

AS/400



# DSNX Support

*Version 4*



AS/400



# DSNX Support

*Version 4*

**Take Note!**

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

**First Edition (August 1997)**

This edition applies to the licensed program IBM Operating System/400 (Program 5769-SS1), Version 4 Release 1 Modification 0, and to all subsequent releases and modifications until otherwise indicated in new editions.

Make sure that you are using the proper edition for the level of the product.

Order publications through your IBM representative or the IBM branch serving your locality. If you live in the United States, Puerto Rico, or Guam, you can order publications through the IBM Software Manufacturing Solutions at 800+879-2755. Publications are not stocked at the address given below.

IBM welcomes your comments. A form for readers' comments may be provided at the back of this publication. You can also mail your comments to the following address:

IBM Corporation  
Attention Department 542  
IDCLERK  
3605 Highway 52 N  
Rochester, MN 55901-7829 USA

or you can fax your comments to:

United States and Canada: 800+937-3430  
Other countries: (+1)+507+253-5192

If you have access to Internet, you can send your comments electronically to [IDCLERK@RCHVMW2.VNET.IBM.COM](mailto:IDCLERK@RCHVMW2.VNET.IBM.COM); [IBMMAIL](mailto:IBMMAIL), to [IBMMAIL\(USIB56RZ\)](mailto:IBMMAIL(USIB56RZ)).

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

© **Copyright International Business Machines Corporation 1997. 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

<b>Notices</b> . . . . .	v	Releasing a Phase . . . . .	4-2
Programming Interface Information . . . . .	vi	DSNX Security Considerations . . . . .	4-4
Trademarks . . . . .	vi	Coexistence Considerations . . . . .	4-4
<b>About DSNX Support, SC41-5409</b> . . . . .	vii	Sending the Same Resource to Several Nodes . . . . .	4-5
Who Should Use This Book . . . . .	vii	DSNX Processing Considerations . . . . .	4-5
Prerequisite and Related Information . . . . .	viii	AS/400 Processing of CLISTs Sent by NetView DM . . . . .	4-5
Information Available on the World Wide Web . . . . .	viii	Replacing Objects on an AS/400 System . . . . .	4-7
<b>Chapter 1. Introduction DSNX Support</b> . . . . .	1-1	Retrieving Objects Other Than Members or Save Files . . . . .	4-7
<b>Chapter 2. Host System Programming Considerations for DSNX</b> . . . . .	2-1	NetView DM Session and Node Considerations . . . . .	4-8
Using the AS/400 System in a NetView DM Environment . . . . .	2-1	NetView DM Sessions . . . . .	4-8
Systems in a NetView DM Network . . . . .	2-1	Direct Node . . . . .	4-8
AS/400 and NetView DM Terms . . . . .	2-2	Intermediate Node . . . . .	4-9
NetView DM Functions Supported by OS/400 DSNX . . . . .	2-2	Data Considerations Using NetView DM . . . . .	4-9
DSNX File and Member Considerations . . . . .	2-3	Considerations Using DSNX on the AS/400 System . . . . .	4-10
VTAM/NCP Programming Considerations . . . . .	2-4	<b>Chapter 5. Sample Procedures for DSNX</b> . . . . .	5-1
Physical Unit Definition Parameters . . . . .	2-4	Distributing Programs to Systems that Use a Previous Release . . . . .	5-1
Logical Unit Definition Parameters . . . . .	2-5	Distributing an AS/400 Application Library . . . . .	5-2
NCP Generation Macro Considerations . . . . .	2-5	Retrieving and Printing a History File . . . . .	5-4
Host System Work Sheet . . . . .	2-6	Transferring Program Temporary Fixes (PTFs) . . . . .	5-5
<b>Chapter 3. Before Running DSNX</b> . . . . .	3-1	Transferring Spooled File Entries . . . . .	5-5
OS/400 DSNX Support . . . . .	3-1	<b>Chapter 6. DSNX Differences</b> . . . . .	6-1
DSNX Host Interface Subsystem . . . . .	3-2	Differences from System/36 DSNX Support . . . . .	6-1
DSNX Request Processor Subsystem . . . . .	3-4	<b>Appendix A. NetView DM to OS/400 DSNX Configuration Example</b> . . . . .	A-1
DSNX/&pcs. Subsystem . . . . .	3-4	System/370 Host Considerations . . . . .	A-2
NetView DM Host to DSNX Host Interface Configuration . . . . .	3-5	Definitions for HANODE Node . . . . .	A-3
Host System Logon Modes . . . . .	3-7	Definitions for DALB60 Node . . . . .	A-5
DSNX Logon Modes (via NetView DM) . . . . .	3-7	TU1021 Definitions . . . . .	A-6
DSNX Start-Up Considerations . . . . .	3-7	<b>Appendix B. DSNX Problem Analysis</b> . . . . .	B-1
Defining Your AS/400 System to NetView DM . . . . .	3-7	DSNX Logging at Local Request Processor . . . . .	B-2
DSNX Host Interface-to-DSNX Request Processor Configuration . . . . .	3-8	DSNX Logging at Remote Request Processor . . . . .	B-3
DSNX Host Interface-to-DSNX/&pcs. Configuration . . . . .	3-10	DSNX Logging with DSNX/Client Access . . . . .	B-4
DSNX/&pcs.-to-PC Configuration . . . . .	3-12	Area 1: DSNX Host Interface . . . . .	B-5
Defining Your PC Nodes to NetView DM . . . . .	3-13	Area 2: DSNX Request Processor . . . . .	B-5
Work with DSNX/PC Distribution Queues (WRKDPCQ) . . . . .	3-13	Area 3: DSNX/Client Access . . . . .	B-6
<b>Chapter 4. DSNX Considerations</b> . . . . .	4-1	Area 4: Local DSNX/PC Queue Management . . . . .	B-6
DSNX Naming Conventions . . . . .	4-1	DSNX Journal Analysis . . . . .	B-6
Object Naming . . . . .	4-1		
DSNX Scheduling . . . . .	4-2		

DSNX Journal Formats from DSPJRN	
Command . . . . .	B-9
Deleting Entries from the QDSNX Journal	B-13
<b>Appendix C. DSNX Request Descriptions</b>	C-1
<b>Appendix D. NetView DM to DSNX Data Flow</b>	D-1
Intermediate Node Successful Add Function	D-2
Journal Entries for Intermediate Node—Add Data . . . . .	D-4
Intermediate Node Successful Retrieve with Compress Function . . . . .	D-4
Journal Entries for Intermediate Node—Retrieve with Compress . . . . .	D-6
Intermediate Node Unsuccessful Delete Function . . . . .	D-6
Journal Entries for Intermediate Node—Unsuccessful Delete Function . . . . .	D-7
Intermediate Node—Unsuccessful Replace Function . . . . .	D-7
Journal Entries for Intermediate Node—Unsuccessful Replace . . . . .	D-8
Direct Node Successful Replace Function . . . . .	D-8
Journal Entries for Direct Node—Replace . . . . .	D-9
Direct Node Successful Retrieve with Compress Function . . . . .	D-9
Journal Entries for Direct Node—Retrieve with Compress . . . . .	D-10

