
QTFTP00920	QTCP	009562		QSYSWRK	52	0	3
QPASVRS	QSYS	008789	QPASVRS	QSYSWRK	51	0	0
QZLSFILE	QUSER	008781		QSERVER	50	0	2
QTPOP00809	QTCP	008837		QSYSWRK	42	0	1
QZHQSSRV	QUSER	008821		QUSRWRK	42	0	1
QTLPD02966	QTCP	008803		QSYSWRK	42	0	1
QTVDEVICE	QTCP	008815	QTVDEVMG	QSYSWRK	38	0	1
QTLPD03029	QTCP	008804		QSYSWRK	38	0	2
Q400FILSVR	QSYS	008742			29	0	1
QACSOTP	QUSER	008792		QCMN	25	0	0
QPASVRS	QSYS	008786	QPASVRS	QSYSWRK	25	0	0
QTPOP00753	QTCP	008836		QSYSWRK	22	0	1
QZSOSIGN	QUSER	009503		QUSRWRK	22	0	2
QPASVRS	QSYS	008787	QPASVRS	QSYSWRK	21	0	0
QTFTP00932	QTCP	009559		QSYSWRK	18	0	3
QZRCSRVS	QUSER	009498		QUSRWRK	18	0	1
QQQTEMP1	QSYS	008743			16	0	0
QLZPSERV	QUSER	008793		QCMN	15	0	0
QDCPOBJ2	QSYS	008734			15	0	0
QIWVPPJT	QUSER	008825		QSYSWRK	14	0	1
QLUR	QSYS	008739			12	0	0
QTFPJTCP	QUSER	008831		QSERVER	8	0	1
QQQTEMP2	QSYS	008744			5	0	0
QDCPOBJ1	QSYS	008733			2	0	0
QDCPOBJ1	QSYS	008733			2	0	0

*** End of Listing ***

Figure 130. Work with DASD report (Part 2 of 2)

8.4.4 Using the Performance Explorer (PEX)

Performance Explorer (PEX) gathers statistics and lets you define the reports you want based on these statistics. In a sense, PEX is much like the Performance

Tool monitor function. They both collect data. However, PEX provides a greater level of detail. Note that PEX does not do any analysis for you.

The collection functions and related commands of Performance Explorer are part of the OS/400 operating system. The reporting function and its associated commands are part of the Performance Tools for AS/400 Licensed Product, the Manager feature.

To use PEX, access the IBM Performance Tools Main Menu, as shown in Figure 131, by typing `GO PERFORM` on the command line.

```
PERFORM                  IBM Performance Tools for AS/400
                                                              System:   ITSOSYS1
Select one of the following:

     1. Select type of status
     2. Collect performance data
     3. Print performance report
     4. Capacity planning/modeling
     5. Performance utilities
     6. Configure and manage tools
     7. Display performance data
     8. System activity
     9. Performance graphics
    10. Advisor

    70. Related commands


Selection or command
===> 5

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F16=System main menu
(C) COPYRIGHT IBM CORP.
```

Figure 131. Performance Tools Main menu

From this menu, enter option `5` (Performance utilities) as shown in Figure 132.

```
                              Performance Utilities

Select one of the following:

     1. Work with job traces
     2. Work with Performance Explorer
     3. Select file and access group utilities




Selection or command
===> 2

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 132. Performance Utilities menu

From this menu, enter option `2` (Work with Performance Explorer) as shown in Figure 133.

```
                          Work with Performance Explorer

 Select one of the following:

      1. Add Performance Explorer Definition (ADDPEXDFN)
      2. Change Performance Explorer Definition (CHGPEXDFN)
      3. Remove Performance Explorer Definition (RMVPEXDFN)
      4. Start the Performance Explorer (STRPEX)
      5. End the Performance Explorer (ENDPEX)
      6. Print Performance Explorer Reports (PRTPEXRPT)
      7. Delete Performance Explorer Data (DLTPEXDTA)









 Selection or command
 ===>

 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 133. Work with Performance Explorer display

Again, while this information can be useful in analyzing performance-related disk problems, such as disk access time or disk IOP throughput, it does not provide detailed information about real time DASD consumption.

8.4.5 Using Enhanced PEX Trace

Enhanced PEX Trace is an advanced collection and reporting capability provided through a set of additional performance commands that can be added to the base PEX facility. Using one of the reporting commands, NETSIZE, you can obtain detailed DASD consumption information from within the Enhanced PEX Trace facility.

Note

The Enhanced PEX Trace utility is intended for use only by AS/400 customers, IBM Service Professionals, and Business Partners experienced in performance investigation on the AS/400 system. Analysis of the results requires a thorough understanding of the AS/400 factors, including user application design.

Refer to the following section for information as to how to obtain the enhanced PEX Trace commands.

8.4.5.1 Getting Enhanced PEX Trace commands

The Enhanced PEX Trace collection and reporting commands are provided through a set of “glossary” and real PTFs.

There are PEX Trace PTFs for OS/400 V4R4 and V4R3. The delivery vehicle for the PEX library (named QYPINT) is APAR MA19096. The PTFs to install include:

- MF19888
- MF19985
- MF20357
- MF20363
- MF20614
- MF20638
- SF50564
- SF50576

Tip

- These are *not* full, "real" PTFs. They do not fix code. They provide new function.
- Do not attempt a LODPTF or APYPTF.
- Follow the instructions in the PTF cover letters exactly.

These PTFs include two libraries:

- Library QYPINT

  This library contains the PEX Trace Main Menu (PEXTRC01) and the supporting commands to simplify the collection and reporting of detailed PEX Trace data.

- Library SMTRACE

  This library contains the reports (queries) over data collected using the PEX Trace commands from the PEX Trace Main Menu in library QYPINT.

Note

The *SAVFs (save files) for these libraries can also be downloaded from the Web at: `http://www.as400service.ibm.com/p_dir/pexgui.nsf`

Other considerations

- Ensure that you have installed the latest PTFs for Performance Tools/400, Performance Explorer and "real" PTFs for Enhanced PEX Trace prior to running any PEX functions.
- Ensure that you have sufficient disk space available. The Enhanced PEX Trace collection stage can generate large disk files, depending on the selection criteria.

Note

If you are unsure how to address these considerations, call your service provider for more in-depth assistance.

8.4.5.2 Collecting Enhanced PEX Trace information

Once you have installed the Enhanced PEX Trace commands, follow these steps to collect Enhanced PEX Trace information:

1. Make sure that QYPINT is in your library list. If not, use `ADDLIBLE QYPINT` to add it to your library list.
2. Type `GO PEXTRC01` to get to the PEX Trace Main Menu (as shown in Figure 134).

3. Prior to collecting any Trace data, verify that there are no other PEX sessions (collections) currently active. To do this, enter option `1` (Display Active PEX Session Name) from the PEX Main Menu (as shown in Figure 134). The Display Active PEX Sessions display appears (Figure 135).

```
PEXTRC01                     PEX Trace Main Menu
                                                            System:   ITSOSYS1
Select one of the following:

     1. Display Active PEX Session Name
     2. Start PEX Trace Data Collection
     3. End PEX Trace Data Collection
     4. PEX Trace Reports Menu
     5. Query Active PEX Session Status

     6. Display Collected PEX Session Names
     7. Display PEX Trace Run Information for a Session
     8. Display PEX Trace Run Time for a Session

     9. Work with Submitted Jobs

    10. Delete PEX Trace Session Data

Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
F13=Information Assistant  F16=AS/400 main menu
```

Figure 134. PEX Trace Main Menu

```
                      Display Active PEX Sessions


        Session Name       Total Active Sessions = 1
        ITSOSample

                                                              Bottom
F3=Exit
```

Figure 135. Display Active PEX Sessions

If an active PEX session exists, it must be ended. Enter option `3` (End PEX Trace Data Collection) before a new PEX Trace session is started.

When there are no active PEX sessions, PEX Trace data can be collected.

Enter option 2 (Start PEX Trace Data Collection). A prompt screen (Figure 136) appears.

```
                        V4R4 VERSION OF SMTBCH (SMTBCH)

 Type choices, press Enter.

 Definition/MBR name  . . . . . . DFN
 Type of problem  . . . . . . . . PROBLEM
 Length of trace in minutes . . . DURATION
 Maxdata to collect . . . . . . . MAXDATA        500000
 Library for collected data . . . DATALIB        QYPINT
 Min CPU sample (milliseconds)  . CPUSAMPLE      200
 Trace MI CALL Events?  . . . . . MIBRACKET      *NONE
 I/O Counts/Details . . . . . . . PDIOTIMDTL     *NONE
 Break MSG when time is up? . . . BREAKMSG       N
 Submit job to batch? . . . . . . SBMJOB         Y
 Job queue name . . . . . . . . . JOBQ           QCTL
 Job queue library  . . . . . . . JOBQLIB        *LIBL
 Trace specific jobs/tasks? . . . ADDJOBS        N
 Convert PEX Trace Data?  . . . . CVTPEX         N
 PRTPEXRPT & CPYSPLF to LIB?  . . PRTPEX         N
 Job Priority for PRTPEXRPT . . . PRTPEXPTY      51
                                                                      Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 136. Storage Management Trace in Batch (SMTBCH) display

The values you specify for the parameters on this command define the type and amount of information that the PEX Trace collects.

4. Specify values using the recommendations in Table 5 for collecting DASD consumption information.

Table 5. PEX Trace recommended parameter values

Parameter	Description	Recommended value
DFN	Definition/Member Name	User choice. We recommend that you include a reference to the date and type of information collected.
PROBLEM	Type of information to collect	*DISKSPACE for DASD consumption information.
DURATION	The elapsed time (in minutes) before the collection ends	30 to 60 minutes for *DISKSPACE.
DATALIB	Library for collected data	If you specify a library name other than QYPINT, the library is created if it does not already exist.
MIBRACKET	Trace MI call events	Specify *ENTEXTMI if you want to see application program names in your PEX reports.
CVTPEX	Convert PEX trace data	Y - This automatically generates the QAYPExxxx database files into your library

All values that are not explicitly mentioned in Table 5 should be left as the default value, unless otherwise recommended by your service provider.

5. Press Enter to submit the PEX Trace data collection to batch.
6. When the collection stage is complete, you are ready to start printing DASD consumption reports.

8.4.5.3 Printing Enhanced PEX Trace reports

Although not specifically designed for DASD consumption problems, Enhanced PEX Trace reports can be tuned, using the NETSIZE reporting command, to find DASD consumption information. This section explains how to do this, gives examples of useful reports, and helps you understand how to interpret these reports.

You can also write your own queries over the database files produced during the PEX collection phase. Refer to 8.4.5.5, "Writing your own queries" on page 167, for a list of the files generated, and Appendix B.12 of the *AS/400 Work Management Guide,* SC41-5306, for detailed file layouts.

Using the NETSIZE reporting command

Once you have collected PEX information, create and print Enhanced PEX Trace reports by following these steps:

1. From the PEX Trace Main Menu, as shown in Figure 134 on page 161, enter option `4` (PEX Trace Reports Menu). Refer to Figure 137 for an example.

```
PEXTRC02                    PEX Trace Reports Menu
                                                           System:   ITSOSYS1
Select one of the following:

     1. Full File Opens
     2. Physical Disk IOs
     3. Activation Group Creates
     4. Faults
     5. Disk Space Consumption
     6. Address Consumption
     7. Size of Objects Used During PEX Trace Collection
     8. CPU Utilization
     9. Program Activations and Deactivations






Selection or command
===> 5

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
F13=Information Assistant  F16=AS/400 main menu
```

Figure 137. PEX Trace Reports Menu

2. Enter option `5` (Disk Space Consumption) and a prompt screen appears (Figure 138).

```
                      Object/Segment size changes (NETSIZE)

Type choices, press Enter.

PF132 and PEX files mbr name . .   *FIRST        Name, *FIRST
PF132 and PEX file library . . .                 Name
Submit as batch job? . . . . . .   *NO           *YES, *NO
Force rescan of PF132 mbr? . . .   *NO           *YES, *NO
TDE num (16 *HEX* digits)  . . .   *ALL
Include all threads of TDE?  . .   *NO           *NO, *YES
Starting Date-Time . . . . . . .   *ALL
Ending Date-Time . . . . . . . .   'YYYY-MM-DD-HH.MM.SS.SSSSSS'
Number of intervals  . . . . . .   30            10-1000
Both TEMPs and PERMs?  . . . . .   Y             Y, N
TEMP segment/objects?  . . . . .   N             Y, N
PERM segments/objects? . . . . .   N             Y, N




                                                                       Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 138. NETSIZE command prompt

3. Specify the member name and library used during the PEX data collection stage. Leave all other values as the default.

Note

Refer to *AS/400 Performance Explorer Tips and Techniques,* SG24-4781, Section 6.6.5 for detailed information about the NETSIZE command.

4. A job is then submitted to the batch queue which produces four reports that provide information about DASD consumption during the collection period. These reports are:
   - Net size change (in bytes) of individual segments
   - Net size change, grouped by (name, object type, segment type)
   - Net size change, grouped by (object, segment description)
   - Net size change by job or task

   Examples of each report are shown in the following figures.

```
                              Net size change (in bytes) of individual segments.
                              (total size change at bottom of report)
                              QUERY NAME . . . . . NETSIDRPT
                              LIBRARY NAME . . . . SMTRACE
                              FILE         LIBRARY      MEMBER       FORMAT
                              NETSIDOBJ2   JIMPEXDTA    R000000001   NETSIDOBJ2
                              DATE . . . . . . . . 09/21/99
                              TIME . . . . . . . . 14:22:25

                 Net size     Object/segment  SID   Object/segment        Object  Segment  Description
                 change       address         type  name                  type    type
                 (in bytes)

                     20,480    D202D4DDCA00     T   QPADEV000LCOOK         1AEF    007C    TEMPORARY - PROCESS
   CTL SPACE
                     16,384    DA5D3791ED00     T   QWCJOBINDEX            0EEF    0001    TEMPORARY - INDEX
                     16,384    E3B3891B0C00     T   COOK                   0EC4    0001    INTERACTIVE PROFILE
                     16,384    F2CF52BFA200     T   QWCSBSINDEX            0EEF    0001    TEMPORARY - INDEX
                      8,192    D006F418B200     T   QPADEV000LCOOK         1AEF    007D    TEMPORARY - PROCESS
   CTL SPACE
                      8,192    D57023646C00     T   WMHQ Data Queue Cach   19EF    0001    TEMPORARY - SPACE
                      8,192    E17EEA6D3D00     T   TNCDFNDIR              19EF    0001    TEMPORARY - SPACE
                      8,192    C3214DE9A500     T   HEAP DATA SEG TYPE     0000    20D6    HEAP DATA SEG
                      4,096    CB416DED5700     T   COOK                   0EC4    0002    INTERACTIVE PROFILE
                      4,096-   CFF68325B200     T   *DESTROYED             0000    0000
                      8,192-   C03A54884C00     T   *DESTROYED             0000    0000
                     16,384-   E8689E536700     T   *DESTROYED             0000    0000
                     16,384-   80989AE20100     T   *DESTROYED             0000    0000

   FINAL TOTALS
   TOTAL             61,440
* * *  E N D  O F  R E P O R T  * * *
```

Figure 139. Net size change (in bytes) of individual segments

```
 Net size change, grouped by (name, obj-type, seg-type).
                              (total net change at bottom of report)
                                 QUERY NAME . . . . . NETOBJRPT2
                                 LIBRARY NAME . . . . SMTRACE
                                 FILE         LIBRARY      MEMBER       FORMAT
                                 NETOBJRPT1   JIMPEXDTA    R000000001   NETOBJRPT1
                                 DATE . . . . . . . . 09/21/99
                                 TIME . . . . . . . . 14:22:25

                    Net size   Object/Segment                Description
                    change     name,
                    (bytes)    obj type, seg type

                     20,480    QPADEV000LCOOK       1AEF 007C  TEMPORARY - PROCESS CTL SPACE
                     16,384    QWCSBSINDEX          0EEF 0001  TEMPORARY - INDEX
                     16,384    QWCJOBINDEX          0EEF 0001  TEMPORARY - INDEX
                     16,384    CXXK                 0EC4 0001  INTERACTIVE PROFILE
                      8,192    WMHQ Data Queue Cach 19EF 0001  TEMPORARY - SPACE
                      8,192    TNCDFNDIR            19EF 0001  TEMPORARY - SPACE
                      8,192    QPADEV0000069K       1AEF 007D  TEMPORARY - PROCESS CTL SPACE
                      8,192    HEAP DATA SEG TYPE   0000 20D6  HEAP DATA SEG
                      4,096    BAZZAGOWER QPADEV003 0EC4 0002  INTERACTIVE PROFILE
                     45,056-   *DESTROYED           0000 0000

FINAL TOTALS
TOTAL                61,440
* * *  E N D  O F  R E P O R T  * * *
```

Figure 140. Net size change, grouped by (name, object type, segment type)

```
                                    Net size change, grouped by Object/Segment description.
                                     Destroyed and unknown types group under a blank description.
                                     Total net size change at bottom of report.

                                          QUERY NAME . . . . . NETDSCRPT2
                                          LIBRARY NAME . . . . SMTRACE
                                          FILE         LIBRARY      MEMBER       FORMAT
                                          NETDSCRPT1   JIMPEXDTA    R000000001   NETDSCRPT1
                                          DATE . . . . . . . . 09/21/99
                                          TIME . . . . . . . . 14:22:26

09/21/99  14:22:26                                            PAGE    1

                 Net size change   Description
                 (bytes)
                          32,768   TEMPORARY - INDEX
                          28,672   TEMPORARY - PROCESS CTL SPACE
                          20,480   INTERACTIVE PROFILE
                          16,384   TEMPORARY - SPACE
                           8,192   HEAP DATA SEG
                          45,056-

FINAL TOTALS
TOTAL                     61,440
* * *  E N D  O F  R E P O R T  * * *
```

Figure 141. Net size change, grouped by (object, segment description)

```
                                          Net size change by Job or Task.
                                          QUERY NAME . . . . . NETTDERPT2
                                          LIBRARY NAME . . . . SMTRACE
                                          FILE         LIBRARY      MEMBER       FORMAT
                                          NETTDERPT1   JIMPEXDTA    R000000001   NETTDERPT1
                                          TASKINFO     JIMPEXDTA    JIM0914A     QYPETASKI
                                          DATE . . . . . . . . 09/21/99
                                          TIME . . . . . . . . 14:22:26

             Net size change   Job/Task          Job/Task          Job         Job
             (bytes)           task count        name              user        number

                      86,016   000000000001456E  QPADEV000L        COOK        100752
                       8,192-  00000000000006FD  IPRTR00008
                      16,384-  0000000000000045  RMSRVCTKLO

FINAL TOTALS
TOTAL                 61,440
* * *  E N D  O F  R E P O R T  * * *
```

Figure 142. Net size change by job or task

Note

The *AS/400 Performance Explorer Tips and Techniques,* SG24-4781, redbook provides additional examples of the Performance Explorer functions and examples of the Enhanced Performance Explorer trace support. A good source of detailed information on how to interpret enhanced PEX Trace reports is contained in Chapter 8.