Direct Node Successful Replace with Decompress Function . . . . .	D-11
Journal Entries for Direct Node—Replace with Decompress . . . . .	D-12
Direct Node Successful Retrieve CLIST Function . . . . .	D-12
Journal Entries for Direct Node—Retrieve CLIST . . . . .	D-13
Direct Node Unsuccessful Initiate Job Function . . . . .	D-13
Journal Entries for Direct Node—Initiate Job . . . . .	D-14
<b>Bibliography</b> . . . . .	H-1
IBM Publications . . . . .	H-1
Communications and Programming . . . . .	H-1
NetView . . . . .	H-1
NetView Distribution Manager . . . . .	H-2
Advanced Communications Function for Virtual Telecommunications Access Method (ACF/VTAM) . . . . .	H-2
Systems Network Architecture (SNA) . . . . .	H-2
Data Link Control . . . . .	H-2
Communications Controllers . . . . .	H-2
Personal Computer . . . . .	H-2
System/38 . . . . .	H-2
<b>Index</b> . . . . .	X-1

---

## Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

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

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 the software interoperability coordinator. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Address your questions to:

IBM Corporation  
Software Interoperability Coordinator  
3605 Highway 52 N  
Rochester, MN 55901-7829 USA

This publication could contain technical inaccuracies or typographical errors.

This publication may refer to products that are announced but not currently available in your country. This publication may also refer to products that have not been announced in your country. IBM makes no commitment to make available any unannounced products referred to herein. The final decision to announce any product is based on IBM's business and technical judgment.

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

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

---

## Programming Interface Information

This book is intended to help the customer to use DSNX support. This book documents General-Use Programming Interface and Associated Guidance Information provided by DSNX support.

General-Use programming interfaces allow the customer to write programs that obtain the services of DSNX support.

---

## Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM	OS/2
AnyNet	OS/400
Application System/400	SystemView
AS/400	System/370
IBM	System/390
NetView	VTAM
Operating System/400	400
Operational Assistant	

Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

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

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

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Inc.

Other company, product, and service names, which may be denoted by a double asterisk (\*\*), may be trademarks or service marks of others.



---

## About DSNX Support, SC41-5409

This book is intended for the programmer who is responsible for configuring the AS/400 system to use the communications and systems management functions. For the AS/400 system, this support includes the following: change management functions in an IBM NetView Distribution Manager NetView DM network, and problem management functions in a network.

This book should also be useful to the host system programmer who is responsible for adding the AS/400 system into the host system network.

For a list of related publications, see the Bibliography.

---

### Who Should Use This Book

Using this book, the AS/400 programmer can:

- Configure the AS/400 system to use the distributed systems node executive (DSNX) support.
- Coordinate change management and distribution activities with the host system NetView DM programmer.
- Perform central site problem analysis for the AS/400 systems in a network.

Using this book, the host system programmer can:

- Generate the Virtual Telecommunications Access Method/Network Control Program (VTAM/NCP) host system to include the AS/400 system as a node in a NetView DM communications network.
- Coordinate change activities with the AS/400 programmer in the NetView DM network.

You should be familiar with the following to use the information in this book:

- AS/400 programming terminology. You should also be familiar with the terminology of the host system.
- Data communications concepts.
- Configuration and communications information that is provided in the books: *Communications Management*, SC41-5406, and *Communications Configuration*, SC41-5401.

---

## Prerequisite and Related Information

For information about other AS/400 publications (except Advanced 36), see either of the following:

- The *Publications Reference* book, SC41-5003, in the AS/400 Softcopy Library.
- The *AS/400 Information Directory*, a unique, multimedia interface to a searchable database that contains descriptions of titles available from IBM or from selected other publishers. The *AS/400 Information Directory* is shipped with the OS/400 operating system at no charge.

---

## Information Available on the World Wide Web

More AS/400 information is available on the World Wide Web. You can access this information from the AS/400 home page, which is at the following uniform resource locator (URL) address:

<http://www.as400.ibm.com>

Select the Information Desk, and you will be able to access a variety of AS/400 information topics from that page.

---

## Chapter 1. Introduction DSNX Support

**IBM NetView Distribution Manager (NetView DM)** is a licensed program that gives you the capability to plan, schedule, and control the exchange of data between the **host system** (the primary or controlling computer) and one or more remote sites. **OS/400 distributed systems node executive (DSNX)** support allows the AS/400 system to be part of the NetView DM network. DSNX support is a function of the operating system that receives and analyzes requests from NetView DM on the host system.

You can use DSNX support to distribute files and input streams in a network controlled by a System/370\* or System/390\* host system. You can use this function for central site programming and maintenance and also for the distribution of AS/400 objects.

**Note:** When you install a new release of the OS/400 operating system, all outstanding NetView DM requests on the AS/400 system are lost. If requests are deleted on the AS/400 system, the plan/phase on the NetView DM host remains in PENDING status. You should ensure that no NetView DM requests are outstanding on the NetView DM host for the system being installed (or any secondary nodes attached to the system) before installing a new release of the OS/400 operating system.

DSNX support allows the AS/400 system, System/36, and the personal computer to become part of a NetView DM network. The AS/400 system functions as a *direct node* to the host system or as an *intermediate node* between the host system and other AS/400 systems, personal computers, and System/36s.

DSNX support enables one or more AS/400 systems or System/36s (and the host system) to distribute, through a NetView DM host system, objects among other AS/400 systems, System/36s, and personal computers in the network. You can also use DSNX to process distribution lists received from NetView DM and forward the requests to other AS/400 systems, System/36s, or personal computers.

When DSNX support is active on the AS/400 system, very little operator action is required. The NetView DM host system controls all transfers of information between the nodes and the NetView DM host system.

Depending on which configuration of DSNX you choose to use, the following communications support is required:

- OS/400 DSNX running as an application program requires **Systems Network Architecture upline facility (SNUF)** for communications with the NetView DM host system. SNUF is the communications support that allows the AS/400 system to communicate with CICS/VS and IMS/VS application programs on a host system. For more information on SNUF, see the *SNA Upline Facility Programming* book.
- OS/400 DSNX acting as an intermediate node requires an **object distribution/Systems Network Architecture distribution services (SNADS)** connection to other AS/400 systems and System/36s. Object distribution is a function that allows a user to send source and data files, save files, input streams, spooled output files, and messages to another user, in this case, on a SNADS network. SNADS is an IBM\* asynchronous distribution service that

defines a set of rules to receive, route, and send electronic mail in a network of systems. For more information on SNADS, see the *SNA Distribution Services* book.

- OS/400 DSNX-sender distributing NetView DM requests to personal computers attached to an AS/400 system requires **advanced program-to-program communications (APPC)**, which is the communications support that allows programs on an AS/400 system to communicate with programs on other systems that have compatible communications support. For more information on APPC, see the *APPC Programming* book.

---

## Chapter 2. Host System Programming Considerations for DSNX

The host system programmer should read the following summary of programming considerations. This information should not be needed by the AS/400 programmer. When the support is configured, the AS/400 programmer must be provided with certain parameter values that were specified during host system generation.

Additional information about the NetView DM and VTAM programs can be obtained from the books listed in the "Bibliography" on page H-1.

---

### Using the AS/400 System in a NetView DM Environment

A host system, using the IBM NetView Distribution Manager (NetView DM) licensed product, can perform change management activities on the AS/400 system in a NetView DM network. The host system must be using NetView DM or Distributed Systems Executive (DSX) Version 3.2, a licensed program running under the Virtual Storage Extended (VSE) operating system.

NetView DM allows the host system to send requests to an AS/400 system that will:

- Retrieve, send, or delete database file members and save files and other objects.
- Retrieve, send, delete, or start batch job streams.
- Send messages to the system operator message queue.

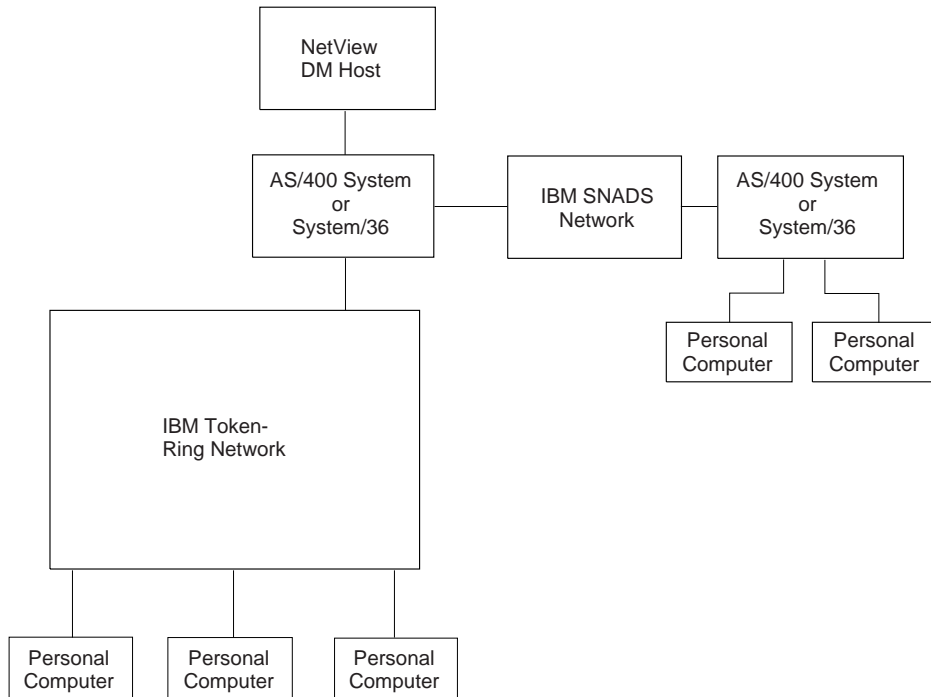
NetView DM allows the host system to retrieve, send, and delete files, programs, formats, and procedures in a network of computers. NetView DM uses plans to define the work to be done. A plan may consist of one or more phases. OS/400 DSNX can start a transfer by requesting that a held phase be released from a NetView DM host. OS/400 DSNX can also respond to requests made by the NetView DM host system.

### Systems in a NetView DM Network

A NetView DM network consists of the host system (an IBM System/370 system, an IBM System/390 system, or a 30xx or 43xx processor) and can contain one or more of the following:

- An AS/400 system defined as node type SSP
- A System/36 defined as node type SSP
- Personal computers defined as node type PDOS

Each of these is called a **node** (one of the systems or devices in a network). Phase information includes the name of the node being communicated with, the schedule used for communications, and the list of requests NetView DM should make to the node during the session. Each phase in a NetView DM plan may require multiple LU-to-LU sessions (one at a time) between NetView DM and DSNX. Figure 2-1 on page 2-2 illustrates a NetView DM network. You can include all of the configurations or only those that are appropriate to your particular application. The host system to an AS/400 system configuration is discussed first as the basic DSNX configuration, and the other configurations are discussed later.



RSL001-2

Figure 2-1. A DSNX Network

## AS/400 and NetView DM Terms

The AS/400 system and NetView DM frequently use different terms to refer to the same thing. The following table lists some NetView DM terms and shows the corresponding AS/400 term.

NetView DM Term	AS/400 Equivalent Term
CLIST	Batch job stream
Panel	Display file
Data set	Physical database file member
Program	Program, other objects

## NetView DM Functions Supported by OS/400 DSNX

With OS/400 DSNX support on one or more AS/400 systems, the NetView DM host system can provide:

- File member support
  - Retrieve database file members from an AS/400 system.
  - Send sequential files that were prepared at the host system to an AS/400 system.
  - Send database file members that were created at an AS/400 system to another AS/400 system.
  - Delete database file members from an AS/400 system.
  - Compress and decompress file members for send and retrieve operations.

- Synchronize data, to the last successfully received record, after an error for send and retrieve operations.  
See “Data Considerations Using NetView DM” on page 4-9 for information on preparing data for NetView DM storage if the data is to be sent to file members.
- Save file support
  - Retrieve save files from an AS/400 system.
  - Delete save files from an AS/400 system.
  - Send save files to an AS/400 system.
  - Synchronize data, to the last successfully received record, after an error for send and retrieve operations.
- All other object support
  - Retrieve AS/400 objects. A valid object type is any object type that can be specified on the OBJTYPE parameter of the Save Object (SAVOBJ) command.
  - Send AS/400 objects. A valid object type is any object type that can be specified on the OBJTYPE parameter of the Save Object (SAVOBJ) command.
  - Delete objects from an AS/400 system.
  - Synchronize data, to the last successfully received record, after an error for send and retrieve operations.  
See “Data Considerations Using NetView DM” on page 4-9 and “NetView DM Session and Node Considerations” on page 4-8 for more information about sending and retrieving objects.
- Batch job support
  - Retrieve input streams from an AS/400 database member.
  - Run batch job streams that were stored at the host system. The batch job streams could be created on the host system or an AS/400 system.
  - Delete input streams that are in an AS/400 database member.
  - Send batch job streams to an AS/400 database member.
- Message support
  - Send messages to an AS/400 system operator using the QSYSOPR message queue.

## DSNX File and Member Considerations

OS/400 DSNX support accepts *sequential* files, *source* members, and *batch job stream* members that were prepared at the NetView DM host system. These objects may contain double-byte characters. These types of objects are prepared for the network by the NetView DM host system when the PREPARE DATASET or the PREPARE CLIST control statement is used.

OS/400 DSNX support also accepts database file members created on another AS/400 system.

If a database file member is added or replaced onto another AS/400 system, the physical file must exist on the receiving AS/400 system before the add or replace request is processed. The logical record length of the existing file on the AS/400 system must match the logical record length specified in the NetView DM request.

Information about how files sent by NetView DM are replaced on an AS/400 system is provided under "Replacing Objects on an AS/400 System" on page 4-7.

DSNX cannot be used to upgrade an AS/400 system to a new release.

---

## VTAM/NCP Programming Considerations

Before communications can occur between the NetView DM host system and OS/400 DSNX, VTAM/NCP generation must be done on the host system. All AS/400 systems to be included in the network must be defined during the VTAM/NCP generation. Because each AS/400 line is represented as a physical unit to VTAM/NCP, each AS/400 line that the DSNX support uses requires a physical unit definition in the generation. And because the DSNX session is considered an SNA logical unit, the session requires a logical unit definition.

A description of all the parameters affecting DSNX definition types follows.