8.4.5.4 Interpreting Enhanced PEX Trace reports

The reports identify which objects and which jobs used the most DASD during the collection period. However, as you can see from the sample reports listed in

8.4.5.3, “Printing Enhanced PEX Trace reports” on page 163, PEX reports contain a lot of detailed technical data. The flexibility of the tool allows you to produce many customized reports. To build a report specific to performance, capacity, or utilization, contact an AS/400 performance consultant for recommendations.

Assess whether this DASD usage is normal or excessive. Investigate the application programs you suspect of contributing to the DASD consumption problem.

8.4.5.5 Writing your own queries

You can write your own queries over the database files produced by the collection phase of Enhanced PEX Trace. To do this, you need to understand:

- Which files are generated by each type of problem selected in the collection stage. See Figure 136 on page 162.
- What information is stored in each file.

Files generated

Table 6 shows a list of all files generated by the PEX Trace collection stage.

Table 6. Files generated by PEX Trace

File name	Description
QAYPEANAL	Analysis Tracking File
QAYPEASM	Auxiliary Storage Management Event Data
QAYPEBASE	Base Event Data
QAYPECICFG	Basic Configuration Data
QAYPECOCFG	Common Configuration Data
QAYPEDASD	DASD Event Data
QAYPEDSRV	DASD Server Event Data
QAYPEEVENT	Event Mapping Data
QAYPEFQCFG	Hardware Configuration Frequency Data
QAYPEHEAP	Heap Event Data
QAYPEHMON	Hardware Data
QAYPEHTOT	Hardware Instruction Totals Data
QAYPEHWCFG	Hardware Configuration Data
QAYPEHWMAP	Hardware Mapping Data
QAYPEJVA	Java Event Data
QAYPEJVCI	Java Class Info
QAYPEJVMI	Java Method Info
QAYPEJVNI	Java Name Info
QAYPELBRKT	Lic Bracketing Data
QAYPELCPLX	Complex MI Instruction List

File name	Description
QAYPELICI	Lic Name Mapping Data
QAYPELJOB	Job List
QAYPELLIC	Lic Procedure List
QAYPELMET	Metric List
QAYPELMI	MI Program List
QAYPELNAMT	Task Name List
QAYPELNUMT	Task Number List
QAYPEMBRKT	MI Bracketing Event Data
QAYPEMICPX	Complex MI Instruction Mapping Table
QAYPEMII	MI Name Mapping Data
QAYPEMIPTR	MI Pointer Data
QAYPEMIUSR	MI User Event Data
QAYPENLIC	Lic Name Data
QAYPENMI	MI Name Data
QAYPEPERD	Periodic Mode Event Data
QAYPEPGFLT	Page Fault Event Data
QAYPEPPANE	Profile Pane Data
QAYPEPSUM	Profile Summary Data
QAYPEPWDW	Profile Window Data
QAYPERLS	Database Level Indicator File
QAYPERMPM	Resource Mgmt Process Mgmt Event Data
QAYPERMSL	Resource Mgmt Seize Lock Event Data
QAYPERUNI	General Run Data
QAYPESAR	Segment Address Range Event Data
QAYPESEGI	Segment Mapping Data
QAYPESTATS	Statistics Data
QAYPESTCFG	Stats Configuration Data
QAYPES36	Advanced/36 Event Data
QAYPETASKI	Task Data
QAYPETIDX	Trace Index Data
QAYPETRCFG	Trace Configuration Data
QAYPETRCPT	Trace Point Data
QAYPEUNKWN	Unknown Event Data
QAYPEUSRDF	User-Defined Event Data

Data file layouts

A complete list of PEX Trace file layouts is contained in Appendix B.12 of the *AS/400 Work Management Guide*, SC41-5306.

8.5 Summary

As previously mentioned, the information described in this chapter is based on Version 4 Release 4 of OS/400. It is by no means exhaustive. The IBM tools described in this chapter may be enhanced, or new ones provided, in future releases of the operating system. Further information and assistance in identifying and addressing DASD consumption problems may be available from your local service provider.

Appendix A. Architecture

The *system processor's* role is to run the programs and perform the calculations related to the unique business application programs for your AS/400 system. The *service processor's* role is to manage accessing the hardware. The system processor says “I want” or “I want to do” and the service processor says “I'll go get it for you” or “I'll do that for you”, rather like a husband and wife relationship.

This appendix provides a more detailed accounting of the theory and functions associated with various AS/400 system components for those readers requiring more technical depth. While this detailed knowledge is not necessary to operate the AS/400 system, a background knowledge of the technical architecture is useful to visualize what goes on “under the covers”.

We discuss the architecture of these components in this appendix:

- Buses
- SPCN
- AS/400 IOP/IOAs
- DASD I/O cabling
- Device controllers

A.1 AS/400 buses and interfaces

Buses are the vehicle used by the system to transport information to and from the processors, main storage and the I/O devices. Bus categories include the Private bus, Main Storage bus, System I/O bus, SPCN bus, and the Serial Link bus (commonly called the Optical Bus). Their relationship is diagramed in Figure 143.

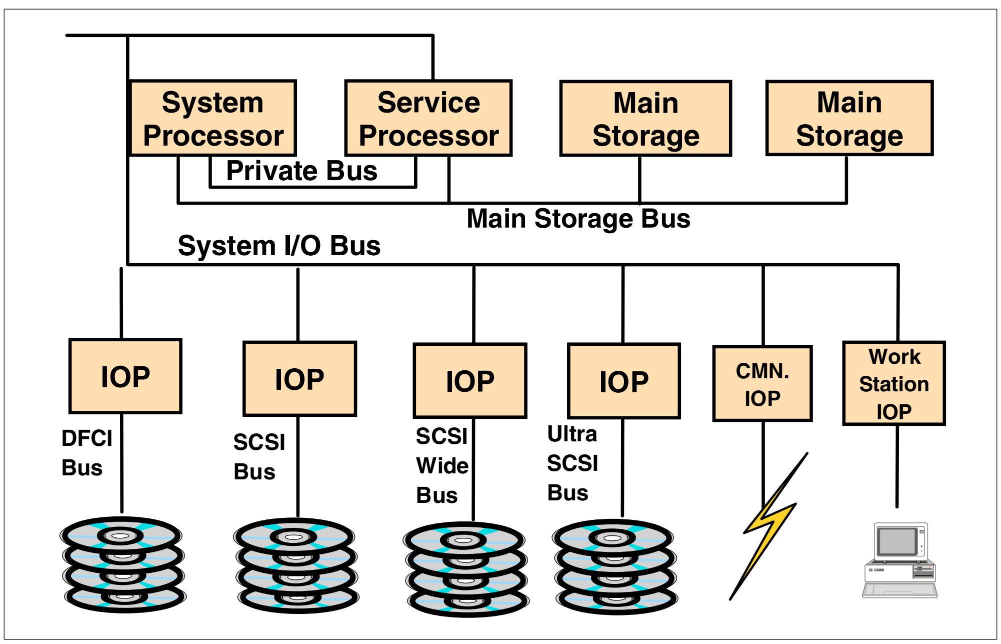


Figure 143. AS/400 buses

A *bus* is a physical facility on which data is transferred to all destinations. One or more conductors are used for transmitting signals or power on the bus.

Buses used on the AS/400 system include:

- Private bus
- Main storage bus
- I/O (system I/O) bus: Both SPD and PCI architecture
- IOP to device bus
  - SPD: Both DFCI and SCSI standards
  - PCI: SCSI standard
- System Power Control Network bus (SPCN)

Interfaces are the cabling that connect the IOP and IOAs to the device controllers. The controller passes information to the I/O devices as a permanent record, in the form of printed, displayed, or written information.

Interface types include the Device Function Controller Interface (DFCI) and the Small Computer System Interface (SCSI). Three variations are referenced in Figure 143 on page 171.

A.1.1 Private and main storage buses

This section introduces the function of the private and main storage buses. Refer to Chapter 1, “System architecture” on page 3, for more detailed information about both the system and service processors.

Private and main storage buses are internal buses. The private bus consists of control lines between the service processor and the system processor. The private bus carries programs, instructions, commands, and information between the system processor and the service processor. Control lines built into the private bus switch control between the processors.

The private bus is necessary because the system processor has no read only storage (ROS) and cannot give bus control function until loaded with Licensed Internal Code during an IPL. The service processor controls the bus to access its own code and the system processor code from the load source I/O bus unit. The service processor diagnostic support uses the private bus to obtain control of the I/O bus when errors occur in the system processor.

The main storage bus carries information such as programs, instructions, and data between the main storage cards, the system and service processors, and the I/O bus control units (BCU). In AS/400 systems, BCUs are located within the system processor and the service processor, as well as within the bus adapter cards.

A.1.2 I/O bus (system I/O buses)

An I/O bus (also known as a data bus) is used to communicate data internally and externally to and from a processing unit, storage, and peripheral devices. It provides all signal, power, and ground connections to the internal adapters (IOPs and IOAs).

There are two architectures of I/O buses:

- System Products Division (SPD)
- Peripheral Component Interconnect (PCI)

The SPD bus is named after the division which developed it. It is used across multiple platforms. SPD has been the state of the art technology for decades.

PCI is an industry standard format that allows the AS400 to choose from a wide range of devices to integrate into the system. PCI technology is gaining popularity.

All system buses on the AS/400 system available through 1999 are SPD technology. Several of the interfaces between IOP/IOA and individual device buses are available in PCI technology.

A.1.3 I/O bus (IOP/IOA to device buses)

The IOP or IOA connection to the device buses are referred to as I/O buses. I/O buses provide the same function between the IOP or IOA and their attached devices as the system bus does between the system processors and the IOP and IOAs.

A.2 SPD and PCI architecture

SPD and PCI technology provide the same function in an AS/400 system. However, they appear quite different from each other, as pictured in Figure 144.

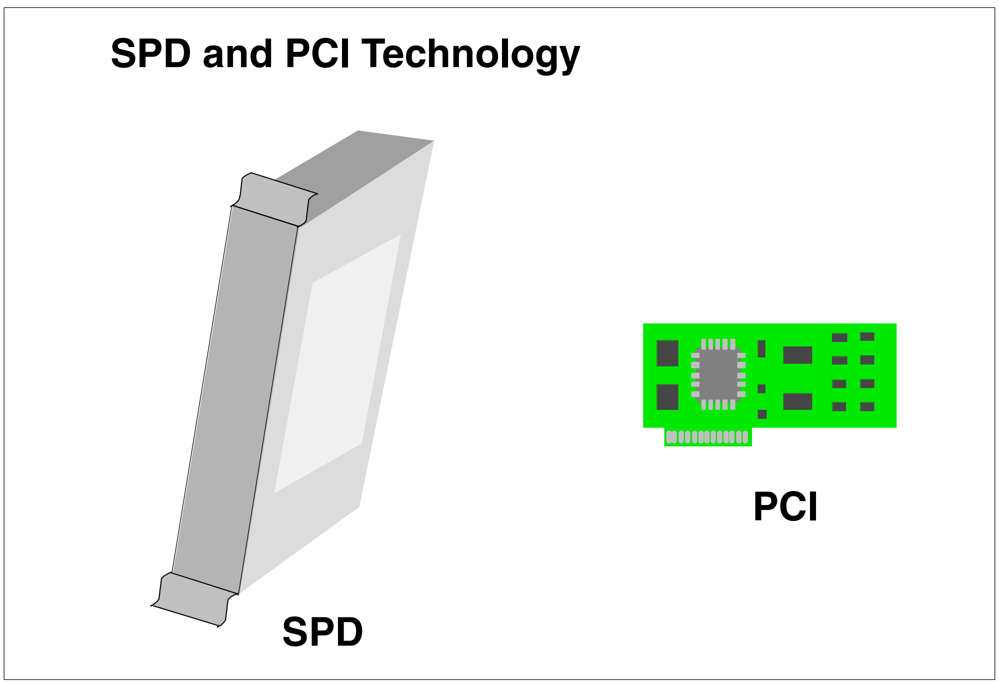


Figure 144. Typical SPD and PCI IOP/IOA packaging

Most functions supported with SPD cards have equivalent function cards in PCI format. For example, IOP or IOA to device buses come in both SPD and PCI technologies. The fundamental bus architecture of the AS/400 system remains unchanged when using PCI adapters. The AS/400 IOP continues to:

- Offload the main processor's workload
- Isolate the host from adapter and network errors
- Manage, configure, and service the adapters

PCI architecture offers advantages in flexibility for non-AS400 system structures.

Some products are designated Customer Setup (CSU). Customer Installable Features (CIF) was introduced on the Model 600, 170 and 150 for orders of additional features, such as disk features, PCI I/O cards, and external cables. PCI cards enable an implementation of CIF.

SPD and PCI buses are further distinguished with different technologies, as discussed in A.2.1, “Device Function Controller Interface (DFCI) bus” on page 174, and A.2.2, “Small Computer System Interface (SCSI) bus” on page 174.

Whether SPD or PCI is available depends on the system model. Early models, in particular, only have SPD technology. Many models, manufactured beginning with OS/400 V4R1, may have only SPD, or may have a mix of SPD and PCI, or may have all PCI components. The system model number referenced determines which technology is available.

The number of processors are too numerous to mention in this redbook. For a listing of current SPD and PCI processor choices, refer to the *AS/400 System Builder,* SG24-2155, or the *AS/400e System Handbook,* GA19-5486.

A.2.1 Device Function Controller Interface (DFCI) bus

The DFCI bus uses a protocol to communicate with the IOPs. In earlier generations of computing architecture, this was the preferred industry standard.

The DFCI bus is the IBM implementation of the ANSI IPI-3 industry standards. In IPI-3, the system processor stays linked to the IOP/IOA until whatever the process the IOP/ IOA is to perform completes. Any device on the IOP/IOA communicates with any other device on the same IOP/IOA through the system processor. The device controllers in this architecture depend on the IOP and system processor to function.

Device types that utilize the DFCI interface on the AS/400 system include:

- 9331 Diskette Unit
- 9332 Disk Unit
- 9335 Disk Controller Unit - Model A01
- 9335 Disk Storage Unit - Model B01
- 9336 Disk Unit
- 9347 Tape Unit

These device types are explained in A.2.2, “Small Computer System Interface (SCSI) bus” on page 174.

A.2.2 Small Computer System Interface (SCSI) bus

As device interface technology improves, there is less utilization of devices requiring DFCI. Small computer system interface (SCSI) interface provides the same function, but more flexibility.

The SCSI bus is designed to support an any-to-any communications protocol with device-to-device transfer permitted between devices within the same device unit. The use of SCSI enables advances such as RAID-5 and integrated hardware disk compression within a device subsystem.

An advantages of the SCSI bus over the DCFI bus is that the device controllers can work independently from the IOP/IOA and the system processor. Thus, the system processor issues a command to the device and disconnects from the device before the device completes the requested function. This frees the system processor and the IOP/IOA to go on with other tasks, while the device completes the requested function. This improved speed increases the amount of system throughput.

A.2.2.1 SCSI options on the AS/400 system

SCSI architecture is implemented in several forms on the AS/400: SCSI wide, SCSI1 and SCSI2.

Low-end AS/400 models use SCSI to connect to tape and disk devices. Mid- and high-end AS/400 models connect these SCSI device types:

- 2440 tape unit
- 2800 and 2801 integrated disk units
- 2802 integrated disk units (SCSI wide)
- 636x tape unit
- 7208 tape unit
- 9346 tape unit
- 9348 tape unit
- 9337 disk unit models (0XX and 1XX use SCSI)
- 9337 disk unit models (2XX, 4XX, and 5XX use SCSI wide)
- All internal disk unit models

SCSI wide bus

The SCSI wide bus is the same architecture as the SCSI bus, except the SCSI wide bus has a wider bandwidth on which to transfer data.

SCSI wide is used on the 9337 model 2XX, 4XX, 5XX, and storage units of some Advance Series and eSeries system models.

SCSI 1 and SCSI 2

The different types of SCSI (SCSI-1 and SCSI-2) refer to the driver and receiver used.

The number 2 in SCSI-2 implies that it is the second version of SCSI standards. SCSI-2 supports SCSI-1 since it consists of the SCSI 1 basics.

All interfaces on the internal and external devices on Stage 1 processors (that is, the AS/400 Model B) use SCSI-1.

Beginning with Stage 2 systems (as in the AS/400 Model D, E, and F models), interfaces on the integrated (internal) devices supported by the base Multi-Function IOP (MFIOP) are SCSI-2.

The SCSI-2 interface supports SCSI-1 devices, such as the #2800 and #2801 disk unit devices. The #2802 is a SCSI-2 device running on the same MFIOP. Internal DASD installed in a #5051, #5052 or #5058, utilize the SCSI-2 interface (except repackaged #2800 and #2801 DASD).

The #5051 or #5052 and all storage expansions manufactured subsequently run on an IOP to provide a SCSI-2 interface. SCSI-1 devices are supported in the first seven logical slots.

External storage developed for the AS/400 system, beginning with the model 2XX 9337s and continuing with the Enterprise Storage and Versatile Storage Servers, attach using SCSI-2 interfaces.

The following lists distinguish SCSI-1 from SCSI-2:

- SCSI-1 is single-ended. Characteristics include:
  - Low cost.
  - Do not require twisted pair wire.
  - Maximum cable length of 19.605 feet.
  - Susceptible to electrical noise.
  - Used primarily for connections within a cabinet.
  - Termination is done at the end of each cable or on the device.
  - Distance between device and bus must be less than 3.921 inches.
  - Distance between device and device must be at least 13.070 inches.
- SCSI-2 has a differential end. Characteristics include:
  - Higher cost.
  - Maximum cable length 81.688 feet. Twisted pair wire recommended.
  - Less susceptible to electrical noise than SCSI single-ended.
  - Primarily used for connections external to a cabinet.
  - Termination is done at the end of each cable or on the device.
  - Distance between device and bus must be less than 26.140 inches.
  - Distance between device and device must be at least 13.070 inches.

A.2.3 The system power control network (SPCN) bus

The system power control network (SPCN) bus carries data that controls and monitors power between AS/400 racks, towers, and units.

Implementing a SPCN bus enables the system to receive and post System Reference Codes (SRCs) and messages relating to power when traditional paths are unavailable because of the loss of power. The SPCN bus enables the system through relays to control the power to turn off or on individual components within the system.

In stage one hardware (before the creation of SPCN), the entire system is powered off to service any one area of the system. If power is lost to any area, the entire system loses power.