## Physical Unit Definition Parameters

The following parameters on the physical unit (PU macroinstruction) definition apply to the DSNX support:

**ADDR = xx**

Specifies the **station address**, a 2-character hexadecimal value from 01 to FE.

**DISCNT = NO/YES**

Specifies whether VTAM/NCP is to disconnect the physical unit when the last logical unit session is ended. DISCNT=NO allows the AS/400 system to remain connected when no sessions are active; the physical unit is deactivated when the last device on the line is varied off. DISCNT=YES disconnects the AS/400 system when the last session ends; the DSNX support remains active until the device is varied off. DISCNT=YES also causes the VTAM program to ignore the AS/400 vary off request. If switched lines and multiple locations are configured, specify DISCNT=YES.

**IDBLK = 056**

IDBLK must be specified as 056 for an AS/400 system.

**IDNUM = *number***

The IDBLK and IDNUM parameters make up the SDLC exchange identifier. These parameters are specified only for a switched line.

**ISTATUS = ACTIVE/INACTIVE**

Specifies whether the physical unit should be activated when its major node is activated.

**MAXDATA = 2057**

Specifies the maximum amount of data, including the transmission header and request/response header, that the physical unit can receive. The AS/400 system accepts a maximum of 2057 bytes.



**MAXOUT = 7**

Specifies the number of Path Information Units (PIUs) that VTAM/NCP will send to the AS/400 system before requesting a response. For best performance, 7 should be specified.

**PUTYPE = 2**

The physical unit type must be 2.

**SSCPFM = USSCS**

Specifies that the AS/400 logical units associated with this physical unit use character-coded messages for communications with VTAM/NCP. The AS/400 system requires character-coded messages.

**USSTAB = *name***

Specifies the name of a USS definition table. For DSNX, USSTAB must support the PL1 format of the LOGON command.

## Logical Unit Definition Parameters

The following parameters on the logical unit (LU macroinstruction) definition apply to the DSNX support:

**ENCR = NONE**

Specifies the type of encryption to be used. Encryption is not supported by the AS/400 system for the DSNX support, so NONE must be specified.

**LOCADDR = *address***

Specifies the local address of the session. The **local address** is equivalent to a logical unit number and matches the LOCADR parameter on the device description of the AS/400 system.

**ISTATUS = ACTIVE/INACTIVE**

Specifies whether the logical unit is to be activated when the physical unit is activated.

**PACING = *count***

Specifies the way pacing is to be handled between VTAM/NCP and the logical unit. **Pacing** controls the rate of data flow between the OS/400 program and the host system. Pacing allows the receiver to control the rate at which the sender sends requests.

Each OS/400 logical unit has both a sending and a receiving pacing value. The AS/400 system supports all valid values for sending and receiving pacing from 0 to 63. A value of 0 indicates pacing will not be enforced.

**DLOGMOD = *name***

Specifies the logon mode table entry to be used in the bind command to this LU.

---

## NCP Generation Macro Considerations

Caution should be used when VTAM/NCP is generated for use with NetView DM/DSNX. The value for the MAXDATA parameter on the VTAM NCP generation macroinstruction must be large enough to accommodate the request/response unit (RU) size defined for the NetView DM/DSNX session. If the MAXDATA parameter value is not large enough to contain both an RU and SNA header information, then

the VTAM program does not send all the data sent to it by NetView DM. If the file sent by the VTAM program is:

- Not large enough, a message may be logged to the DSNX error log file that states that the received file is not complete. If the message is sent, the file that is not complete exists on your system.
- Too large, the NetView DM session may end abnormally.

**Note:** Do not confuse the VTAM NCP generation macroinstruction with the PU macroinstruction (described on page 2-4), which also has a MAXDATA parameter.

---

## Host System Work Sheet

The work sheet, shown in Figure 2-2 should be used to coordinate the AS/400 sub-system configurations and the VTAM/NCP host system generation. This work sheet can be used for the configuration of the SNUF support used with DSNX and the APPC support used to send alerts.

It is recommended that you use this work sheet in one of the following ways:

- Have the host system personnel copy and fill out the work sheet and then use those values to configure an AS/400 system for DSNX or alerts.
- You configure the AS/400 system, copy and fill out the work sheet, and then give the work sheet to the host system personnel.

Generation Parameter Description	AS/400 Value	Host Configuration Entry
The value specified for the IDNUM parameter of the PU macro during VTAM/NCP generation.	Line parameter EXCHID first 3 characters=056 for an AS/400 system	
The value specified for the ADDR parameter of the PU macro during VTAM/NCP generation.	Controller STNADR	
The value specified in the START procedure for VTAM/NCP as the SSCPID (system services control point identifier).	Controller SSCPID	
The values specified for the LOCADR parameters of the LU macros during VTAM/NCP generation. Up to 255 LU addresses can be assigned to an AS/400 system under one physical unit.	Device LOCADR	
The name of the NetView DM application ID under which the specified plan was submitted. The application ID should be provided by the NetView DM coordinator at the host site.	APPID parameter in the RLSRMTPHS command	

---

Figure 2-2. AS/400 Support VTAM/NCP Configuration Work Sheet

---

## Chapter 3. Before Running DSNX

Before you can use DSNX, you must configure the proper support on your system. This chapter discusses the following configuration topics:

- OS/400 DSNX support
  - DSNX host interface subsystem
  - DSNX request processor subsystem
  - DSNX/&pcs. subsystem
- NetView DM host to DSNX host interface configuration
- Host system logon modes
  - DSNX logon modes (via NetView DM)
  - DSNX start-up considerations
- Defining your AS/400 system to NetView DM
  - DSNX host interface-to-DSNX request processor configuration
  - DSNX host interface-to-DSNX/&pcs. configuration
  - DSNX/&pcs.-to-PC configuration
  - Defining your PC nodes to NetView DM
- DSNX/PC distribution queues

---

### OS/400 DSNX Support

The DSNX support on the AS/400 system includes the following:

- The DSNX host interface. The host interface:
  - Receives NetView DM requests from a System/370 or System/390 host through a System Network Architecture upline facility (SNUF) connection.
  - Routes NetView DM requests through object distribution and SNA distribution services (SNADS) to the end (destination) nodes.
  - Receives responses from end nodes using object distribution. **End nodes** are nodes in APPN networks that can be a source or target node, but do not provide any routing or session services to any other node.
  - Sends NetView DM responses to the System/370 or System/390 host through a SNUF connection.
- The DSNX request processor. The request processor handles DSNX distributions when the distribution gets to the AS/400 system that is the end node. The request processor:
  - Receives NetView DM requests from object distribution (or SNADS).
  - Processes the request.
  - Routes a response back to the host interface node.
- The DSNX/&pcs.. The DSNX/&pcs. is used to keep distributions for personal computers that are directly attached to a particular AS/400 system. The DSNX/&pcs.:
  - Receives NetView DM requests from SNADS.
  - Places the requests for each personal computer on a queue.
  - Sends the NetView DM requests to the personal computer when the personal computer requests them.

- Receives NetView DM responses from the personal computer and routes them into SNADS for delivery to the host interface node.

Each part of the DSNX support runs under a subsystem. For the host interface and the DSNX/pcs., you specify the subsystem to be used. For the request processor, the QDSNX subsystem is shipped with the AS/400 system.