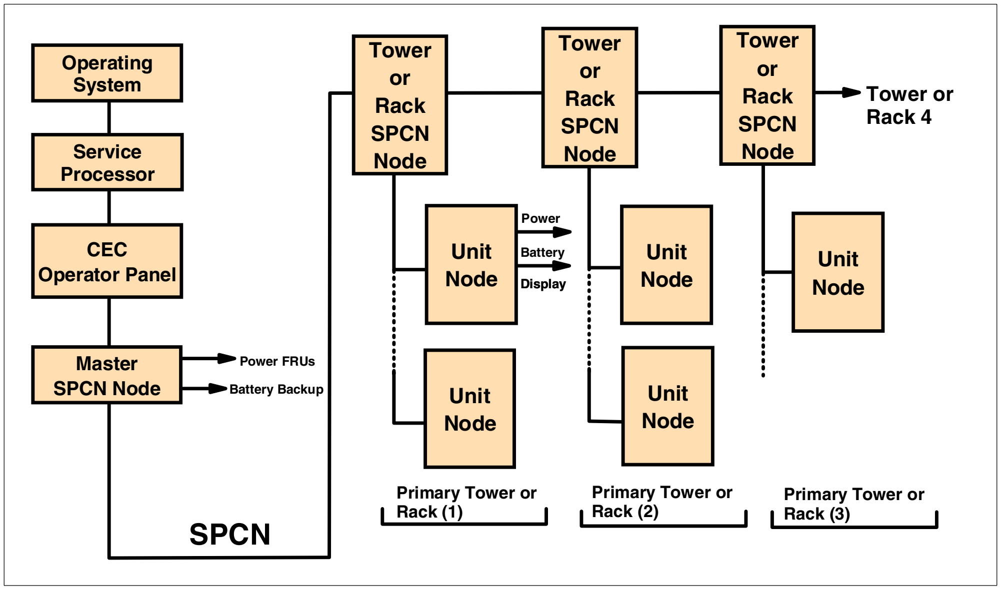


Figure 145. SPCN bus architecture

Most integrated units and internal devices in AS/400 systems use SPCN. SPCN uses microprocessors and a serial communication network to connect power in the rack and rack-mounted units. SPCN permits the operating system access to system power status and faults which are not available otherwise.

A.2.3.1 SPCN power components

The primary hardware components of SPCN, in addition to standard power components, are SPCN nodes and the SPCN interface. Refer to Figure 145 to observe that the power system in each tower/rack and each tower- or rack-installed device unit contains an SPCN node. An *SPCN node* is a microprocessor placed in the SPCN network, making the connecting points necessary for distributing power, sending commands, and reporting status. A SPCN node interfaces to the unit's power supply and local display panel. Each tower/rack node has its own power supply and remains powered on as long as there is building power available to the rack.

The location of a node in the network determines what interface it has to the power supply, power backup units, display panels, and rack emergency power-off (EPO) circuits.

A.2.3.2 Nodes

The three type of nodes are:

- Master node
- Tower or rack node
- Unit node

There is one master node for the network, one tower/rack node for each tower or rack, and one unit node for each tower/rack mounted device unit. Each node and its placement in the tower/rack is shown in Figure 145.

Master node

The master node is the interface between the operating system and the serial network. The master node is a microprocessor located in the primary system tower or rack. The master node is programmed to issue network commands and poll for status from other SPCN nodes in the network.

The master node communicates with the operating system (through the control panel), exchanging commands and network status. The master node monitors power and status of the power supply. During power interruption the master node starts backup power. The master node is powered on as long as building power is available to the tower or rack.

Command responses and error status from the SPCN network are collected by the master node and formatted for return to the operating system.

Tower or rack node

A tower or rack SPCN node is located in each tower or rack. It receives power-on and power-off commands from the master node for switching AC power (on or off) to the tower- or rack-mounted units. Each tower or rack node controls power switching to outlets within racks, and to power connections within towers. The outlets are switched on or off in groups of five.

Unit node

The unit node is a microprocessor located inside an AS/400-attached device (with SPCN circuits). Each unit node is programmed to report device status and respond to commands from the master node.

A.3 SPCN interfaces

The SPCN nodes in each unit are connected to rack ports with copper cables or through internal cabling, depending on the location of the unit, and to tower nodes through internal cabling. The nodes in each tower or rack are serially connected to other towers or racks with copper or optical cables.

In SPCN-enabled systems, bulk power modules, system regulators, battery backup units, power components of internal DASD and tape, as well as other system power components can be monitored and controlled. If errors occur, the system operator message queue (QSYSOPR or sometimes QSYSMSG), also is notified of the failing power condition. The message contains detailed information about the failure, often identifying a field replaceable unit (FRU) to be replaced.

The first external units to use the SPCN network were introduced with OS/400 V2R3.

Beginning with the Advanced Series AS/400 models, the SCSI buses can be quiesced, using SPCN, for a few seconds. This allows a single storage unit to be physically removed from the system and serviced in concurrent mode. This is accomplished through a call command, as in `CALL QCNCMNT`.

Beginning with the update of OS/400 V3R7 (February 1997), this function is an option from the Hardware Service Manager (HSM) menu screen. Beginning with OS/400 V4R1, this function is more efficiently incorporated under the control of SLIC.

A.4 AS/400 IOP and IOA

The AS/400 system has a fixed number of system processors that can be installed in a single frame. This number varies by model. The AS/400 system uses a multi-microprocessor configuration to provide the multiprocessor advantage, such as DASD, tape, communications, workstations, and so on.

A.4.1 Hierarchy of microprocessors

Figure 146 shows the hierarchy of processors including the main system processor, the service processor, and other processors, each dedicated to a particular I/O device type.

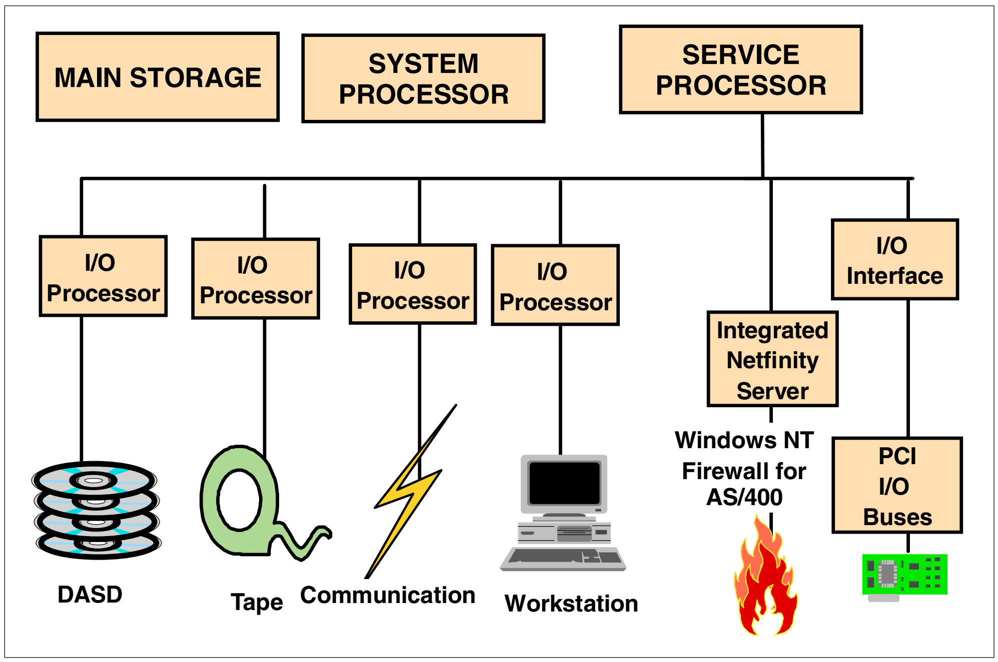


Figure 146. Hierarchy of microprocessors

When the main system processor encounters a request for data to be written to or read from any I/O device, the request is delegated to a particular microprocessor dedicated to that I/O device. The device handles the operation, while the system processor continues with another application program. On a fully configured Model 740, for example, there can be approximately 12 system processors, one service processor, and 237 I/O microprocessors on the system.

This hierarchy of microprocessor design provides the AS/400 system with outstanding performance in the commercial, transaction-based environment. Microprocessor technology is easily used and updated any time without disrupting the rest of the system.

A.4.2 Input output processor (IOP) and input output adapter (IOA)

A DASD subsystem attachment consists of one or more IOA functions and one IOP function. The DASD subsystem performs the following operations:

- Sends and receives data while performing the following checks:
  - I/O bus parity read/write
  - Read data parity and write data parity on the interface
  - Parity of all data passed through the IOP/IOA card
- Stores the status of error conditions
- Some error recovery routines

The DASD IOP converts I/O bus directives to DASD controller directives using Licensed Internal Code (LIC). On AS/400 systems, the DASD I/O processor is also called a *storage controller* or a *magnetic storage device controller*.

The magnetic storage device controller card is a combination I/O processor and I/O adapter card. It permits the attachment of one or more magnetic storage devices to the AS/400 system.

The IOP is varied on during a system initial microcode program load (IMPL). The IOP vary on basic assurance test (BAT) checks the IOP hardware and the IOP memory, and then loads the IOP memory with the subsystem Licensed Internal Code (LIC).

Every disk enclosure on the AS/400 system is attached to a supported feature IOP. Some storage IOPs support only one disk enclosure, and some support many.

The selection of the proper DASD IOP depends on the level of system availability protection needed, potential growth requirements, system performance, and whether any existing DASD units are migrated as a part of a system upgrade.

It is important to understand the relationships between the various storage IOPs and their supported disk enclosures. A reference for storage IOPs and the DASD they support is the *AS/400e System Handbook*, GA19-5486.

There are three different methods of supporting DASD on AS/400 systems:

- Multifunction IOP (MFIOP)
- IOPs which support internal DASD units beyond the capability of the MFIOP
- External DASD

All of these have different types of DASD IOPs.

A.4.2.1 Service processor and MFIOPs

The AS/400 system uses an architecture which combines multiple functions within some IOPs. One of the IOPs with this ability is the base Multi-Function IOP (MFIOP). Not only is it an input/output processor, but it also has the service processor incorporated within its hardware. The MFIOP is a standard feature on Stage 2 AS/400 systems.

MFIOPs support up to 20 integrated DASD units. They also support tape devices and communication adapters, and in some cases, workstation adapters. These adapters most often attach the workstation to function as the system console.

The tape device attached to the MFIOP is used as the default Alternate IPL load source. One of the communication adapters is often used to connect to Electronic Customer Support (ECS).

Service processor

The service processor is similar to the system processor in how it operates. However, the functions it performs are quite different from those of the system processor.

The service processor functions are much the same for all models of the system. However, the logic performing those functions is packaged considerably different on the B30 through B70 models of the system than on other models of the system. On the B30 through B70 models, the service processor functions are performed by a processor that is more or less dedicated to just those functions. When service processor functions are not performed, that processor is idle. On other models of the system, the service processor functions are performed by a Multi-Function Input/Output Processor (MFIOP). The MFIOP provides service processor functions on the system, and serves as the IOP for some of the I/O devices on the system.

Multifunction IOP (MFIOP)

All AS/400 systems must have a base MFIOP. The MFIOP type is determined by the workstation controller requirements.

Prior to OS/400 Version 4, all MFIOPs use SPD technology. Beginning with OS/400 V4, some MFIOPs are SPD and some are PCI. Which technology is used depends on the model to which it is referred.

Terminology tip

A base MFIOP comes standard on all AS/400 models. Beginning with the AS/400e series systems, however, other IOP cards support several functions. So the term MFIOP is not limited to the base MFIOP. All references to MFIOP in this redbook refer to the base MFIOP. The Integrated PC Server and Integrated Netfinity Server are later generation MFIOPs.

SPD MFIOPs

All DXX, EXX, FXX, 2XX, 3XX, 4XX and 5XX models of the AS/400 system use SPD MFIOPs. Refer to Figure 147 on page 182 for a diagram. These MFIOPs incorporate both the service processor and a SCSI bus (on which the load source disk unit resides, on PowerPC models the alternate load source CD-ROM resides, and one alternate load source tape unit can reside). The MFIOP incorporates up to two communication ports and may incorporate a workstation adapter port.

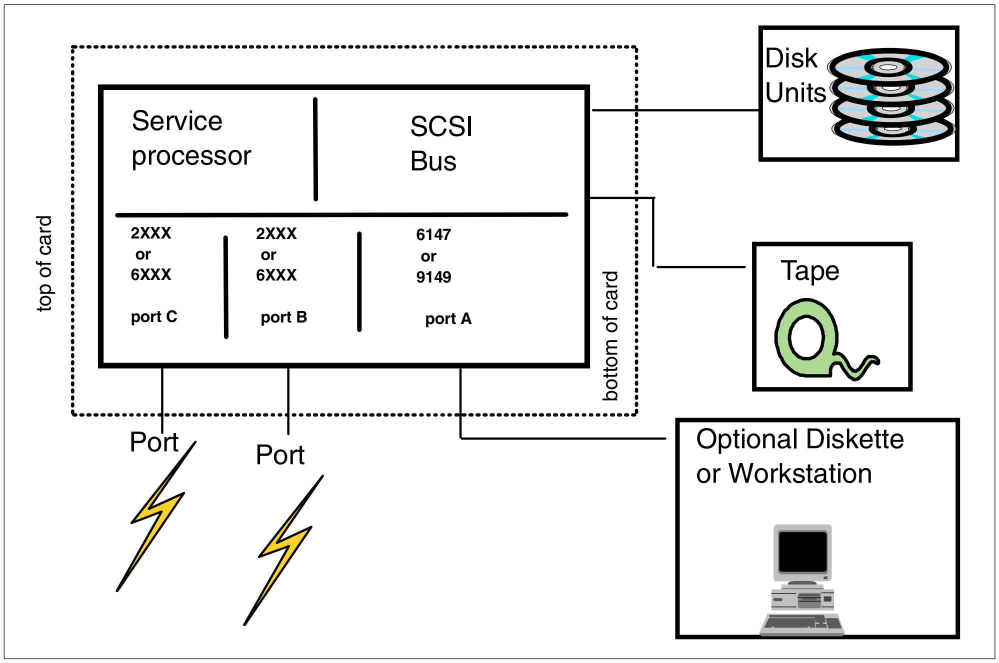


Figure 147. SPD MFIOP

For additional detail regarding the earlier technology IOPs, refer to the *AS/400 System Builder,* SG24-2155. Refer to the *AS/400e System Handbook,* GA19-5486, for the current version and release supported models.

Advanced MFIOP

With the introduction of the e-Series model 6XX models, some models continued to use SPD base MFIOPs and other models moved to the PCI base MFIOP. The PCI-based MFIOP was integrated into the system.

Removable PCI feature adapters determine what features the MFIOP supports. In effect, the PCI MFIOP became a system board that PCI adapters plug into.

Beginning with OS/400 V4R1, the base MFIOP gained many advanced functions. This includes the SPD MFIOPs used in the Models Model X40s, X30s and X50s and also the PCI MFIOPs used in the models X20s. Refer to Figure 148 to observe these functions.

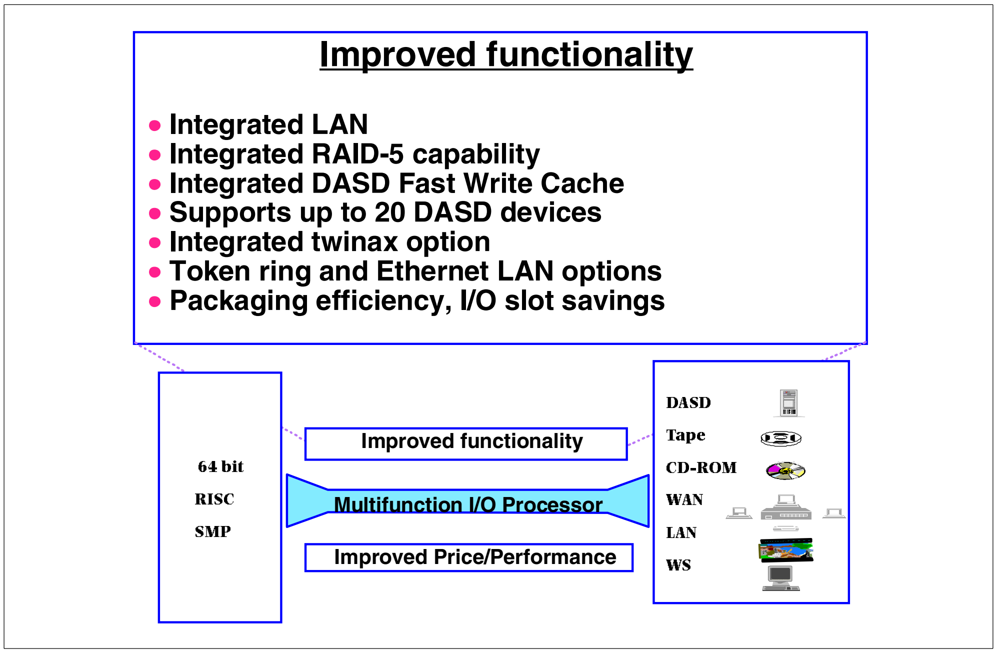


Figure 148. Advanced MFIOP

PCI Multifunction IOP for all models without SPD MFIOPs

Beginning with OS/400 V4R1, models of AS/400 system from the smallest up through the model S20/620 use the integrated base MFIOP. Only models X30 and above use SPD. The transition to PCI continues to the future.

Configuration tip

With PCI technology, there must be careful planning about where to physically place PCI adapters. If configuration rules are not followed, feature adapters and their supported devices may not function and/or may not show up in hardware configuration screens.

One configuration rule is the minimum Version/Release required to support the installed hardware components. Devices, adapters or features may not function, may not show up in hardware configuration screens, or may work unpredictably unless the prerequisite OS/400 code is installed.

These problems can appear to be hardware component failures. However, the hardware component is not failing. It is a configuration or installation failure.

For help in planning placement of IOAs in a PCI board, refer to the *AS/400 Installation Guide* for your system model.

For a listing of IOPs supporting disk, refer to the *AS/400e System Handbook,* GA19-5486, for products available for the current product line, or *AS/400 System Builder,* SG24-2155, for all products since the AS/400 system inception in 1988.

A.4.2.2 Considerations when installing DASD IOPs

The physical placement of an IOP affects the performance of devices attached to it, as well as other controllers on the same bus. Consider the following rules regarding DASD IOPs and placement for best performance:

- **Physical location**

  For proper configuration, locate DASD IOPs directly on the bus. Disk units experience timing failures when installed on a bus extension, because the bus extension's clocking function differs from that of the bus.

- **LIC on IOPs**

  Licensed Internal Code is loaded to the IOP during an IPL of the system. On occasion, an IOP loses the LIC stored in volatile memory on the IOP card.

  LIC is restored to some IOPs without an IPL. When the RESET parameter on the VRYCFG command is set to *YES, the VRYCFG command reloads LIC to workstation controller IOPs and communication controller IOPs.

  Note: The microcode for magnetic storage controller IOPs cannot be reset with a VRYCFG. An IPL is necessary to reload LIC to a magnetic storage controller IOP.

A.5 DASD I/O cables within the AS/400 system

DASD I/O cables connect the I/O devices to their respective IOP/IOAs. This section gives an overview of the device cables within the AS/400 system and special considerations pertaining to the device cables. Detailed information on interface features is in the *AS/400 System Builder,* SG24-2155, and *AS/400e System Handbook,* GA19-5486.