## DSNX Host Interface Subsystem

The DSNX host interface is the part of OS/400 DSNX that communicates with the host. All host interface jobs are started by the host through the SNUF program start request. These jobs must run under a subsystem description. You can define one or make changes to an existing subsystem description, as shown in the following examples.

To start DSNX host interface jobs, a communications entry must exist in some subsystem description for the SNUF device that is used to communicate with the host. This communications entry must contain a default user ID (no special authority is needed) and a job description (which the default user is permitted to use). If no subsystem description is found containing a communications entry for the SNUF device, the system default communications entries are used.

In the same subsystem description that contains the communications entry, there must be a routing entry that specifies how the DSNX host interface jobs are started, as shown in the following examples. The **routing entry** is an entry in a subsystem description that specifies the program to be called to control a routing step that runs in a subsystem.

**Example 1:** Use the following commands if you want to use an existing subsystem. The example uses the DSNX request processor subsystem (QGPL/QDSNX) shipped with the AS/400 system. For more information about the DSNX request processor subsystem, see “DSNX Request Processor Subsystem” on page 3-4.

1. Add a communications entry for each device through which the DSNX host interface communicates with the host. In this example one entry is added for device SNUFDEV.

```
ADDCMNE SBSD(QGPL/QDSNX) DEV(SNUFDEV) JOBD(QGPL/QDSNX)
DFTUSR(QDSNX)
```

The QDSNX job description and QDSNX user profile are also shipped with the AS/400 system.

2. Add a routing entry.

```
ADDRTGE SBSD(QGPL/QDSNX) SEQNBR(20) CMPVAL('PGMEVOKE' 29)
PGM(*RTGDTA) CLS(QGPL/QDSNX)
```

The QDSNX class is shipped with the AS/400 system.

To start the QDSNX subsystem, type the command:

```
STRSBS QGPL/QDSNX
```

For this example, when the host starts the DSNX host interface, the job names are: *jobnum/QDSNX/SNUFDEV*.

**Example 2:** Use the following commands if you want to create a separate subsystem for the DSNX host interface. Although any library can be used, this example uses library QGPL.

1. Create a subsystem description called DXSBSD (any name can be used).

```
CRTSBSD SBSD(QGPL/DXSBSD) POOLS((1 *BASE))
TEXT('DSNX host interface subsystem') AUT(*USE)
```

2. Create an output queue called DXOUTQ (any name can be used). This name is used on the job description. You may also choose to use an existing output queue rather than create one.

```
CRTOUTQ QGPL/DXOUTQ TEXT('DSNX host interface output queue')
```

3. Create a user profile to be used in the communications entry (in this example DXUSR). A **user profile** is an object with a unique name that contains the user's password, the list of special authorities assigned to a user, and the objects the user owns.

```
CRTUSRPRF USRPRF(DXUSR)
```

You can also use an existing user profile.

4. Create a job description called DXJOBBD (any name can be used). This name is used in the communications entry. You may also choose to use an existing job description rather than create one.

```
CRTJOBBD JOBBD(QGPL/DXJOBBD) OUTQ(QGPL/DXOUTQ) USER(DXUSR)
LOG(4 0 *SECLVL) AUT(*CHANGE)
TEXT('DSNX host interface job description')
```

5. Add a communications entry for each device through which the host interface communicates with the host. In this example one entry is added for device SNUFDEV.

```
ADDCMNE SBSD(QGPL/DXSBSD) DEV(SNUFDEV) JOBBD(QGPL/DXJOBBD)
DFTUSR(DXUSR)
```

6. Create a class called DXCLS (any name can be used). This name is used in the routing entry. You may also choose to use an existing class rather than create one. See the *Work Management* for more information about subsystems.

```
CRTCLS CLS(QGPL/DXCLS) TIMESLICE(10000) DFTWAIT(300)
TEXT('DSNX host interface class')
```

7. Add a routing entry.

```
ADDRTGE SBSD(QGPL/DXSBSD) SEQNBR(25) CMPVAL('PGMEVOKE' 29)
PGM(*RTGDTA) CLS(QGPL/DXCLS)
```

To start the DXSBSD subsystem, type the command:

```
STRSBS QGPL/DXSBSD
```

For this example, when the host starts the DSNX host interface, the job names are: *jobnum/DXUSR/SNUFDEV*.

## DSNX Request Processor Subsystem

The request processor is the part of OS/400 DSNX that processes the request. The request processor runs under the QGPL/QDSNX subsystem, which is shipped with the AS/400 system.

- If the AS/400 system is configured as a direct node and the request processor is not active, any request sent to the AS/400 system will not be done until the request processor is started and NetView DM takes further action.
- If the AS/400 system is configured as an intermediate node and the QDSNX subsystem is not active, any request sent to the AS/400 system is placed in an internal system queue until the subsystem is started. To start the QDSNX subsystem, type the command:

```
STRSBS QGPL/QDSNX
```

The job that is started is: *jobnum/QDSNX/QDSNX*.

The QDSNX subsystem is shipped with a routing entry for the request processor automatic-start job. This routing entry should not be deleted. If the routing entry is deleted by mistake, recreate the entry by using the following commands:

```
ADDRTGE SBSD(QGPL/QDSNX)  
SEQUBR(25) CMPVAL('QDSNX')  
PGM(QSYS/QDXDDOER)  
CLS(QGPL/QDSNX)
```

## DSNX/&pcs. Subsystem

The DSNX/&pcs. is the part of OS/400 DSNX that holds the distributions sent to the personal computers that are directly attached to the AS/400 system. The personal computer starts the DSNX/&pcs. over an LU6.2 session both when the personal computer wants any distributions that are waiting on the queue on the AS/400 system and when the personal computer wants to send a response back to the host.

The personal computer communicates with an AS/400 system using an APPC session. The configuration details are discussed in "DSNX/&pcs.-to-PC Configuration" on page 3-12. Because the APPC device is automatically configured, the DSNX/&pcs. normally runs under the subsystem with the default \*APPC routing entry. If you do not want the DSNX/&pcs. jobs running under the subsystem with the \*APPC routing entry, you can create a new subsystem (or use an existing one) and add a communications entry for the APPN remote location. For more information about configuring for an APPN network, refer to the *APPN Support* or the *Communications Configuration* book.

The following example changes an existing subsystem description to run the DSNX/&pcs. jobs. The subsystem is QGPL/QDSNX, which is the DSNX request processor subsystem shipped with the AS/400 system.

1. When the session to the personal computer is established, you must determine the remote location of the personal computer. The remote location name should match the controller name for a local area network line or the device description for a twinaxial data link control (TDLC) line. For more information about the personal computer configuration, see "DSNX/&pcs.-to-PC Configuration" on page 3-12. For this example, the remote location is DXPC00.

2. Add a communications entry for the remote location.

```
ADDCMNE SBSD(QGPL/QDSNX) RMTLOCNAME(DXPC00) JOB(QGPL/QDSNX)
DFTUSR(QDSNX)
```

The QDSNX job description and QDSNX user profile are shipped with the AS/400 system.

To start the QDSNX subsystem, type the command:

```
STRSBS QGPL/QDSNX
```

For this example, when the personal computer starts the DSNX/&pcs., the job names are: *jobnum/user-ID/DXPC00*.