- **DFCI (Device Function Controller Interface)**

  I/O cables on earlier disk devices (9332, 9335, and 9336) carry data and information using DFCI.

  Note: The 9336 I/O cable is DFCI even though the 9336 drive itself is a SCSI drive. This is because the #6112 IOP driving the 9336 uses DFCI protocol.

- **SCSI (Small Computer System Interface)**

  I/O cables from the #6500 IOP to the 9337 models 0XX and 1XX DASD carry data/information using the SCSI protocol.

- **SCSI wide**

  The 9337 2XX and 4XX models use the SCSI wide protocol. The #6501, #6502 and #6512 IOP supports SCSI wide. SCSI wide has a wider bandwidth, thus enabling more data throughput. Because of the wider bandwidth, the cable that the 2XX, 4XX and 5XX models use is physically different than the model 0XX or 1XX. When upgrading from a 0XX or 1XX model to a 2XX, 4XX, or 5XX model, be sure you have the appropriate cable available.

- **Cable lengths**

  IBM I/O cables come in standard lengths. For example, the 9337 model 2XX IOP cables come in lengths of 2, 4, 6, or 20 meters. However, if the standard IBM cables do not meet the customer's needs, cables can be special ordered from IBM, as long as they meet industry standards. Industry standards, set by ANSI, for these same cables allow for 25 meters. If a customer needs the extra 5 meters, the longer lengths can be ordered through a request for price quotation (RPQ). IBM Marketing specialists can access the online sales information system to obtain more information on this.

- **RPQ cost covers more than just parts**

  When a RPQ is installed, the system or unit configuration changes. Sometimes the supporting rules and service support changes as a result. When an RPQ is ordered, these factors are considered. Because of this, the price of an RPQ can be more than the base cost of the parts involved.

Recommendation

Do not bypass the RPQ process by ordering the parts separately, thinking they will cost less. Installing a set of parts can leave the unit, as well as the system, in an unsupported status. RPQ provides the support structure required to protect the customer's investment.

A.6 AS/400 device controllers

An understanding of the device controllers within the AS/400 system is of use in planning DASD hardware configurations. This understanding is especially valuable when considering a proper configuration for a mirrored protected environment, because controller level is one of the protection levels. We discuss mirroring in 2.10.1, "Mirrored protection" on page 27.

A.6.1 Controller functions

The disk unit controller attaches to the I/O processor and handles the information transfer between the I/O processor and the disk units. Controllers synchronize the operations of such devices with the AS/400 system as a whole.

DASD controller characteristics

Disk units have different types of controllers based on disk type.

Current technology storage units have the controller either integrated into the storage unit's logic card, or into the IOP that is driving the storage unit.

The 280x disk units have the controller integrated into the disk units logic card. On these integrated storage units, the controllers are addressed by the system as 00 through 07 based on the unit's physical location on each SCSI bus. The physical location of the card book determines controller address (slot 23/24 or slot 25/26), as well as the position of the disk unit within the card book.

Usual replacement procedure for the 280x units is to replace the unit as a whole, (the logic card and head disk assembly (HDA)), rather than to replace the separate components. However, the logic card can be ordered separately if circumstances warrant.

Early disk units, such as the 9332, have their own controller housed within the device. It is a card located above the storage unit slots. The controller is addressed through a switch setting at the back of the box.

The 9335, model A01, is a separately housed controller used to control up to four 9335 model B01 disk units, each with two storage units. Because of controller contention, performance requirements can dictate that the A01 controllers attach no more than two B01 disk units. The A01 controller is addressed through a switch setting at the back of the box.

A.7 DASD terminology

Often our understanding of concepts is inhibited by the use of one term with multiple meanings. This section sets a basis for terms related to disk technology.

Disk units and storage units
Disk units are devices which are further divided into independently addressed sections known as *storage units*. The storage units are also known as *addressable storage units* (Figure 149). Think of storage units as actuators, because each storage unit is a single actuator.

The number of storage units per disk unit varies by disk unit type and model. For example, the 9332 Model 400 has two storage units. Storage units are addressable by the system based upon the possible number of storage units within a disk unit. For example, the 9335 disk unit has device addresses 0 through 7, and the 9332 disk unit has device addresses 0 through 1.

Head disk assemblies (HDAs)
Storage units are housed in a head disk assembly (HDA).

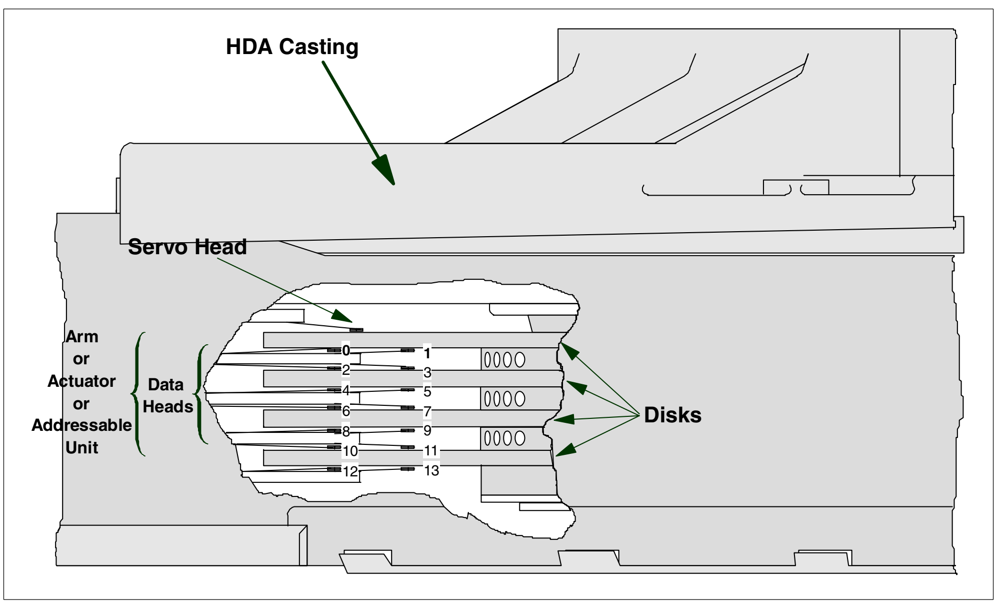


Figure 149. Addressable storage unit

On all current model disk technology, a single storage unit is enclosed in one HDA. The one-to-one ratio makes it possible to replace a single storage unit without affecting a second storage unit and its data.

In some examples, multiple storage units are enclosed in a single HDA. For example, the 9335 has two storage units per HDA.

DASD tracks

Information is written on narrow rings on disk platter surfaces. Each ring is called a *track*. There are numerous tracks on a given disk platter surface. When a read/write head needs to either read information from or write information to a disk unit, it positions itself over a single track.

Refer to Figure 150 for a visual representation of the DASD track as it relates to the individual platter.

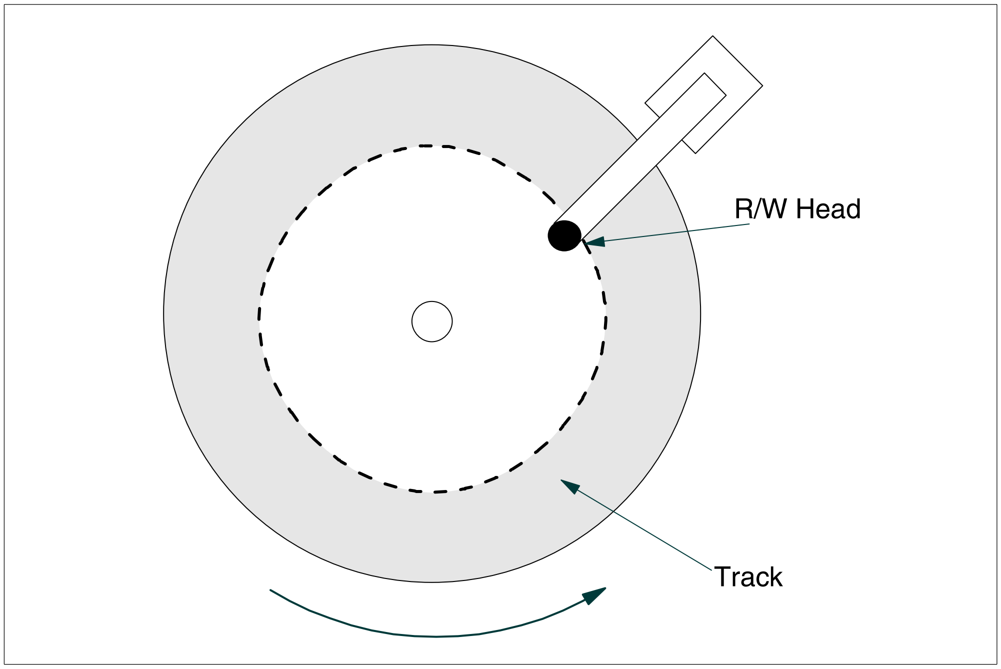


Figure 150. DASD track

DASD cylinders

To improve performance, several heads are manufactured in tandem on a single access mechanism (often referred to as an *arm*). Multiple parts of an object are written or read simultaneously as each head reads “its” track.

The collective total of a given track on each platter is called a *cylinder*. For example, track 1 on the first platter along with track 1 on all other platters within the same HDA, comprise cylinder one.

Refer to Figure 151 on page 188 for a graphic, which shows the relationship between a track and cylinder.

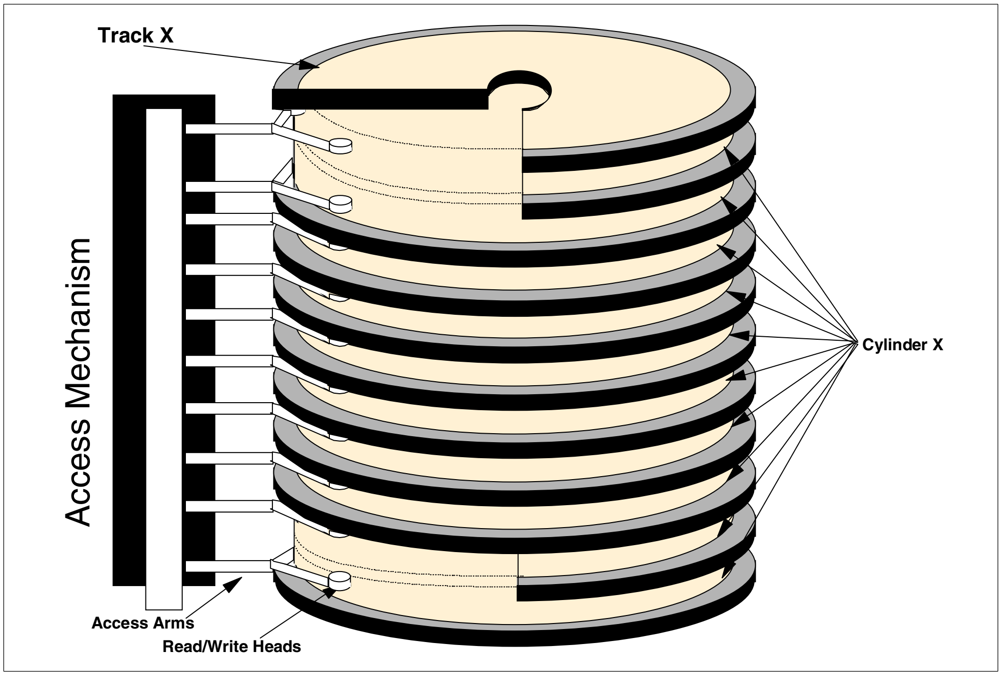


Figure 151. DASD cylinder

DASD sectors

Each data track is divided into several sectors. At the beginning of the group of sectors is an index mark to identify the beginning of the sectors on a track. Refer to Figure 152 for a visual representation of this arrangement.

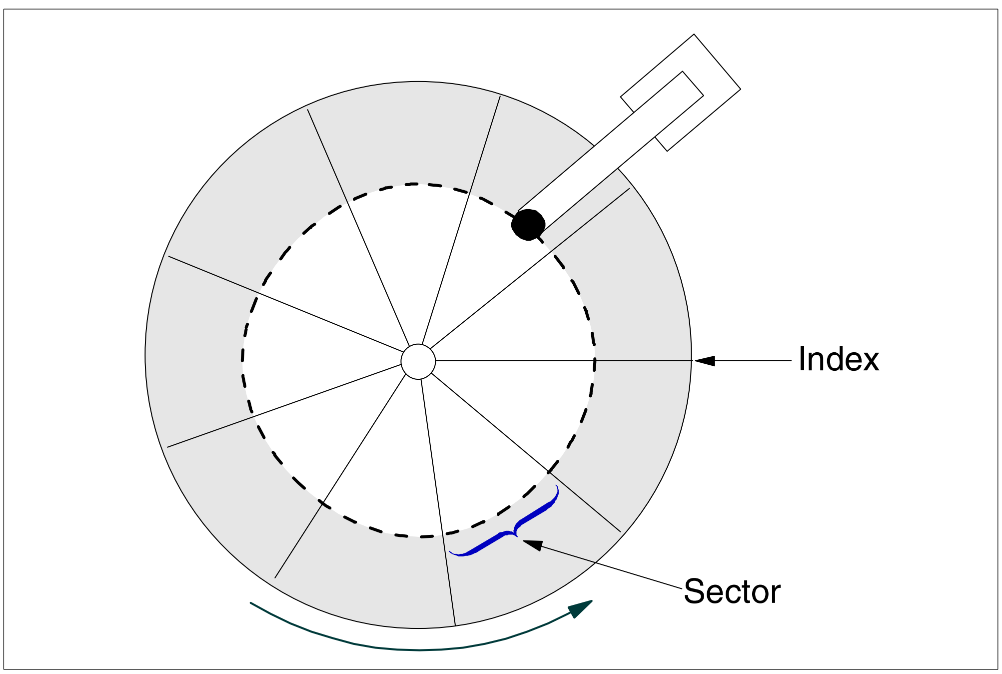


Figure 152. DASD sector

The type of IOP the DASD is attached to determines the sector format the data is written in. Figure 153 shows the two types of formats used on DASD attached to the AS/400 system.

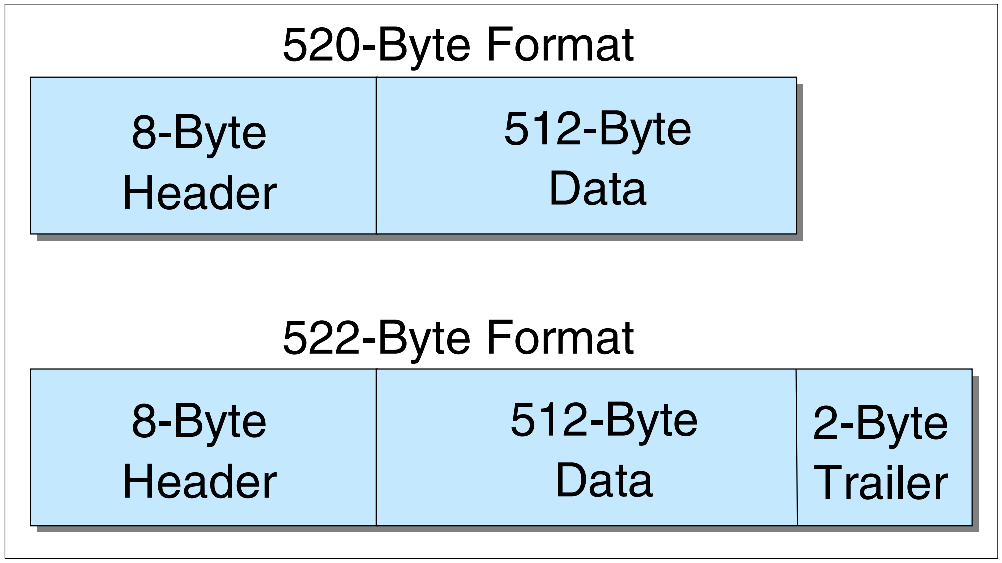


Figure 153. DASD sector format

The 520-byte format is associated with internal DASD attached to IOPs installed in the CISC platform, with one exception: The 520-byte format structure is used with the internal DASD attached to the MFIOP up through the AS/400 5xx models.

The 522-byte format is used for disk units that have compression enabled, and for RAID-capable internal DASD. RAID-capable disk units are those disks installed in a #5051 or a #5052 (when a RAID-capable IOP is used). DASD using the 520-byte structure does not support RAID. The extra two bytes of information in the trailer area denote this to the system.

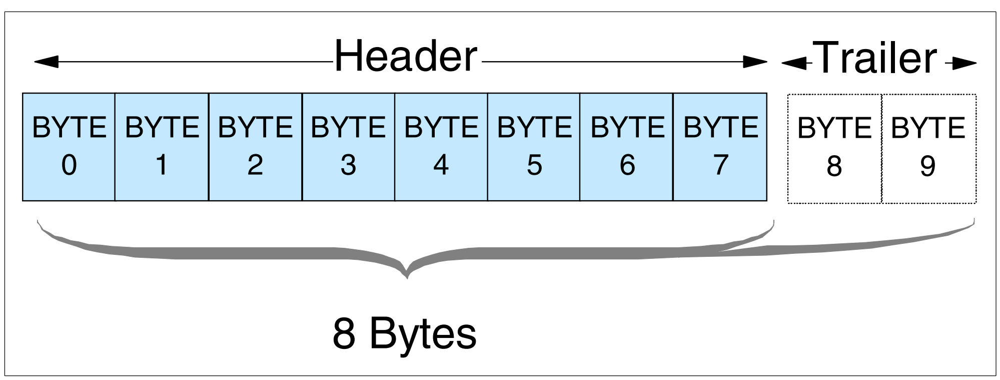


Figure 154. Header information

Figure 154 provides a visual representation of the header area found on a disk. The header contains information about the sector, for example, that the data on the sector is unprotected, checksummed, DPY protected, or mirror protected.

There are bits that contain flags to indicate if data within the sector is damaged. There is an address for the sector. And there is much more sector information contained within the header.

Depending on the IOP card that the drive is attached to, the disk is formatted to either a 520- or 522-byte format. The sector format is controlled by the code on the IOP or IOA and is transparent to storage management. Each format creates an 8-byte header. The data is always 512 bytes. The extra 2 bytes on the 522-byte format are used internally. From a user perspective, the block size is 512, always.

Most of the newer IOPs, when running the later versions of LIC, use a 522-byte format. These newer IOPs can read from or write to a drive that has been previously formatted to 520, but the drive does not take advantage of DPY or caching advantages that the IOP card could have provided. The compression sector size is 522 bytes.

A.7.1 Read/Write I/O process

Refer to Figure 155 through Figure 158 on page 192 for the steps of the read and write process. These figures describe and help visualize the read/write process.

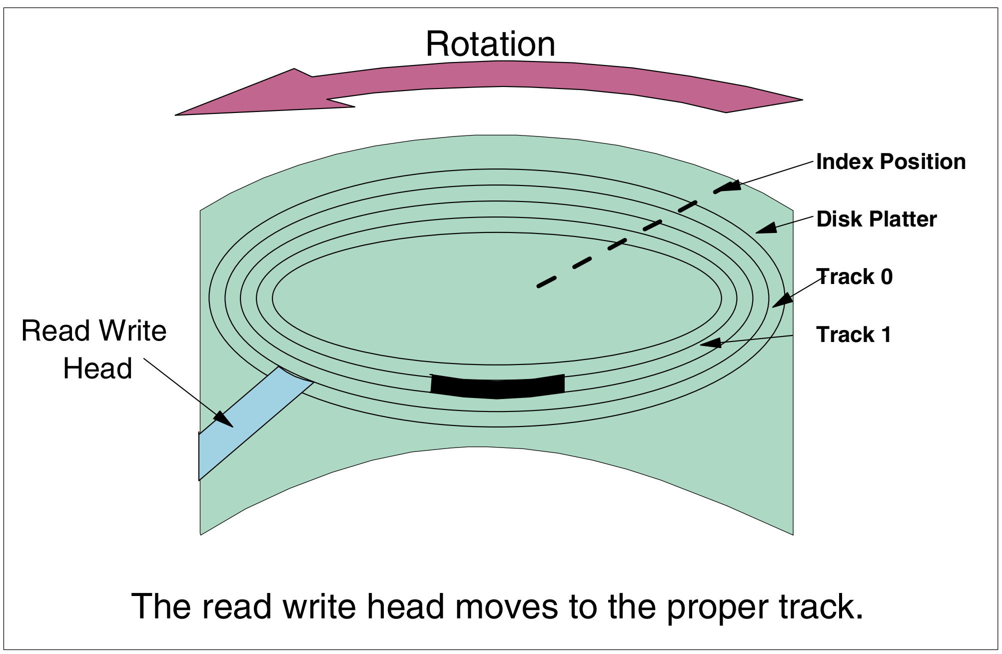


Figure 155. Read/write head move (seek) (1)

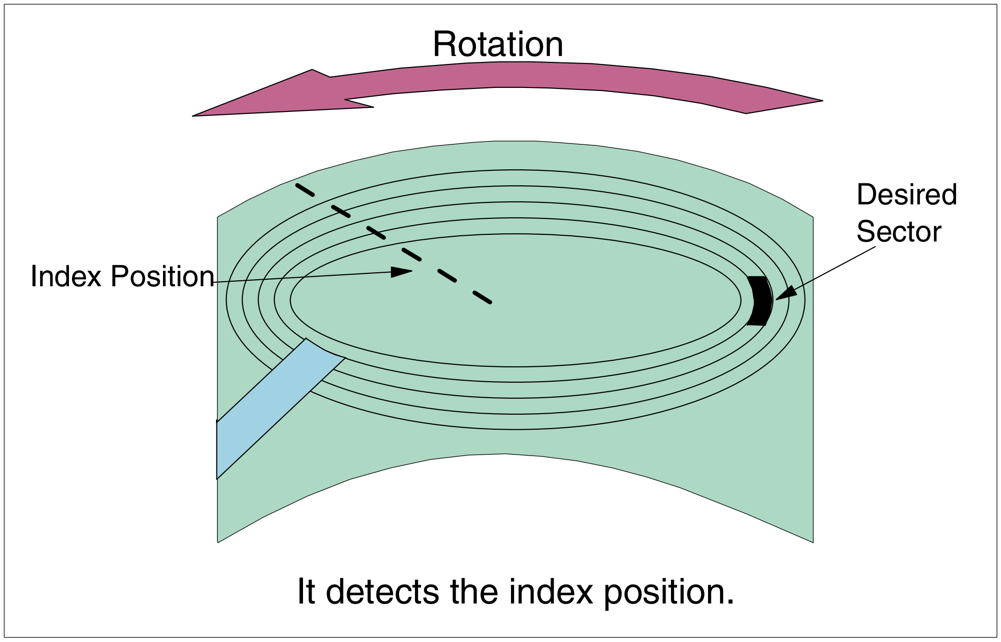


Figure 156. Read/write index detection (2)

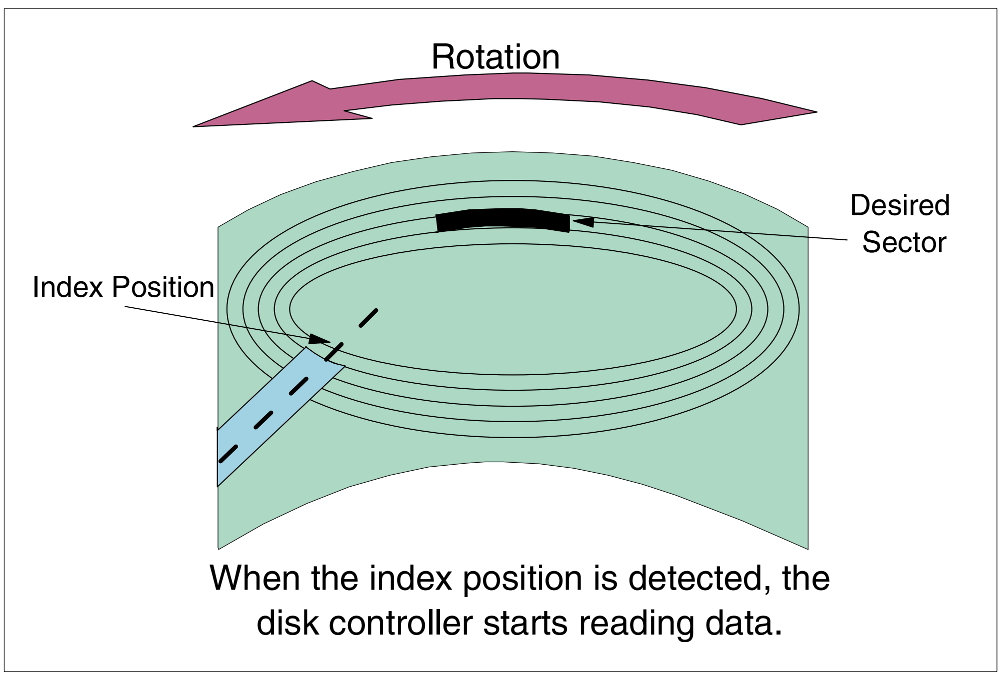


Figure 157. Read/write data read (3)

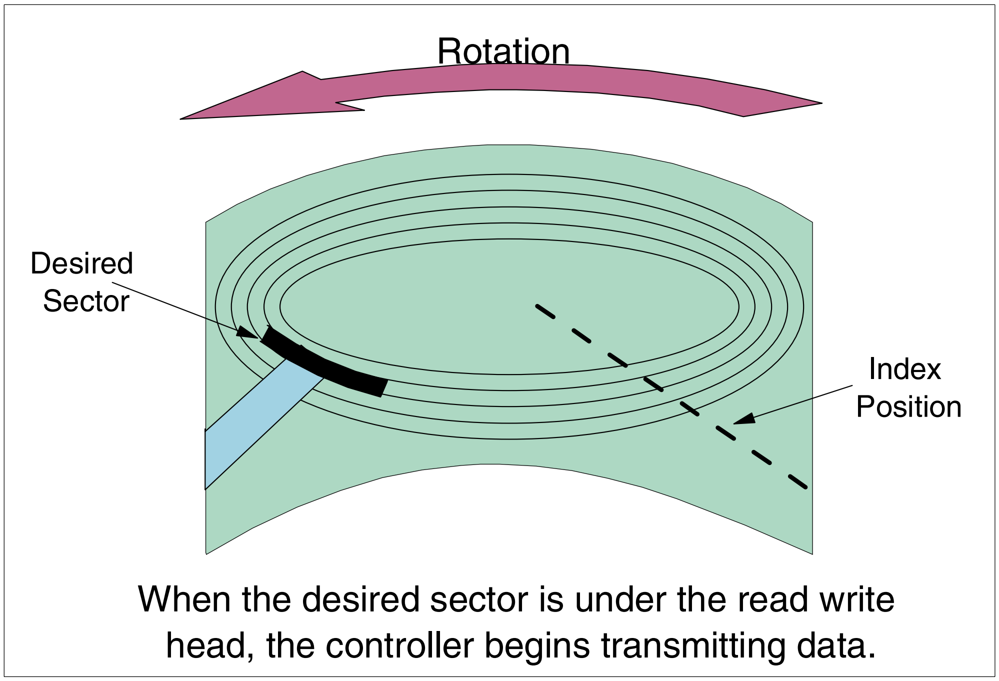


Figure 158. Read/write data transmission (4)

Appendix B. Storage management terms used in SLIC

This appendix provides descriptions of some frequently used AS/400 storage-related terminology:

Address The location in the storage of a computer where particular data is stored.

Auxiliary Storage Pool (ASP)
A group of storage units defined from the disk units that make up auxiliary storage. ASPs provide a means of isolating certain objects on a specific disk unit as a method to prevent the loss of data due to disk media failures on other disk units.

Extent This is a variable-sized piece of DASD to which a virtual address range has been assigned.

Free space directory
This is a pageable machine index which tracks all unassigned sectors on all disk units. It is used in CISC systems only.

Magnetic tape
Plastic tape coated on one side with a magnetic material that stores information as varying patterns of magnetization.

Main storage pool
A division of main storage which allows the user to reserve main storage for processing a job or group of jobs, or to use the pools defined by the system. As of OS/400 V4R4, up to 64 pools can be set up in one AS/400 system.

Page (noun)
A page is made up of eight disk sectors (4 KB) or 4 KB of main memory.

Page (verb)

An action of moving a page of data from main memory to disk or vice versa.

Permanent address
An address used to locate a piece of data that survives an IPL.

Permanent directory
Used by storage management to keep track of permanent addresses currently being used and their corresponding locations on DASD.

Real address (storage)
Main storage.

Sector The smallest amount of information that can be written to or read from a disk or diskette during a single read or write operation. A sector consists of 512 bytes of data and an 8-byte sector header to give a total size of 520 bytes. There are some other bits that belong to hardware (used when disk compression is activated, for example), but they can be ignored for the purposes of storage management.

Segment Identifier (SID)
Refers to the 4 or 5 high-order bytes of an 8-byte virtual storage address. The big SID is composed of a 5-byte SID and 3-byte offset. It encompasses an address range of 16 MB with 4096 occurrences of 4

© Copyright IBM Corp. 2000

KB pages. The little SID consists of 5 bytes called the Segment Group Identifier (SGID), which addresses 16 MB of storage. One SGID consists of up to 256 little SIDs, each of which addresses 64 KB of storage. The number of little SIDs is expressed in byte 5 (bits 40 to 47) of the address.

Static directory
This is used to store VLIC persistent data such as the disk locations of the permanent directory and so on.

Storage
Devices such as disks and tapes that store data magnetically or optically. Although slower than a computer's internal electronic memory, storage devices provide virtually unlimited capacity and preserve data integrity.

Temporary address
An address used to locate a piece of data that does not survive an IPL.

Temporary directory
Used by storage management to keep track of temporary addresses currently being used and their corresponding locations on DASD.

Temporary library
A library that is automatically created for each job to contain temporary objects that are created by the system for that job. The objects in the temporary library are deleted when the job ends. The system name for the temporary library is QTEMP.

Temporary objects
Objects, such as data paths or compiler work areas, that are automatically deleted by the system when the operating system is loaded.

Virtual address (storage)
An addressing scheme that allows external disk storage to appear as main storage.

Appendix C. Non-removable storage (DASD)

In this section, we identify and describe the various DASD options available for AS/400 system installation. Topics include both new and old DASD that are supported attached to current AS/400 systems.

C.1 DASD available for the AS/400 system

The listed DASD in this section include those supported in current technology AS/400 systems. For information regarding DASD supported in previous technology AS/400 system platforms, refer to the *AS/400 System Builder*, SG24-2155.

Available disk enclosures for the AS/400 systems are discussed in the following sections. Internal and external storage options are discussed.

C.1.1 280X

The DASD described in this section are not all supported in current AS/400 systems. The #2800 and #2801 repackaged DASD are not supported in the current AS/400e series systems. The #2802 DASD is supported in the AS/400e series.

The internal DASD storage units are available in three capacities: 1.3 GB, 988 MB, and 320 MB. Each card book contains two storage units of the same capacity. The AS/400 system has a minimum configuration of 1.2 GB of internal storage. This is accomplished with two #2800 card books, one #2801, or one #2802 card book. If a #2801 or #2802 card book is used as the base configuration, then an additional, optional #2800, #2801 or #2802 can be included. The system is capable of accommodating a maximum of two card books.

As the AS/400 DASD evolves, more and more circuitry that was formerly on separate field replaceable units (FRUs) are a part of the base unit. For example, the read/write control cards that were attached to the 9335 HDA are included in the logic card of the #2800. The controller that is housed in a separate model of the 9335 is also included in the same logic card on the 2800. The individual disk units have become an entire separate subsystem within themselves. The 9337 High Availability (HA) model (another term sometimes used in place of device parity protection (DPY) models) is an example of this.

C.1.2 9337 DASD

While 9337 DASD can no longer be ordered new from IBM, they are the only external storage subsystems still supported to attach to AS/400 systems 7xx models. The characteristics remain the same on 9337 DASD regardless of the platform to which it is attached.

Refer to Figure 159 on page 196 to observe some of the physical characteristics of the 9337 subsystem.

© Copyright IBM Corp. 2000

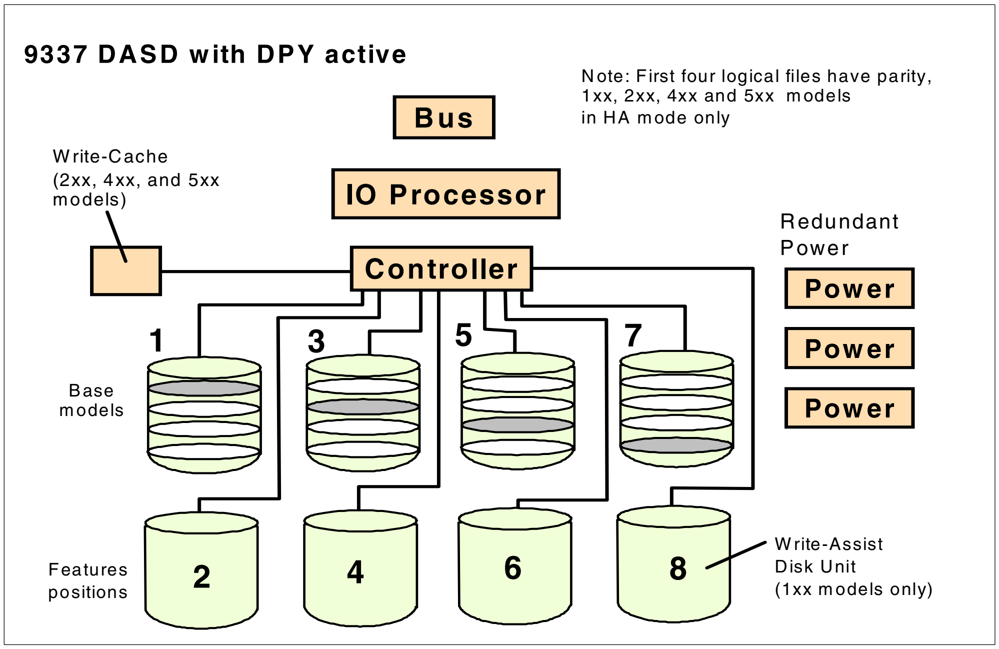


Figure 159. 9337 DASD

The supported models are those that attach to the feature #6501 Magnetic Storage Controller IOP only. Each #6501 IOP supports up to two of the 9337-2XX, 4XX or 5XX models.

Every 9337 model has a built-in storage controller as well as redundant power supplies which permit continued operation if one of the three power supplies fails. All 9337 models use 3.5-inch disk technology and offer better performance than previous external DASD models. When in device parity (DPY) mode also known as High Availability (HA) mode, a non-volatile Write Cache on the storage controller of the 2XX, 4XX and 5XX models takes the place of the write assist drive (WAD) storage unit of the 1XX models. This provides better performance.

It is important to be aware that on the device parity protected models, not all the physical capacity is usable for customer data, and is not observable on disk configuration capacity screens. The base storage units (those that contain the parity stripes) display a lower capacity than the others. The capacity used by the parity stripes is deducted from the total physical capacity. The capacity shown on system displays is the capacity available for customer data.

For additional detail per subsystem, refer to the *AS/400 System Builder*, SG24-2155.

9337 Dynamic RAID Spare function (Hot spare)

The Dynamic RAID Spare function, otherwise known as a *hot spare*, allows one of the disk drives to be assigned as a spare in the 9337 subsystem when in RAID-5 mode. It is not addressable by the AS/400 system, thus reducing the maximum number of addressable disk drives in the 9337 from eight to seven.

In the event of a single disk drive failure in a 9337 with hot spare, the 9337 automatically and immediately begins restoring the failed disk drive data to the hot spare, again by using the parity data areas from the other disk drives.

A customer can activate a hot spare feature in RAIDable 9337s when RAID is activated. Hot spare is not an option if mirror protection is used. In a 9337 with hot spare, if a single disk unit fails, the data of the failed unit is rebuilt on the spare drive from the parity stripes of the others. This is done automatically and may often be completed before the service representative arrives.

Since the data rebuild process can begin before the failed disk drive is physically replaced, it reduces the time that the 9337 is in the “exposed” mode to the time it takes to rebuild the hot spare. The “exposed” mode is when there is the potential of another disk drive failing in the same 9337 which could cause the 9337 to cease operation. After physical replacement of the failed drive, it becomes the new hot spare.

This Dynamic RAID Spare Function is also available on 9337 210, 215, 220, 225, or 240 models that do not have the 4M cache installed on the controller as an RPQ (843795).

How to define a hot-spare configuration is documented in the *Backup and Recovery Guide*, SC41-5304.

C.1.3 Enterprise Storage Server

The Enterprise Storage Server (ESS) is a Storage Area Network (SAN) storage solution. It supports fibre-channel attachment. Capacity ranges from 400 GB to over 11 TB. The ESS architecture supports high availability requirements by using redundant components. Date replication services extend access to data, while using a concurrent copy. Rapid data duplication provides extensive capabilities to serve, manage, and protect information in a 7 by 24 environment. The ESS incorporates and builds on the capabilities of the Versatile Storage Server, as discussed in the following section.

The Enterprise Storage Server (ESS) attaches to the AS/400 system using a #6501 Magnetic Storage Controller (SPD). The #6501 requires V3R1 or later. When attached to the #6501, the ESS emulates the 9337-5xx or 9337-5xx drive based on the size of the disk units installed.

For further information on the Versatile Storage Server, refer to the *AS/400e System Handbook*, GA19-5486.