---

## NetView DM Host to DSNX Host Interface Configuration

Some of the values specified during DSNX configuration must match values specified at the host system during generation of the communications network.

A host system work sheet is provided in Figure 2-2 on page 2-6. You can do either of the following:

- Have the host system personnel provide you with the information for the work sheet and then use those values to configure DSNX.
- Fill out the work sheet yourself, configure DSNX using the values you specified on the work sheet, and then provide that information to the host system personnel. They can then use it to generate the host communications network.

You describe the AS/400 system support by using the configuration menu options, which present a series of displays that prompt you for the needed configuration information.

The line, controller, and device descriptions define the physical characteristics of the communications connection that is used by DSNX and a description of the host system with which they communicate. The information in the line description is needed to establish the connection with the host system.

For DSNX, you specify the following in a line description:

- The physical characteristics of the communications line
- That the AS/400 system is the secondary data link role (\*SEC)

The following control language (CL) commands are used to define the configuration for the system in a program called CRTDSNX. This example shows these commands used in a CL program; the configuration can also be defined using the configuration menus.

```

/*****/
/*
/* MODULE: CRTDSNX
/*
/*
/* LANGUAGE: CL
/*
/* FUNCTION: Configures a line, controller, and device for DSNX
/*
/* SDLC nonswitched line DSNXLIN
/* Host Controller DSNXCTL
/* SNUF Device SNUFDEV
/*
/*****/
PGM
/* Create the line description for the nonswitched SDLC line
CRTLINS DLC LIND(DSNXLIN) + /* Call the line DSNXLIN
R SRCNAME(LIN011) + /* LIN011 assigned by this system
ONLINE(*NO) + /* Do not vary on automatically
ROLE(*SEC) + /* Secondary data link role

/* Create the host controller description
CRTCTLHOST CTLD(DSNXCTL) + /* Call the controller DSNXCTL
LINKTYPE(*SDLC) + /* The line will be SDLC
ONLINE(*NO) + /* Do not vary on automatically
APPN(*NO) + /* Not APPN capable, could be (*YES)
/* depending on VTAM/NCP levels
LINE(DSNXLIN) + /* The line will be DSNXLIN
STNADR(C1) + /* Station address

/* Create the device description for SNUF
CRTDEVSNUF DEVD(SNUFDEV) + /* Call the device SNUFDEV
LOCADR(01) + /* Device local address
RMTLOCNAME(DSNXLOC) + /* Required
ONLINE(*NO) + /* Do not vary on automatically
CTL(DSNXCTL) + /* The controller will be DSNXCTL
PGMSTRRQS(*YES) + /* Program start request
RCDLEN(32761) + /* Record length DSNX will use
BLKLEN(32761) + /* Block length DSNX will use
DFTPGM(QSYS/QDXHRTR) + /* Default program
HOST(*CICS) + /* Host system uses CICS
APPLID(DSXNDM) + /* APPLID not required to match host
ENDPGM

```

See the *Communications Configuration* book for additional descriptions and examples of how to configure for a SNUF network. Also see the information about host programming considerations in the *SNA Upline Facility Programming* book.

OS/400 DSNX is an application program. It communicates with the host system by using the SNUF support. When a request is received that SNUF recognizes as being from the host, the OS/400 DSNX support is started. The SNUF support and the line, controller, and device descriptions contain the information needed for communications.

To establish communications between the AS/400 system and a remote system, use the Vary Configuration (VRYCFG) command to activate the desired line, con-



troller, and device to be used by your application. This establishes a data link between the AS/400 system and the NetView DM host system. The host system must start a NetView DM transmission to the AS/400 system. The QDSNX subsystem must be started for requests to be processed.

---

## Host System Logon Modes

The NetView DM host system obtains the parameters for the **bind command** (an SNA command used to start a session and define the characteristics of that session) it sends to the node from a logon mode table entry. The logon mode table in which the entry is found is identified by the *dlogmod* parameter on the LU or PU macroinstruction for the node. An entry within the logon mode table is defined by using the MODEENT macroinstruction.

## DSNX Logon Modes (via NetView DM)

In NetView DM, the host system refers to the *logon mode* parameter in the generalized interactive executive (GIX). **GIX** is a function of the NetView Distribution Manager licensed program that provides the host system user with interactive use of NetView Distribution Manager.

The following is an example of a DSNX logon mode.

```
ASDSNX  MODEENT LOGMODE=ASNDM4K,FMPROF=X'03',TSPROF=X'04',
          PRIPROT=X'B0',SECPROT=X'B0',
          COMPROT=X'4000',RUSIZES=X'8989',
          PSERVIC=X'00000000000000000000000000000000'
```

## DSNX Start-Up Considerations

The DSNX host interface is started after the session is bound, when the first DSNX data (such as a request header) is received. The logical unit on which transmission is started is determined by NetView DM and by the VTAM/NCP generation. The host interface subsystem and the request processor must be started at the AS/400 system for the NetView DM work to complete.

---

## Defining Your AS/400 System to NetView DM

Each node that communicates with NetView DM must be defined to NetView DM. In NetView DM, the nodes and resources to be used for OS/400 DSNX communications are defined using the GIX menu on the host system. From the Generalized Interactive Executive (GIX) Master Selection menu, select option 1 (Configure network), and then define the *node type* of each AS/400 node as SSP. Select option 4 (Define nodes), then option 1 (Create) or option 2 (Change), to define the following:

- The *node name* of an AS/400 system defined as a direct node can be any valid NetView DM node name. A request to a direct node runs on the AS/400 system that is directly attached to the host system.

The *node name* of each AS/400 system defined as an intermediate node is the current system name as defined in the system network attributes.

- The *logical unit* parameter value you specify must match the name assigned on one of the LU statements during VTAM/NCP generation.

This is the logical unit that corresponds to the SNUF device created on the AS/400 system attached to the host system.

You can use both direct and intermediate nodes if you define multiple logical units; one or more logical units for direct and one logical unit dedicated as intermediate.

- The *logon mode* parameter will show a default value that was defined during NetView DM installation. You can use this value for the AS/400 system.
- The *logon ID* parameter specifies the user ID on the end node where the request will run.
- The *password* parameter specifies the password of the AS/400 userid.
- The *connection type* can be specified as intermediate or direct.

See “NetView DM Session and Node Considerations” on page 4-8 for considerations when defining direct and intermediate nodes. You can also define these nodes with the NetView DM batch utilities.

After the nodes are defined, you can create groups of these nodes, either using option 7 (Manage groups) on the GIX menu or using the DEF GROUP control statement in the SUBMIT batch utility. Also, you must specify the node name or group name (not both) during the definition of a phase, either using option 4 (Prepare plans) on the GIX menu or using the NODE or GROUP parameter of the DEF PHASE control statement in the SUBMIT batch utility.

Resources to be used in a NetView DM transmission can be defined to NetView DM. Both resource definition and resource assignment for node type SSP can be done by selecting option 1 (Configure network) on the GIX menu.

For more information on GIX, see the *NetView Distribution Manager User's Guide*.