C.1.4 Versatile Storage Server

With the IBM Versatile Storage Server (VSS), disk storage can be consolidated to use Web-based management data sharing for similar servers and dynamic capacity allocation function. The VSS delivers centralized management and stored data. It also provides sharing of disk storage for a variety of UNIX, Windows NT, and AS/400 servers. Built using the 7133 Serial Disk Subsystem, centralized management is simplified using IBM StorWatch, with a Java-compliant Internet browser.

The primary enclosure of the Versatile Storage Server includes the storage server, two 7133 drawers, and the power control system. There is space for two additional 7133 disk drawers. The primary enclosure can be combined with up to two 2105-100 expansion enclosures.

The VSS attaches to the AS/400 system using a #6501 Magnetic Storage Controller (SPD). The #6501 requires V3R1 or later. When attached to the #6501, the Versatile Storage Server emulates the 9337-580 or 9337-590 based on the size of the disk unit installed.

Notes

- Each connection to an AS/400 system using a #6501 supports a maximum of sixteen disk units. There can be a maximum of eight #3001 Host Interface Adapters per 2105 VSS. If eight #6501s are attached to eight #3001s, this dedicates the 2105 VSS to the AS/400 system and gives a maximum capacity of 536.3 GB when emulating 9337-580s, 1099.5 GB when emulating 9337-580s, and 1099.5 GB when emulating 9337-590s.
- In most cases, the minimum cache memory of 512 MB is best for use with the AS/400 system, but the expert cache function of OS/400 normally provides better performance. The Versatile Storage Server Expansion Enclosure includes space for seven 7133-010, 7133-020, 7133-D40 drawers and a power control system.

C.2 Internal DASD: SPCN versus Non-SPCN

SPCN or non-SPCN packaged DASD can be installed in DASD positions controlled by the MFIOP, only if working with an SPCN system. If using a non-SPCN system (Model 300), you cannot move SPCN packaged DASD into the MFIOP supported slots because this will damage the SPCN packaged DASD.

Releases beyond V3R2 allow concurrent maintenance of DASD when protection is in place for non-SPCN systems.

If you have a non-SPCN or SPCN tower with V4R1 or later, concurrent DASD repair is available, as that function is built into the SLIC code. V4R1 can only be installed on 4XX models or above because V4R1 can only operate on a PowerPC version of hardware.

Due to separate removable tray assemblies and separate power supplies, concurrent repair of AS/400 internal disks is available for the first time with the 3XX models. Concurrent repair allows a failed storage unit, which has been protected by RAID-5 or mirroring, to be replaced by service personnel while the system is still running. Redundant power supplies in the system (model 3xx and higher) provide backup power for the internal drives, should one power supply fail.

Note: The version and release references in the following list of IOPs indicates minimum level required for that IOP.

SPD IOPs available for internal DASD include:

- #6502: 2 MB cache RAID/Mirroring/Unprotected (V3R2)
- #6530: No cache Mirror protection only (V3R2)
- #6512: 4 MB cache RAID/Mirroring/Unprotected (V3R6, V3R7)
- #6532: 4 MB cache RAID/Mirroring/Unprotected (V4R1)

- #6533: 4 MB cache RAID/Mirroring/Unprotected (V4R2), integrated hardware disk compression enabled with V4R3
- #9751 MFIOP with RAID (Ultra SCSI) V4R1
- #9754 MFIOP with RAID (Ultra SCSI) V4R2, integrated hardware disk compression enabled with V4R3

Refer to A.4, “AS/400 IOP and IOA” on page 179, for details of the IOP cards and their capabilities.

For a complete listing of internal DASD, refer to *AS/400 Installation and Upgrade*, SY44-5950. Locate and reference Appendix C, “Configuration Rules for AS/400 Models”, and find the chart for internal DASD devices. Be certain to reference the correct level of documentation for your system.

C.3 Expansion Towers

Expansion towers expand the capacity of the AS/400 system by providing housing for additional storage devices. The towers are architectural for either SPD or PCI technology.

For further details and to understand which disk and tape units are supported, refer to the *AS/400e System Handbook*, GA19-5486.

C.3.1 #508x Storage Expansion Tower

The technology found in this tower is strictly SPD, regarding the bus structure as well as the IOPs that plug into the card cage. This tower was first introduced in the CISC platform in the form of a #5060. It was designed to provide an additional storage tower. There is no room for workstation or communication or any other type of IOP.

Three slots are included in the card cage for SPD IOP cards. The first slot contains the bus receiver card for data and signal traffic to and from the AS/400 system unit it is attached to. The second slot is designated as the DASD IOP slot that controls up to 16 DASD found inside the frame of the tower. The third slot is designated to control the #5051 and #5052 that provide slots for an additional 16 DASD.

The total number of DASD slots available in this box with a storage expansion unit installed is 32 DASD. This tower is considered an additional system I/O bus and used in place of a #507x system expansion tower.

The storage expansion towers of today (#508x) operate under the same characteristics as described here.

C.3.2 #5065 Storage/PCI Expansion Tower

This technology, although PCI at the tower, uses the optical bus connections, which is SPD to the currently available AS/400 system units.

The #5065 Storage/PCI Expansion Tower is a disk expansion tower with PCI adapter support. With the #5065, you have support for lower-cost PCI cards and faster disk units in denser packaging, along with Ultra2 SCSI performance. It is capable of supporting 45 one-inch Ultra2 Disk units, and has 12 PCI IOA slots (supporting up to three PCI RAID Disk Unit Controllers) and two removable media slots.

The #5065 Tower requires V4R4, and attaches to models 720, 730, and 740 (plus 6XX and SXX) via an SPD optical bus. With the #2748 IOP installed, the Read Cache Device can be installed and utilized. For additional restrictions and considerations for use of this tower refer to the *AS/400e System Handbook*, GA19-5486, or the *AS/400 System Builder*, SG24-2155.

C.3.3 #5066 1.8 I/O Tower

The #5066 1.8 I/O Tower includes two 1063 Mbps optical bus cards, various cables, including optical cables, and the 1.8M I/O Tower. The #5066 provides 24 PCI IOA slots, space for 90 disk units, space for four removable media devices, battery backup, redundant/hot swap power supplies, and two PCI LAN/WAN/Workstation IOPs (CCIN #2824). It supports Ultra2 SCSI disk units. The #5066 also supports up to four removable media devices (internal tape or CD-ROM).

The #5066 is actually two #5065 Storage/PCI Expansion Towers installed in a #5066 1.8M I/O Tower. It reports to the system as two #5065s.

Appendix D. WRKDASD sample code

This appendix contains the sample code and programming instructions required to create the WRKDASD command described in 8.4.3, “Using APIs” on page 149. The instructions to obtain this code is found in Appendix E, “Using the additional material” on page 217.

1. Create a library called WRKDASD to contain the Work with DASD command and associated files and programs:

```
CRTLIB LIB(WRKDASD) TEXT('Work with DASD command')
```

2. Create CMD, CL, DDS and RPG source files using the following commands:

```
CRTSRCPF FILE(WRKDASD/QCMDSRC) TEXT('CMD Source Members')
CRTSRCPF FILE(WRKDASD/QCLSRC) TEXT('CL Source Members')
CRTSRCPF FILE(WRKDASD/QDDSSRC) TEXT('DDS Source Members')
CRTSRCPF FILE(WRKDASD/QRPGLESRC) TEXT('RPGILE Source Members')
```

3. Create a source member, with member type PF, in the QDDSSRC source file for each of the following files:
   - DATAFILE (Figure 161 on page 202)
   - DBIOFILE (Figure 162 on page 203)
   - HOLDFILE (Figure 163 on page 203)
4. Create the physical files using the following commands:

```
CRTPF FILE(WRKDASD/DATAFILE) SRCFILE(WRKDASD/QDDSSRC)
CRTPF FILE(WRKDASD/DBIOFILE) SRCFILE(WRKDASD/QDDSSRC)
CRTPF FILE(WRKDASD/HOLDFILE) SRCFILE(WRKDASD/QDDSSRC)
```

5. Create a source member, with member type DSPF, in the QDDSSRC source file for the WRKDASD display file (Figure 164 on page 204).
6. Create the WRKDASD display file using the following command:

```
CRTDSPF FILE(WRKDASD/WRKDASD) SRCFILE(WRKDASD/QDDSSRC) DFRWRT(*NO)
```

7. Create a source member, with member type CLP, in the QCLSRC source file for the GETDBIO program (Figure 166 on page 205).
8. Create the GETDBIO program using the following command:

```
CRTCLPGM PGM(WRKDASD/GETDBIO) SRCFILE(WRKDASD/QCLSRC)
```

9. Create a source member, with member type RPGLE, in the QRPGLESRC source file for the WRKDASD program (Figure 177 on page 216).
10. Create the WRKDASD RPG program using the following command:

```
CRTBNDRPG PGM(WRKDASD/WRKDASD) SRCFILE(WRKDASD/QRPGLESRC)
```

11. Create a source member, with member type CMD, in the QCMDSRC source file for the WRKDASD command (Figure 178 on page 216).
12. Create the WRKDASD command using the following command:

```
CRTCMD CMD(WRKDASD/WRKDASD) PGM(WRKDASD/WRKDASD) SRCFILE(WRKDASD/QCMDSRC)
```

After completing these instructions you, should end up with the following objects in your library.

© Copyright IBM Corp. 2000

```
5769SS1 V4R4M0  990521                Display Library            14:50:42  Page 1

Library . . . . . . . . . . . . . :   WRKDASD
Type  . . . . . . . . . . . . . . :   PROD
Number of objects . . . . . . . . :           11
ASP . . . . . . . . . . . . . . . :     2
Create authority  . . . . . . . . :   *SYSVAL
Text description  . . . . . . . . :   Work with DASD command

  Object      Type      Attribute             Size  Description
  GETDBIO     *PGM      CLP                  36864  Get Database I/O Information
  WRKDASD     *PGM      RPGLE               335872  Work with DASD by job
  DATAFILE    *FILE     PF                   45056  Data File for WRKDASD
  DBIOFILE    *FILE     PF                   49152  Database File I/O Information
  HOLDFILE    *FILE     PF                   45056  File that holds a copy of DATAFILE
  QCLSRC      *FILE     PF                   20480  CL Source Members
  QCMDSRC     *FILE     PF                   20480  Command Source Members
  QDDSSRC     *FILE     PF                   69632  DDS SOURCE MEMBERS
  QRPGLESRC   *FILE     PF                   90112  RPG Source Member File
  WRKDASD     *FILE     DSPF                  8192  Display File for WRKDASD Program
  WRKDASD     *CMD                            4096  Work with DASD Command

                    Total size :           815104

                  * * * * *  E N D   O F   L I S T I N G  * * * * *
```

Figure 160. Library contents of WRKDASD

```
SOURCE FILE . . . . . . .  WRKDASD/QDDSSRC
 MEMBER  . . . . . . . . .  DATAFILE

 SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
   100      A*****************************************************************
   200      A*                                                              *
   300      A*     This file contains all of the information required for    *
   400      A*     the WRKDASD display and report                           *
   500      A*                                                              *
   600      A*****************************************************************
   700      A*
   800      A          R DATAF
   900      A*
  1000      A            KEYFLD1        7  0      TEXT('Primary Key field')
  1100      A            KEYFLD2        7  0      TEXT('Secondary Key field')
  1200      A            QJOB          10         TEXT('Job Name')
  1300      A            QUSR          10         TEXT('User Name')
  1400      A            QJNUM          6         TEXT('Job Number')
  1500      A            QPROG         10         TEXT('Program Name')
  1600      A            QSBS          10         TEXT('Subsystem Name')
  1700      A            QIO            7  0      TEXT('Auxiliary I/O')
  1800      A            QDBIO          7  0      TEXT('Database I/O')
  1900      A            QSTOR          7  0      TEXT('Temporary Storage')
  2000      A          K KEYFLD1                  DESCEND
  2100      A          K KEYFLD2                  DESCEND
                                * * * *  E N D  O F  S O U R C E  * * * *
```

Figure 161. Source listing for the physical file DATAFILE

```
SOURCE FILE . . . . . . .  WRKDASD/QDDSSRC
 MEMBER  . . . . . . . . .  DBIOFILE

 SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
            A*****************************************************************
            A*                                                               *
            A*     This file contains the output from CPYSPLF of the spool   *
            A*     file produced by the WRKJOB OPTION(*OPNF) command         *
            A*                                                               *
            A*****************************************************************
            A*
            A          R DBIOF
            A*
            A            FLD1           1         TEXT('Spacer')
            A            DBFILE        10         TEXT('File Name')
            A            FLD2           2         TEXT('Spacer')
            A            DBLIB         10         TEXT('Library Name')
            A            FLD3           2         TEXT('Spacer')
            A            DBMBR         10         TEXT('Member Name')
            A            FLD4          14         TEXT('Spacer')
            A            DBTYPE         3         TEXT('File Type')
            A            FLD5           2         TEXT('Spacer')
            A            DBIO           7         TEXT('Database I/O')
            A            FLD6          71         TEXT('Spacer')
            A          K DBIO
            A          K DBFILE
                       * * * *  E N D  O F  S O U R C E  * * * *
```

Figure 162. Source listing for the physical file DBIOFILE

```
SOURCE FILE . . . . . . .  WRKDASD/QDDSSRC
 MEMBER  . . . . . . . . .  HOLDFILE

 SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
            A*****************************************************************
            A*                                                               *
            A*     This file holds a copy of the DATAFILE during             *
            A*     resequencing                                              *
            A*                                                               *
            A*****************************************************************
            A*
            A          R HOLDF
            A*
            A            KEYFLD1        7  0      TEXT('Primary Key field')
            A            KEYFLD2        7  0      TEXT('Secondary Key field')
            A            QJOB          10         TEXT('Job Name')
            A            QUSR          10         TEXT('User Name')
            A            QJNUM          6         TEXT('Job Number')
            A            QPROG         10         TEXT('Program Name')
            A            QSBS          10         TEXT('Subsystem Name')
            A            QIO            7  0      TEXT('Auxiliary I/O')
            A            QDBIO          7  0      TEXT('Database I/O')
            A            QSTOR          7  0      TEXT('Temporary Storage')
            A          K KEYFLD1                  DESCEND
            A          K KEYFLD2                  DESCEND
                       * * * *  E N D  O F  S O U R C E  * * * *
```

Figure 163. Source listing for the physical file HOLDFILE

```
SOURCE FILE . . . . . . .  WRKDASD/QDDSSRC
 MEMBER  . . . . . . . . .  WRKDASD

 SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
A*****************************************************************
A*                                                               *
A*     This is the display file used by the WRKDASD program      *
A*                                                               *
A*                                                               *
A*****************************************************************
A*
A* Command Key Definition
A*
A                                      CF03(03 'End of Program')
A                                      CF05(05 'Refresh')
A                                      CF06(06 'Sort by DB I/O')
A                                      CF07(07 'Sort by Temp Stor')
A                                      CF08(08 'Sort by Aux I/O')
A                                      CF12(12 'Cancel')
A*
A* Heading Screen
A*
A          R HEADING                   OVERLAY
A                                  1 33'Work with DASD'
A                                      DSPATR(HI)
A                                  3  2'Type options, press Enter.'
A                                      COLOR(BLU)
A                                  4  4'5=Work with Open Files   6=Work wi-
A                                      th Spool Files   7=Work with Job Lo-
A                                      cks'
A                                      COLOR(BLU)
A                                  6 57'Aux I/O  DB I/O  Temp'
A                                      COLOR(WHT)
A                                  7  2'Opt Job        User       Number P-
A                                      rogram    Subsystem  Request Reques-
A                                      t Storage'
A                                      COLOR(WHT)
A*
A*
A* Subfile Detail Screen
A*
A          R SUBFILE                   SFL
A*
A            OPT            1   I  8  3VALUES(' ' '5' '6' '7')
A            QJOB          10   O  8  6
A            QUSR          10   O  8 17
A            QJNUM          6   O  8 28
A            QPROG         10   O  8 35
A            QSBS          10   O  8 46
A            QIO            7  0O  8 57EDTCDE(Z)
A            QDBIO          7  0O  8 65EDTCDE(Z)
A            QSTOR          7  0O  8 73EDTCDE(Z)
A*
A*
A* Subfile Control Screen
A*
A          R CONTROL                   SFLCTL(SUBFILE)
A*
A  55                                  SFLDSPCTL
A  50                                  SFLDSP
A N50                                  LOCK
A                                      SFLSIZ(12)
A                                      SFLPAG(12)
A                                      ROLLUP(31 'Roll up')
A N60                                  ROLLDOWN(32 'Roll down')
A  59                                  SFLEND(*MORE)
A  60                                  SFLCLR
A                                      OVERLAY PROTECT
A            RCDNBR         2  0H      SFLRCDNBR(CURSOR)
A*
```

Figure 164. Source listing for the display file WRKDASD (Part 1 of 2)

```
     A                                22  2'F3=Exit     F5=Refresh     F6=Sort-
     A                                        by DB I/O   F7=Sort by Temp Stor'
     A                                       COLOR(BLU)
     A                                23  2'F8=Sort by Aux I/O   F12=Cancel'
     A                                       COLOR(BLU)
     A  91                            24  2'*** Retrieving Job Information  - -
     A                                       Please Wait ***'
     A  91                                   COLOR(RED)
     A  92                            24  2'*** Retrieving Database I/O inform-
     A                                       ation  -  Please Wait ***'
     A  92                                   COLOR(RED)
     A  97                            24  2'*** Already at top of list ***'
     A  97                                   COLOR(RED)
     A  98                            24  2'*** Already at bottom of list ***'
     A  98                                   COLOR(RED)
                         * * * *  E N D  O F  S O U R C E  * * * *
```

Figure 165. Source listing for the display file WRKDASD (Part 2 of 2)

```
SOURCE FILE . . . . . . .  WRKDASD/QCLSRC
 MEMBER  . . . . . . . . .  GETDBIO

 SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
/*********************************************************************/
/*                                                                   */
/*     This program creates a spool file containing open file        */
/*     information for the job specified by the input parameters:    */
/*                                                                   */
/*     QJOB  = Job Name                                              */
/*     QUSR  = User Name                                             */
/*     QJNUM = Job Number                                            */
/*                                                                   */
/*     It then copies the spool file to the physical file DBIOFILE   */
/*     and replaces the existing records.                            */
/*                                                                   */
/*     The spool file is then deleted.                               */
/*                                                                   */
/*********************************************************************/
/**/
             PGM        PARM(&QJOB &QUSR &QJNUM)

             DCL        VAR(&QJOB) TYPE(*CHAR) LEN(10)
             DCL        VAR(&QUSR) TYPE(*CHAR) LEN(10)
             DCL        VAR(&QJNUM) TYPE(*CHAR) LEN(6)

             OVRPRTF    FILE(QPDSPJOB) HOLD(*YES)
             DSPJOB     JOB(&QJNUM/&QUSR/&QJOB) OUTPUT(*PRINT) +
                          OPTION(*OPNF)
             MONMSG     MSGID(CPF0000)

             CPYSPLF    FILE(QPDSPJOB) TOFILE(WRKDASD/DBIOFILE) +
                          SPLNBR(*LAST)
             MONMSG     MSGID(CPF0000)

             DLTSPLF    FILE(QPDSPJOB) SPLNBR(*LAST)
             MONMSG     MSGID(CPF0000)
             DLTOVR     FILE(*ALL)

             ENDPGM
                         * * * *  E N D  O F  S O U R C E  * * * *
```

Figure 166. Source listing for the CL program GETDBIO

```
SOURCE FILE . . . . . . .  WRKDASD/QRPGLESRC
 MEMBER  . . . . . . . . .  WRKDASD

 SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
      F*********************************************************************
      F*                                                                   *
      F*  PROGRAM:      Work with DASD                                     *
      F*                                                                   *
      F*  LANGUAGE:     ILE RPG                                            *
      F*                                                                   *
      F*  DESCRIPTION:  This program:                                      *
      F*                                                                   *
      F*                (1) Retrieves a list of active jobs, using the     *
      F*                    QGYOLJOB API, and stores them in a userspace   *
      F*                    in the QTEMP library.                          *
      F*                (2) Retrieves open file information for each job,  *
      F*                    by using the CL program GETDBIO, and stores    *
      F*                    it in the database file DBIOFILE.              *
      F*                (3) Extracts the relevant information and stores   *
      F*                    it in an indexed physical file called DATAFILE. *
      F*                (4) Displays or prints the extracted information.  *
      F*                                                                   *
      F*                                                                   *
      F*                The parameters passed to the program are:          *
      F*                                                                   *
      F*                OUTPUT   (* = Screen, or *PRINT = Listing)         *
      F*                SUBSYS   (*ALL = All Subsysts Selected, or Name)    *
      F*                                                                   *
      F*                                                                   *
      F*  APIS USED:    QGYOLJOB - Gets the active jobs on the system      *
      F*                QUSROBJD - Checks if the user space exists         *
      F*                QUSCRTUS - Creates a user space                    *
      F*                QGYGETLE - Retrieves a records from the user space  *
      F*                                                                   *
      F*                                                                   *
      F*********************************************************************
      F*********************************************************************
      F*
      F*
      FWRKDASD   CF   E             WORKSTN SFILE(SUBFILE:RECNO) USROPN
      FDATAFILE  UF A E           K DISK    USROPN
      FHOLDFILE  IF   E           K DISK    USROPN
      FDBIOFILE  IF   E           K DISK    USROPN
      FQSYSPRT   O    F  132        PRINTER OFLIND(*INOF)
      F*
      D*
      DQCMDEXC          PR                  EXTPGM('QCMDEXC')
      D  CMD                         200A   OPTIONS(*VARSIZE) CONST
      D  CMD                          15P 5 CONST
      D*
      DCOMMAND          S             80A   DIM(2) PERRCD(1) CTDATA
      D*
      D****************************************************************
      D*
      D* Record structore for OLJB0200 format  (Receiver Variable)
      D*
      D****************************************************************
      D*
      DOLJB0200         DS
      D*
      D QJOB                    1     10
      D*                                    JOB NAME USED
      D QUSR                   11     20
      D*                                    USER NAME USED
      D QJNUM                  21     26
      D*                                    JOB NUMBER USED
      D QIJID                  27     42
      D*                                    INTERNAL JOB IDENTIFIER
      D QSTAT                  43     52
      D*                                    STATUS
      D QJTYPE                 53     53
      D*                                    JOB TYPE
      D QJSUB                  54     54
      D*                                    JOB SUBTYPE
```

Figure 167. Source listing for the RPG program WRKDASD (Part 1 of 11)

```
     D QRESI                  55     56
     D QJSTAT                 57     57
     D QRES2                  58     60
     D QPROG                  61     70
     D QDATA2                 73     73
     D QAUXIO                 77     80B 0
     D QSBSD                  81    100
     D QTMPST                101    104B 0
     D*
     D*****************************************************************
     D*
     D* Record structure for sort information format
     D*
     D*****************************************************************
     D*
     DSORT             DS
     D*
     D KFNUM                   1      4B 0 INZ(1)
     D KFSTAR                  5      8B 0 INZ(24)
     D KFLEN                   9     12B 0 INZ(4)
     D KFTYPE                 13     14B 0 INZ(9)
     D KFORD                  15     15    INZ('2')
     D KFRESV                 16     16    INZ(X'00')
     D*
     D*
     D*****************************************************************
     D*
     D* RECORD STRUCTURE FOR JOB SELECTION INFORMATION FORMAT
     D*
     D*****************************************************************
     D*
     DOLJS0100         DS
     D*
     D JNAME                   1     10    INZ('*ALL      ')
     D UNAME                  11     20    INZ('*ALL      ')
     D JNUM                   21     26    INZ('*ALL  ')
     D JTYPE                  27     27    INZ('*')
     D JRES                   28     28    INZ(X'00')
     D JPOFF                  29     32B 0 INZ(60)
     D JPNUM                  33     36B 0 INZ(1)
     D JAOFF                  37     40B 0 INZ(70)
     D JANUM                  41     44B 0 INZ(0)
     D JQOFF                  45     48B 0 INZ(74)
     D JQNUM                  49     52B 0 INZ(0)
     D JQNOFF                 53     56B 0 INZ(84)
     D JQNNUM                 57     60B 0 INZ(0)
     D JPJS                   61     70    INZ('*ACTIVE   ')
     D JAJS                   71     74    INZ('RUN ')
     D JJQS                   75     84
     D JJQN                   85    104
     D*
     D*
     D*****************************************************************
     D*
     D* RECORD STRUCTURE FOR RECEIVER VARIABLE DEFINITION FORMAT
     D*
     D*****************************************************************
     D*
     DVARDEF           DS
     D*
     D VNUM                    1      4B 0
     D V1FLEN                  5      8B 0
     D V1KEY                   9     12B 0
     D V1TYPE                 13     13
     D V1RES                  14     16
     D V1DLEN                 17     20B 0
     D V1DISP                 21     24B 0
     D V2FLEN                 25     28B 0
     D V2KEY                  29     32B 0
     D V2TYPE                 33     33
     D V2RES                  34     36
     D V2DLEN                 37     40B 0
     D V2DISP                 41     44B 0
     D V3FLEN                 45     48B 0
```

Figure 168. Source listing for the RPG program WRKDASD (Part 2 of 11)

```
D V3KEY                 49     52B 0
D V3TYPE                53     53
D V3RES                 54     56
D V3DLEN                57     60B 0
D V3DISP                61     64B 0
D V4FLEN                65     68B 0
D V4KEY                 69     72B 0
D V4TYPE                73     73
D V4RES                 74     76
D V4DLEN                77     80B 0
D V4DISP                81     84B 0
D V5FLEN                85     88B 0
D V5KEY                 89     92B 0
D V5TYPE                93     93
D V5RES                 94     96
D V5DLEN                97    100B 0
D V5DISP               101    104B 0
D*
D*
D*****************************************************************
D*
D* Record structure for error code format
D*
D*****************************************************************
D*
DQUSBN            DS
D*
D QUSBNB                 1      4B 0 INZ(100)
D QUSBNC                 5      8B 0
D QUSBND                 9     15
D QUSBNF                16     16
D QUSBNG                17    100
D*
D*
D*****************************************************************
D*
D* Record structure for keys format
D*
D*****************************************************************
D*
DKEYS             DS
D*
D KEY1                   1      4B 0 INZ(0601)
D KEY2                   5      8B 0 INZ(0602)
D KEY3                   9     12B 0 INZ(1401)
D KEY4                  13     16B 0 INZ(1906)
D KEY5                  17     20B 0 INZ(2009)
D*
D*
D*****************************************************************
D*
D* Record structure for List Information format
D*
D*****************************************************************
D*
DLSTINF           DS
D*
D LTOTAL                 1      4B 0
D LRETND                 5      8B 0
D LRHAND                 9     12
D LRLEN                 13     16B 0
D LCOMP                 17     17
D LDATE                 18     30
D LSTAT                 31     31
D LRES1                 32     32
D LILEN                 33     36B 0
D LFIRST                37     40B 0
D LRES2                 41     80
D*
D*****************************************************************
D*
D* Stand-Alone Field Definitions
D*
D*****************************************************************
```

Figure 169. Source listing for the RPG program WRKDASD (Part 3 of 11)

```
     D*
     D                 DS
     DAPINAM                      1     10
     DEXTATR                     31     40    INZ('QSYSLOBJP ')
     DRJOBDF                     68     75    INZ('OBJD0100')
     DSPCAUT                     76     85    INZ('*ALL      ')
     DSPCDMN                     86     95    INZ('*USER     ')
     DSPCINT                     96     96    INZ(X'00')
     DSPCNAM                     97    116    INZ('ACTJOB    QTEMP     ')
     DSPCREP                    117    126    INZ('*YES      ')
     DSPCTXT                    127    176
     DSPCTYP                    177    186    INZ('*USRSPC   ')
     DRCVLEN                    197    200B 0 INZ(8)
     DSPCSIZ                    205    208B 0 INZ(2000)
     D*
     D                 DS
     DRECLEN                      1      4B 0 INZ(104)
     DNUMREC                      5      8B 0 INZ(-1)
     DSELSIZ                      9     12B 0 INZ(104)
     DNUMFLD                     13     16B 0 INZ(5)
     DVARLEN                     17     20B 0 INZ(104)
     D*
     D                 DS
     DGETNUM                      1      4B 0 INZ(1)
     DGETSTA                      5      8B 0 INZ(1)
     D*
     DRCVVAR           DS          2000
     D*
     D*
     D*
     C*****************************************************************
     C*
     C* START OF MAINLINE
     C*
     C     *ENTRY        PLIST
     C                   PARM                    OUTPUT            6
     C                   PARM                    SUBSYS           10
     C*
     C     SAVEKEY       KLIST
     C                   KFLD                    SAVEKEY1          7 0
     C                   KFLD                    SAVEKEY2          7 0
     C*
     C                   EXSR      INIT
     C     OUTPUT        CASEQ     '*'           PROC_SCREEN
     C     OUTPUT        CASEQ     '*PRINT'      PROC_PRINT
     C                   ENDCS
     C                   EXSR      DONE
     C*
     C*****************************************************************
     C*
     C* START OF SUBROUTINES
     C*
     C*****************************************************************
     C*
     C*
     C* This subroutine does all the necessary initialization for
     C* the program
     C*
     C*
     C     INIT          BEGSR
     C*
     C                   TIME                    CLOCK             6 0
     C*
     C*                                                     Screen Output
     C     OUTPUT        IFEQ      '*'
     C                   Z-ADD     1             RCDNBR
     C     *IN05         IFEQ      '0'
     C                   OPEN      WRKDASD
     C                   MOVE      'AUXIO'       MODE              5
     C                   ENDIF
     C                   WRITE     HEADING
     C*                                                     Show Job Message
     C                   EVAL      *IN55='1'
     C                   EVAL      *IN91='1'
     C                   WRITE     CONTROL
     C                   EVAL      *IN55='0'
     C                   EVAL      *IN91='0'
```

Figure 170. Source listing for the RPG program WRKDASD (Part 4 of 11)

```
C*                                                 Printer Output
C                   ELSE
C                   EVAL      *IN20='0'
C                   EVAL      *IN21='1'
C                   EXCEPT    PRINTH
C                   ENDIF
C*
C*                                                 Check if User
C*                                                 Space Exists
C                   CALL      'QUSROBJD'
C                   PARM                    RCVVAR
C                   PARM                    RCVLEN
C                   PARM                    RJOBDF
C                   PARM                    SPCNAM
C                   PARM                    SPCTYP
C                   PARM                    QUSBN
C*                                                 Create User Space
C     QUSBNC        IFGT      0
C     QUSBND        IFEQ      'CPF9801'
C                   CALL      'QUSCRTUS'
C                   PARM                    SPCNAM
C                   PARM                    EXTATR
C                   PARM                    SPCSIZ
C                   PARM                    SPCINT
C                   PARM                    SPCAUT
C                   PARM                    SPCTXT
C                   PARM                    SPCREP
C                   PARM                    QUSBN
C                   PARM                    SPCDMN
C*
C     QUSBNC        IFGT      0
C                   MOVEL     'QUSCRTUS'    APINAM
C                   EXSR      APIERR
C                   ENDIF
C                   ELSE
C                   MOVEL     'QUSROBJD'    APINAM
C                   EXSR      APIERR
C                   ENDIF
C                   ENDIF
C*                                                 Get active job
C*                                                 information and put
C*                                                 in user space
C                   CALL      'QGYOLJOB'
C                   PARM                    OLJB0200
C                   PARM                    RECLEN
C                   PARM      'OLJB0200'    FORMAT             8
C                   PARM                    VARDEF
C                   PARM                    VARLEN
C                   PARM                    LSTINF
C                   PARM                    NUMREC
C                   PARM                    SORT
C                   PARM                    OLJS0100
C                   PARM                    SELSIZ
C                   PARM                    NUMFLD
C                   PARM                    KEYS
C                   PARM                    QUSBN
C*
C     QUSBNC        IFGT      0
C                   MOVEL     'QSYLOBJP'    APINAM
C                   EXSR      APIERR
C                   ENDIF
C*                                                 Clear Database File
C*
C                   MOVE      COMMAND(1)    STRING            80
C                   CALLP     QCMDEXC (STRING:%SIZE(STRING))
C*
C*                                                Show DB I/O Message
C     OUTPUT        IFEQ      '*'
C                   EVAL      *IN55='1'
C                   EVAL      *IN92='1'
C                   WRITE     CONTROL
C                   EVAL      *IN55='0'
C                   EVAL      *IN92='0'
C                   ENDIF
```

Figure 171. Source listing for the RPG program WRKDASD (Part 5 of 11)

```
C*
C*                                                     Retrieve each active
C*                                                     job and process
C                   OPEN      DATAFILE
C                   Z-ADD     0             GETSTA
C     GETSTA        DOUEQ     LTOTAL
C                   ADD       1             GETSTA
C*                                                     Get the next record
C*                                                     from the list
C                   CALL      'QGYGTLE'
C                   PARM                    OLJB0200
C                   PARM                    RECLEN
C                   PARM                    LRHAND
C                   PARM                    LSTINF
C                   PARM                    GETNUM
C                   PARM                    GETSTA
C                   PARM                    QUSBN
C*
C     QUSBNC        IFGT      0
C                   MOVEL     'QGYGTLE'     APINAM
C                   EXSR      APIERR
C                   ENDIF
C*
C*                                                     Check which records
C*                                                     to select
C                   MOVEL     QSBSD         QSBS
C                   EVAL      *IN77='0'
C     SUBSYS        CABEQ     '*ALL'        SKIP                     77
C     SUBSYS        CABEQ     QSBS          SKIP                     77
C     SKIP          TAG
C*
C*                                                     Get Database file
C*                                                     names and I/O counts
C*                                                     and update DATA file
C     *IN77         IFEQ      '1'
C                   Z-ADD     QAUXIO        QIO
C                   Z-ADD     QTMPST        QSTOR
C                   EXSR      GETDBIO
C                   Z-ADD     QIO           KEYFLD1
C                   Z-ADD     QDBIO         KEYFLD2
C                   WRITE     DATAF
C                   ENDIF
C                   ENDDO
C*
C                   ENDSR
C*
C*
C*****************************************************************
C*
C* This subroutine gets Database File I/O information
C* Replace the code in this subroutime with new API
C* when it becomes available
C*
C*
C     GETDBIO       BEGSR
C*
C                   Z-ADD     0             QDBIO
C*
C                   CALL      'GETDBIO'
C                   PARM                    QJOB
C                   PARM                    QUSR
C                   PARM                    QJNUM
C*
C                   OPEN      DBIOFILE
C                   READ      DBIOFILE
C                   DOW       %EOF=*OFF
C     DBTYPE        IFEQ      'PHY'
C                   MOVE      DBIO          DBIO#           7 0
C                   ADD       DBIO#         QDBIO
C                   ENDIF
C                   READ      DBIOFILE
C                   ENDDO
C*
C                   CLOSE     DBIOFILE
C                   ENDSR
C*
C*
```

Figure 172. Source listing for the RPG program WRKDASD (Part 6 of 11)

```
C*****************************************************************
C*
C* This subroutine processes Screen output
C*
C*
C     PROC_SCREEN   BEGSR
C*
C                   Z-ADD     9999999       SAVEKEY1
C                   Z-ADD     9999999       SAVEKEY2
C     SAVEKEY       SETLL     DATAFILE
C*                                                  Read next 12 records
C                   EXSR      READ12
C*
C     *INLR         DOUEQ     '1'
C*
C*                                                  Display Screen
C                   EVAL      *IN50='1'
C                   EVAL      *IN55='1'
C                   EXFMT     CONTROL
C                   EVAL      *IN50='0'
C                   EVAL      *IN55='0'
C                   EVAL      *IN88='0'
C                   EVAL      *IN98='0'
C*                                                  Exit(F3 or F12)
C     *IN03         IFEQ      '1'
C     *IN12         OREQ      '1'
C                   EVAL      *INLR='1'
C                   ENDIF
C*                                                  Refresh Datafile
C     *IN05         IFEQ      '1'
C                   EXSR      REFRESH
C                   ENDIF
C*                                                  Sort Datafile
C     *IN06         IFEQ      '1'
C     *IN07         OREQ      '1'
C     *IN08         OREQ      '1'
C                   EXSR      SORTDATA
C                   EVAL      *IN97='0'
C                   ENDIF
C*                                                  Option Selected
C     *INLR         IFEQ      '0'
C     *IN05         ANDEQ     '0'
C     *IN06         ANDEQ     '0'
C     *IN07         ANDEQ     '0'
C     *IN08         ANDEQ     '0'
C                   EXSR      OPTION
C     *IN88         IFEQ      '0'
C                   EVAL      *IN97='0'
C                   ENDIF
C                   ENDIF
C*
C*                                                  Roll Down
C     *IN31         IFEQ      '1'
C                   EVAL      *IN97='0'
C     *IN59         IFEQ      '1'
C                   EVAL      *IN98='1'
C                   ELSE
C* Clear Subfile
C                   EVAL      *IN60='1'
C                   WRITE     CONTROL
C                   EVAL      *IN60='0'
C*                                                  Read next 12 records
C                   EXSR      READ12
C                   ENDIF
C                   ENDIF
C*                                                  Roll Up
C     *IN32         IFEQ      '1'
C     *IN97         ANDEQ     '0'
C*
C                   Z-ADD     KEYFLD1       SAVEKEY1
C                   Z-ADD     KEYFLD2       SAVEKEY2
C*                                                  Read previous 12 recs
C     1             DO        SCREEN#
C                   READP     DATAFILE                          97
C                   ENDDO
```