## DSNX Host Interface-to-DSNX Request Processor Configuration

There are two different **paths** (routes between any two nodes) distributions can take to get from the DSNX host interface to the DSNX request processor:

- If both are on the same AS/400 system, (always true when the system is a direct node) object distribution distributes the requests and responses between the DSNX host interface and the DSNX request processor as shown in Figure 3-1 on page 3-9.

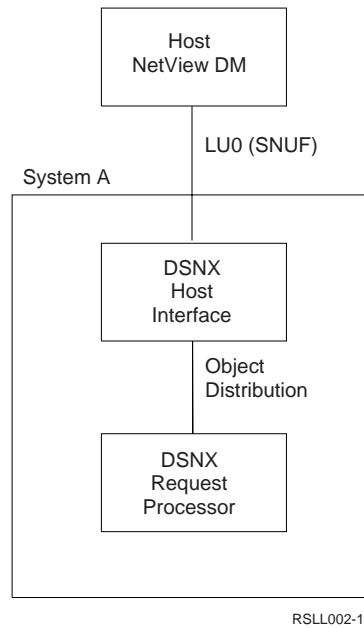


Figure 3-1. DSNX Host Interface and DSNX Request Processor on the Same AS/400 System

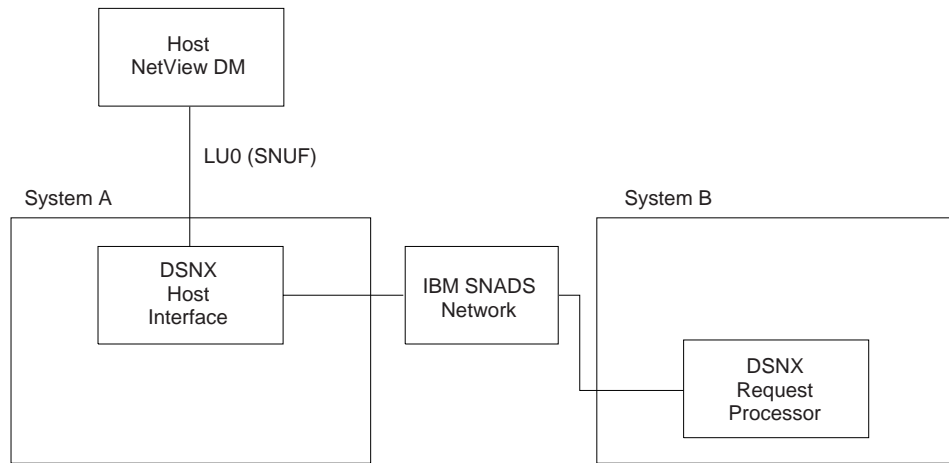
OS/400 object distribution routes requests and responses to the QDSNX user. The QDSNX user must be added to the AS/400 system directory.

In the following example, the AS/400 system name is SYSTEMA. You need to add a directory entry called QDSNX SYSTEMA to the system directory on SYSTEMA. To add this entry, type the Add Directory Entry (ADDDIRE) command.

```
ADDDIRE USRID(QDSNX SYSTEMA) USRD('QDSNX on system A')
        USER(QDSNX) SYSNAME(*LCL)
```

**Note:** The directory entry created for an intermediate request will be used for direct requests also. If only direct requests will be used, create an entry as if intermediate requests were supported.

- If the end node is a different AS/400 system, the requests and responses are routed to the other node using object distribution, which puts them into a SNADS network as shown in Figure 3-2 on page 3-10.



RSL003-1

Figure 3-2. DSNX Host Interface and DSNX Request Processor on Different AS/400 Systems

In this case, you must not only add a directory entry for QDSNX on the host interface node as you did in the previous example, but you must also add a directory entry for QDSNX on the AS/400 end node. Assuming the end node is SYSTEMB, you would type the following command on the end node AS/400 system:

```
ADDIRE USRID(QDSNX SYSTEMB) USRD('QDSNX on system B') USER(QDSNX)
      SYSNAME(SYSTEMB)
```

If the end node is not the host-attached node, you must configure the SNADS routing information at the intermediate systems needed to distribute the request to (and the responses from) the end node. For more information regarding descriptions and examples of how to configure for a SNADS network, refer to the *SNA Distribution Services*.

OS/400 DSNX uses the user ID and the password specified in the NetView DM phase to verify authorization to OS/400 objects. This user ID does not need to be in the system directory for DSNX processing. The user profile and password must be defined on the target system. The **target system** is the system that receives a request from another system to establish communications. The user needs authority to objects specified by NetView DM functions. DSNX performs functions for a specific userid; therefore the authority required is the same as if the user had requested the function interactively.

## DSNX Host Interface-to-DSNX/&pcs. Configuration

All distributions sent to the personal computer are routed to an intermediate AS/400 system via SNADS as shown in Figure 3-3 on page 3-11.

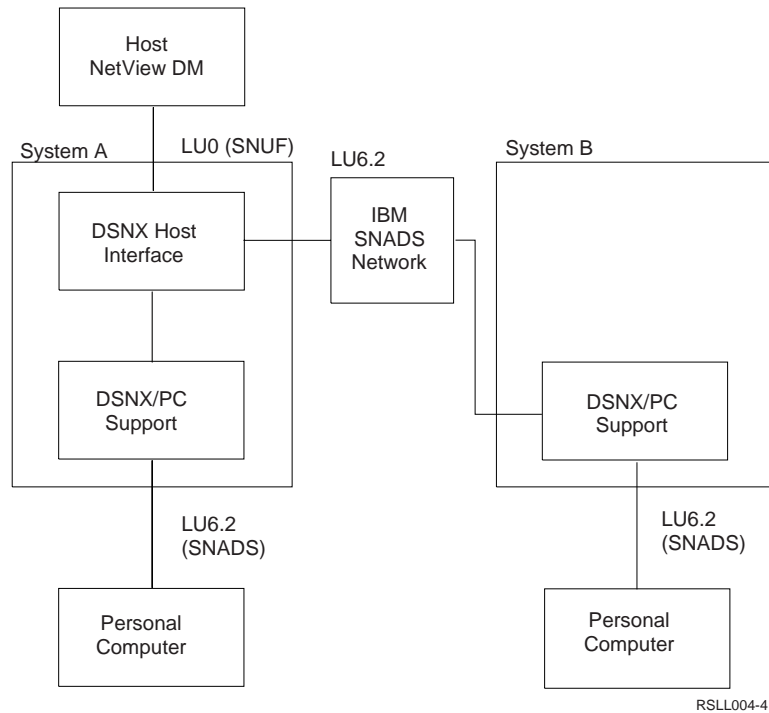


Figure 3-3. Distribution to a Personal Computer with an AS/400 System as an Intermediate Node

As in the DSNX host interface-to-DSNX request processor configuration, you must have the QDSNX user defined on the node that attaches to the personal computer. In addition, on the AS/400 system that has the attached personal computer, you must add a directory entry for each personal computer. Assuming the personal computer is PC1 (the node name defined by NetView DM) and the address is SYSTEMB, you would type the following command:

```
ADDIRE USRID(PC1 SYSTEMB) USRD('PC 1 attached to system B')
      USER(*NONE) SYSNAME(*PC)
```

In this example, directory entries need to be added on System A and System B.