Figure 173. Source listing for the RPG program WRKDASD (Part 7 of 11)

```
     C     *IN97         IFEQ      '1'
       C     SAVEKEY       SETGT     DATAFILE
       C                   ELSE
       C*                                                    Clear Subfile
       C                   EVAL      *IN60='1'
       C                   WRITE     CONTROL
       C                   EVAL      *IN60='0'
       C     1             DO        12
       C*                                                    Read previous 12 recs
       C                   READP     DATAFILE                               79
       C                   ENDDO
       C     *IN79         IFEQ      '1'
       C                   Z-ADD     9999999       SAVEKEY1
       C                   Z-ADD     9999999       SAVEKEY2
       C     SAVEKEY       SETGT     DATAFILE
       C                   ENDIF
       C*                                                    Read next 12 records
       C                   EXSR      READ12
       C                   ENDIF
       C                   ENDIF
       C*                                                    No option selected
       C     *IN88         IFEQ      '1'
       C     *IN05         ANDEQ     '0'
       C     *IN31         ANDEQ     '0'
       C     *IN32         ANDEQ     '0'
       C                   EVAL      *INLR='1'
       C                   ENDIF
       C*
       C                   ENDDO
       C*
       C                   ENDSR
       C*
       C*
       C*
       C*
       C*****************************************************************
       C*
       C* This subroutine reads the next 12 records from DATAFILE
       C*
       C*
       C     READ12        BEGSR
       C*
       C                   EVAL      *IN59='0'
       C                   Z-ADD     12            SCREEN#           5 0
       C*
       C     1             DO        12            RECNO             5 0
       C                   READ      DATAFILE                               59
       C     *IN59         IFEQ      '0'
       C                   WRITE     SUBFILE
       C                   Z-ADD     RECNO         SCREEN#
       C                   Z-ADD     KEYFLD1       SAVEKEY1
       C                   Z-ADD     KEYFLD2       SAVEKEY2
       C                   ENDIF
       C                   ENDDO
       C*
       C     *IN59         IFEQ      '1'
       C     SAVEKEY       SETLL     DATAFILE
       C                   ENDIF
       C*
       C                   ENDSR
       C*
       C*
       C*****************************************************************
       C*
       C* This subroutine processes all entries until they are complete
       C*
       C*
       C     PROC_PRINT    BEGSR
       C*
       C     9999999       SETLL     DATAFILE
       C*
       C     *IN59         DOUEQ     '1'
       C                   READ      DATAFILE                               59
       C   OF              EXCEPT    PRINTH
       C                   EXCEPT    PRINTL
       C                   ENDDO
```

Figure 174. Source listing for the RPG program WRKDASD (Part 8 of 11)

```
 C*
C                   ENDSR
C*
C*
C*****************************************************************
C*
C     OPTION        BEGSR
C*
C* This subroutine checks if an Option has been selected
C*
C*
C                   READC     SUBFILE                                88
C                   DOW       %EOF=*OFF
C*
C                   MOVE      *BLANKS       STRING
C     'WRKJOB JOB(' CAT       QJNUM         STRING
C     STRING        CAT       '/':0         STRING
C     STRING        CAT       QUSR:0        STRING
C     STRING        CAT       '/':0         STRING
C     STRING        CAT       QJOB:0        STRING
C     STRING        CAT       ') OPTION(':0 STRING
C*
C                   SELECT
C                   WHEN      OPT='5'
C     STRING        CAT       '*OPNF)':0    STRING
C                   WHEN      OPT='6'
C     STRING        CAT       '*SPLF)':0    STRING
C                   WHEN      OPT='7'
C     STRING        CAT       '*JOBLCK)':0  STRING
C                   ENDSL
C*
C                   CALLP     QCMDEXC (STRING:%SIZE(STRING))
C*
C                   MOVE      *BLANK        OPT
C                   UPDATE    SUBFILE
C                   WRITE     HEADING
C*
C                   READC     SUBFILE
C                   ENDDO
C*
C                   ENDSR
C*
C*
C*****************************************************************
C*
C     SORTDATA      BEGSR
C*
C* This subroutine sorts the Datafile in a different order
C*
C*
C*                                                Copy DATFILE to
C*                                                HOLDFILE
C*
C                   CLOSE     DATAFILE
C                   MOVE      COMMAND(2)    STRING
C                   CALLP     QCMDEXC (STRING:%SIZE(STRING))
C*
C*                                                Clear DATAFILE
C*
C                   MOVE      COMMAND(1)    STRING
C                   CALLP     QCMDEXC (STRING:%SIZE(STRING))
C*
C*
C                   OPEN      DATAFILE
C                   OPEN      HOLDFILE
C                   READ      HOLDFILE                               44
C     *IN44         DOUEQ     '1'
C*
C                   SELECT
```

Figure 175. Source listing for the RPG program WRKDASD (Part 9 of 11)

```
C                   WHEN      *IN06='1'
C                   Z-ADD     QDBIO         KEYFLD1
C                   Z-ADD     QIO           KEYFLD2
C                   MOVE      'DBIO '       MODE
C                   WHEN      *IN07='1'
C                   Z-ADD     QSTOR         KEYFLD1
C                   Z-ADD     QIO           KEYFLD2
C                   MOVE      'TSTOR'       MODE
C                   WHEN      *IN08='1'
C                   Z-ADD     QIO           KEYFLD1
C                   Z-ADD     QDBIO         KEYFLD2
C                   MOVE      'AUXIO'       MODE
C                   ENDSL
C*
C                   WRITE     DATAF
C                   READ      HOLDFILE                               44
C                   ENDDO
C*
C                   CLOSE     HOLDFILE
C*                                                  Clear Subfile
C                   EVAL      *IN60='1'
C                   WRITE     CONTROL
C                   EVAL      *IN60='0'
C*                                                  Read first 12 records
C*
C                   Z-ADD     9999999       SAVEKEY1
C                   Z-ADD     9999999       SAVEKEY2
C     SAVEKEY       SETLL     DATAFILE
C                   EXSR      READ12
C*
C                   ENDSR
C*
C*
C*****************************************************************
C*
C     REFRESH       BEGSR
C*
C* This subroutine clears current information and re-gathers it
C*
C*
C                   CLOSE     DATAFILE
C                   EXSR      INIT
C*
C                   SELECT
C                   WHEN      MODE='DBIO '
C                   EVAL      *IN06='1'
C                   WHEN      MODE='TSTOR'
C                   EVAL      *IN07='1'
C                   WHEN      MODE='AUXIO'
C                   EVAL      *IN08='1'
C                   ENDSL
C*
C                   ENDSR
C*
C*
C*****************************************************************
C*
C     DONE          BEGSR
C*
C* This subroutine performs end of job processing
C*
C*
C     OUTPUT        IFEQ      '*PRINT'
C                   EXCEPT    ENDLST
C                   ENDIF
C                   EVAL      *INLR='1'
C*
C                   ENDSR
C*
C*
C*****************************************************************
C*
C     APIERR        BEGSR
C*
C* This subroutine prints error messages
```

Figure 176. Source listing for the RPG program WRKDASD (Part 10 of 11)

```
 74000      C*
 74100      C*
 74200      C     APINAM        DSPLY
 74300      C     QUSBND        DSPLY
 74400      C                   EXSR      DONE
 74500      C*
 74600      C                   ENDSR
 74700      C*
 74800      OQSYSPRT   E            PRINTH         1 02
 74900      O                                         58 'Work with DASD'
 75000      O                                        109 'Date'
 75100      O                       UDATE         Y    118
 75200      O                                        123 'Time'
 75300      O                       CLOCK              132 '  :  :  '
 75400      O          E            PRINTH         2
 75500      O                                        123 'Page'
 75600      O                       PAGE          Z    132
 75700      O          E            PRINTH         1
 75800      O                                         69 'Aux I/O'
 75900      O                                         83 'DB I/O'
 76000      O                                        101 'Temporary'
 76100      O          E            PRINTH         2
 76200      O                                          8 'Job Name'
 76300      O                                         21 'User Name'
 76400      O                                         33 'Job Number'
 76500      O                                         42 'Program'
 76600      O                                         57 'Subsystem'
 76700      O                                         70 'Requests'
 76800      O                                         85 'Requests'
 76900      O                                        100 'Storage'
 77000      O          E            PRINTL         1
 77100      O                       QJOB                10
 77200      O                       QUSR                22
 77300      O                       QJNUM               31
 77400      O                       QPROG               45
 77500      O                       QSBS                58
 77600      O                       QIO           J     70
 77700      O                       QDBIO         J     85
 77800      O                       QSTOR         J    101
 77900      O          E            ENDLST      2  2
 78000      O                                        132 '*** End of Listing ***'
 78100      O*
 78200      O*
 78300 **CTDATA COMMAND
 78400 CLRPFM FILE(WRKDASD/DATAFILE)
 78500 CPYF FROMFILE(WRKDASD/DATAFILE) TOFILE(WRKDASD/HOLDFILE) MBROPT(*REPLACE)
                               * * * *  E N D  O F  S O U R C E  * * * *
```

Figure 177. Source listing for the RPG program WRKDASD (Part 11 of 11)

```
SOURCE FILE . . . . . . .  WRKDASD/QCMDSRC
  MEMBER  . . . . . . . . .  WRKDASD

  SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
    100             CMD       PROMPT('Work with DASD')
    200             PARM      KWD(OUTPUT) TYPE(*CHAR) LEN(6) RSTD(*YES) +
    300                         DFT(*) VALUES(* *PRINT) MIN(0) +
    400                         PROMPT('Output' 1)
    500             PARM      KWD(SUBSYS) TYPE(*NAME) DFT(*ALL) +
    600                         SPCVAL((*ALL)) MIN(0) PROMPT('Subsystem +
    700                         Name')
                                * * * *  E N D  O F  S O U R C E  * * * *
```

Figure 178. Source listing for the WRKDASD command

Appendix E. Using the additional material

This redbook also contains additional material that is available for downloading from the Web. See the appropriate section below for instructions on using or downloading each type of material.

E.1 Locating the additional material on the Internet

The Web material associated with this redbook is also available in softcopy on the Internet from the IBM Redbooks Web server. The code is available in a save file that can be downloaded from the redbooks site. Point your Web browser to:

`ftp://www.redbooks.ibm.com/redbooks/SG24-5693`

Alternatively, you can go to the IBM Redbooks Web site at:

`http://www.redbooks.ibm.com/`

Select **Additional materials,** and open the directory that corresponds with the redbook form number.

E.2 Using the Web material

The materials included in the additional materials section of the site are provided “as is”. That means, there is no support available for them.

Go to the ITSO Web site at: `http://www.redbooks.ibm.com`

Select the **Additional Materials** tab. Scroll down the page until you find the **SG245693** directory. Review the readme.txt file for information about how to get the additional materials to your system.

The additional Web material that accompanies this redbook includes:

File name	*Description*
SG245693.zip	Zipped code samples
5693readme.txt	Text document describing how to obtain and load code to customer’s AS/400 system

© Copyright IBM Corp. 2000

Appendix F. Special notices

This publication is intended to help AS/400 system administrators and operators understand and manage the disk storage subsystem of their AS/400 system. The information in this publication is not intended as the specification of any programming interface provided by OS/400 operating system. See the PUBLICATIONS section of the IBM Programming Announcement for OS/400 operating system and AS/400 Performance Tools products for more information about what publications are considered to be product documentation.

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

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

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

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

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

© Copyright IBM Corp. 2000

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

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

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

AFP	AIX
AS/400	AT
CT	DB2
IBM	ImagePlus
Information Warehouse	Netfinity
NetView	OfficeVision
OfficeVision/400	OS/400
RS/6000	S/390
SP	StorWatch
System/36	System/38
System/390	Versatile Storage Server
XT	3090
3890	400

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere.,The Power To Manage., Anything. Anywhere.,TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET and the SET logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product, and service names may be trademarks or service marks of others.

Appendix G. Related publications

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

G.1 IBM Redbooks

For information on ordering these ITSO publications, see “How to get IBM Redbooks” on page 227.

- *AS/400e System Handbook,* GA19-5486
- *AS/400 System Builder,* SG24-2155
- *The System Administrator’s Companion to AS/400 Availability and Recovery*, SG24-2161
- *IBM Versatile Storage Server*, SG24-2221
- *DB2/400 Advanced Database Functions*, SG24-4249
- *Using Tivoli Storage Management to Back Up Databases*, SG24-4335
- *Complementing AS/400 Storage Management Using Hierarchical Storage Management APIs,* SG24-4450
- *Using ADSM to Back Up Lotus Domino*, SG24-4534
- *Using ADSM Hierarchical Storage Management*, SG24-4631
- *A Practical Approach to Managing Backup Recovery and Media Services for OS/400*, SG24-4840
- *DB2/400: Mastering Data Warehousing Functions*, SG24-5184
- *AS/400 Consolidation Strategies and Implementation*, SG24-5186
- *AS/400 Remote Journal Function for High Availability and Data Replication,* SG24-5189
- *AS/400 Client Access Express for Windows: Implementing V4R4M0*, SG24-5191
- *The AS/400 NetServer Advantage*, SG24-5196
- *SAP Business Information Warehouse on the AS/400 System,* SG24-5200
- *Management Central: A Smart Way to Manage AS/400 Systems*, SG24-5407
- *Slicing the AS/400 with Logical Partitioning: A How to Guide*, SG24-5439
- *Introduction to Storage Area Network, SAN,* SG24-5470

The following publications are available in soft copy only from the redbooks Web site: `http://www.redbooks.ibm.com/`

- *Implementing IBM SAA ImagePlus WAF/400 File Cabinet and Optical Libraries*, GG24-4188
- *IBM SAA ImagePlus Workfolder Application Facility/400 and Remote 3995 Library*, GG24-4413
- *AS/400 Performance Explorer Tips and Techniques*, SG24-4781
- *Data Warehousing Solutions on the AS/400*, SG24-4872

© Copyright IBM Corp. 2000

G.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at http://www.redbooks.ibm.com/ for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbooks Collection	SK2T-8038
Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
AS/400 Redbooks Collection	SK2T-2849
Netfinity Hardware and Software Redbooks Collection	SK2T-8046
RS/6000 Redbooks Collection (BkMgr Format)	SK2T-8040
RS/6000 Redbooks Collection (PDF Format)	SK2T-8043
Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

G.3 Other resources

These publications are also relevant as further information sources:

- *Performance Tools/400*, SC41-4340
- *AS/400 Job Scheduler,* SC41-5300
- *AS/400 Backup and Recovery,* SC41-5304
- *AS/400 Work Management Guide*, SC41-5306
- *Performance Tools for AS/400*, SC41-5340
- *Backup Recovery and Media Services/400,* SC41-5345
- *Hierarchical Storage Management Use,* SC41-5351
- *APPC Programming*, SC41-5443
- *AS/400 CL Reference*, *V4R4,* SC41-5722
- *System API Reference*, SC41-5801
- *940x 170, 6xx, 7xx, and Sxx System Installation and Upgrade*, SY44-5950
- *Problem Analysis Guide,* SY44-5955

The following publications are available in soft copy only by visiting the Web site at: http://publib.boulder.ibm.com/pubs/html/as400/online/homeeng1.htm

- *AS/400 Performance Capabilities Reference V4R4,* SC41-0607
- *AS/400 System Operation,* SC41-4203
- *AS/400 Database Programming,* SC41-5701
- *Integrated File System Introduction,* SC41-5711

G.4 Referenced Web sites

These Web sites are also relevant as further information sources:

- AS/400 Beyond Technology Page: `http://www.as400.ibm.com/beyondtech`
- AS/400 Information Center: `http://www.as400.ibm.com/infocenter`
- AS/400 Technical Studio: `http://www.as400.ibm.com/techstudio`
- Hierarchical Storage Management information: `http://www.as400.ibm.com/beyondtech/hsm.htm`
- IBM Redbooks home page: `http://www.redbooks.ibm.com`
- AS/400 Information Center: `http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/INFOCENT.HTM`
- LPAR information: `http://www.as400.ibm.com/lpar/`
- Information on clustering:
  - `http://www.as400.ibm.com/ha/sec2.htm`
  - `http://www.as400.ibm.com/ha/sec23.htm`
  - `http://www.as400.ibm.com/ha/sec24.htm`
- IBM storage home page: `http://www.storage.ibm.com`
- Information on Storage Area Network (SAN): `http://www.storage.ibm.com/ibmsan/index.htm`
- The *Storage Area Networks: Putting Data to Work for e-businesses* white paper: `http://www.storage.ibm.com/ibmsan/whitepaper/strategy.htm`
- The *Disk Arm Requirements* and *Version Performance Considerations* white papers: `http://www.ibm.com.as400/developer/performance`
- Site Preparation and Physical Planning at the AS/400 Technical Studio site at: `http://www.as400.ibm.com/tstudio/planning/plngstrt.htm`
- Additional information on optical library solutions: `http://www.storage.ibm.com/hardsoft/opticalstor/index.html`
- Magnetic Media Controllers information: `http://www.as400.ibm.com/handbook/5486MagC.html`
- Extended Adaptive Cache information: `http://www.as400.ibm.com/beyondtech/ext_adapt_cache.htm`
- PEX information: `http://www.as400service.ibm.com/p_dir/pexgui.nsf`
- AS/400 Online Library: `http://publib.boulder.ibm.com/pubs/html/as400/online/homeeng1.htm`

How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** http://www.redbooks.ibm.com/

  Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

  Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders by e-mail including information from the IBM Redbooks fax order form to:

	e-mail address
In United States	usib6fpl@ibmmail.com
Outside North America	Contact information is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	Country coordinator phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Fax Orders**

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	Fax phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and Redbooks by accessing the IBM Intranet Web site at http://w3.itso.ibm.com/ and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at http://w3.ibm.com/ for redbook, residency, and workshop announcements.

© Copyright IBM Corp. 2000

IBM Redbooks fax order form

Please send me the following:

Title	Order Number	Quantity

First name | Last name

Company

Address

City | Postal code | Country

Telephone number | Telefax number | VAT number

☐ Invoice to customer number

☐ Credit card number

Credit card expiration date | Card issued to | Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Index

© Copyright IBM Corp. 2000

U

V

W

IBM Redbooks review

Your feedback is valued by the Redbook authors. In particular we are interested in situations where a Redbook "made the difference" in a task or problem you encountered. Using one of the following methods, **please review the Redbook, addressing value, subject matter, structure, depth and quality as appropriate.**

- Use the online **Contact us** review redbook form found at http://www.redbooks.ibm.com/
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Document Number **Redbook Title**	SG24-5693-00 AS/400 Disk Storage Topics and Tools Topics and Tools
Review	
What other subjects would you like to see IBM Redbooks address?	
Please rate your overall satisfaction:	O Very Good O Good O Average O Poor
Please identify yourself as belonging to one of the following groups:	O Customer O Business Partner O Solution Developer O IBM, Lotus or Tivoli Employee O None of the above
Your email address: The data you provide here may be used to provide you with information from IBM or our business partners about our products, services or activities.	O Please do not use the information collected here for future marketing or promotional contacts or other communications beyond the scope of this transaction.
Questions about IBM's privacy policy?	The following link explains how we protect your personal information. http://www.ibm.com/privacy/yourprivacy/

© Copyright IBM Corp. 2000

SG24-5693-00

Printed in the U.S.A.

AS/400 Disk Storage Topics and Tools Topics and Tools

SG24-5693-00