You can add a directory entry for each personal computer or you can add one directory entry for all personal computers as follows:

```
ADDIRE USRID(*ANY SYSTEMB) USRD('All PC names')
      USER(*NONE) SYSNAME(*PC)
```

**Notes:**

1. Only NetView DM distributions are routed to directory entries using \*PC as the system name.
2. To send NetView DM distributions to a personal computer, the NetView DM PDOS node name must match the user ID in the OS/400 directory entry for the personal computer.
3. The address in the directory entry for the personal computer must match the AS/400 name defined in the network attributes.
4. When an AS/400 is receiving two or more concurrent NvDM distribution, the user must use different logical unit (LU) numbers on the same or different con-

trollers. If the AS/400 system is not receiving concurrent NvDM distribution, then the identical LU numbers can be maintained.

## DSNX/PCS-to-PC Configuration

The AS/400 system communicates with a personal computer using SNADS over an APPC session. The physical line can be either a **local area network**<sup>1</sup> (LAN) line, a **twinaxial data link control**<sup>2</sup> (TDLC) line, FDDI, a Frame Relay, ISDN, or X.25.

- To configure a local area network line, you must configure either a **token-ring network**<sup>3</sup> or an **Ethernet**<sup>4</sup> network line description and a controller description. The *Client Access/400 for DOS and OS/2 Technical Reference* book contains information about these configurations.
- For a TDLC line, you do not have to do any special configuration if the system value is set to indicate that work stations are automatically configured. To find out if this value is set to automatically configure, type this command:

```
DSPSYSVAL SYSVAL(QAUTOCFG)
```

If the system is not set to automatically configure, you should type the following command:

```
CRTDEV DSP DEVD(DXPC00) DEVCLS(*LCL) TYPE(5150) MODEL(n)  
PORT(n) SWTSET(n) CTL(ct1nn)
```

Model, port, switch setting, and controller must be determined for each system. The *Client Access/400 for DOS and OS/2 Technical Reference* contains information about defining a TDLC line.

After your lines are configured, start the PC **router**, which handles requests to send and receive data from applications on the personal computer and routes them to the appropriate applications on the AS/400 system. For more information about configuring your personal computer, refer to the *Client Access/400 for DOS with Extended Memory Setup* or the *Client Access/400 for OS/2 Setup*.

For a TDLC line, sign-on prompts are displayed. You enter a user ID with the authority to use the APPC device that is automatically configured on the AS/400 system. If your AS/400 system is a secure system, you are prompted for a password.

The user ID that is used to start the PC router must have \*USE authority to the QDXPSEND and QDXPRCV program objects in the QSYS library. This is required for all DSNX PC users. Use the Grant Object Authority (GRTOBJAUT) command to change the authority to \*USE.

When the commands are entered, the AS/400 system automatically configures a line description, controller description, and device description. The session is now

---

<sup>1</sup> The physical connection that allows the transfer of information among devices located on the same premises.

<sup>2</sup> A communications function that allows personal computers, which are attached to the work station controller by twinaxial cable, to use APPC or APPN.

<sup>3</sup> A local area network that sends data in one direction throughout a specified number of locations by using the symbol of authority for control of the transmission line, called a token, to allow any sending station in the network (ring) to send data when the token arrives in that location.

<sup>4</sup> A type of local area network that is supported by the OS/400 licensed program. OS/400 Ethernet provides for support for the Digital Equipment Corporation, Intel\*\* Corporation, and Xerox\*\* standard (Ethernet Version 2) and the IEEE 802.3 Standard. These local area networks use Carrier Sense Multiple Access with Collision Detection (CSMA/CD) as the media access method.

ready for the personal computer to ask the AS/400 system if there are any distribution queues for the personal computer. For the personal computer to ask the AS/400 system, the DSNX/PC transmission executive must be started on the personal computer by typing the following command:

```
\PCDSNX\DSNXTE
```

This command starts a job in the subsystem that has a communications entry for the APPN remote location. Refer to “DSNX/&pcs. Subsystem” on page 3-4 for information about communications entries for the APPN remote location. For more information on the DSNX/PC transmission executive, refer to the *Personal Computer/Distributed System Node Executive Installation and Operation* book.

## Defining Your PC Nodes to NetView DM

To define a personal computer that connects to a particular AS/400 system, use the GIX menus. From the GIX Master Selection menu, select option 1 (Configure network) to define the following:

- The *node type* of each personal computer should be defined as PDOS.

Then select option 4 (Define nodes), option 1 (Create) or option 2 (Change), to define the following:

- The *node name* is the remote location name (the controller name for a local area network or the device description for a TDLC line) of the personal computer. This is defined on the personal computer using the \PCDSNX\DSNXCO command and using the SNADS parameter table (DSNXSNDS.TAB).
- The *logical unit* parameter value you specify must match the name assigned on one of the LU statements during the VTAM/NCP generation for the host-attached system.
- The *directory name* is the AS/400 system name. Use the Display Network Attributes (DSPNETA) command to display the system name of the AS/400 system that the personal computer is attached to.
- The *connection type* must be intermediate.

---

## Work with DSNX/PC Distribution Queues (WRKDPCQ)

The Work with DSNX/PC Distribution Queues (WRKDPCQ) command shows a series of displays that allow the network operator to display, print, or delete DSNX distributions on the queues for personal computers attached to an AS/400 system. The displays show:

- DSNX distributions pending delivery for all personal computers.
- DSNX distributions pending delivery for a selected personal computer.
- Confirmation for deletion of distributions.

When you type WRKDPCQ on the command line and press the Enter key, a display similar to the following is shown:

```

                                Work with DSNX/PC Queues

Type options, press Enter.
 5=Work with queue entries

Option      DSNX/PC      Number of      Status
            Node        Queue Entries
-          PC1            1             Ready
-          PC2            4             Sending
-          PC3            5             Ready
  
```

This display shows information about the DSNX distributions waiting to be sent to a DSNX/PC node. The display shows a list of the personal computers that have distributions placed on a queue and the status of the queue. A status of *Ready* indicates the AS/400 system is not actively sending distributions from the queue to the personal computer node. A status of *Sending* indicates that distributions are being sent.

When you select a specific personal computer node name to work with, either using the WRKDPCQ command or selecting option 5 (Work with queue entries) from the Work with DSNX/PC Queues display, a display similar to the following is shown:

```

                                Work with DSNX/PC Queue Entries

DSNX/PC node . . . . . : PC2
Status . . . . . : Sending

Type options, press Enter.
 4=Delete

-----NetView DM----- Remote LU DSNX DSNX/PC
Opt Plan/Phase Date Time Location Nbr Origin Action
- K2180920/PHASE1 03/03/88 14:35 DSXMVS1 06 RCH36BL2 Inform
- A3031434/PHASE1 03/03/88 14:35 DSXMVS1 06 RCH36BL2 Send
- A3031434/PHASE2 03/30/88 14:35 DSXMVS1 06 RCH36BL2 Retrieve
- A3031435/PHASE3 03/03/88 14:35 DSXMVS1 06 RCH36BL2 Send

Bottom
F3=Exit F5=Refresh F11=Display additional DSNX information F12=Cancel
F14=Delete all F17=Top F18=Bottom
  
```

From this display, you can select queue entries to be deleted.

**Note:** You cannot delete the first distribution in the queue when the status is set to *Sending*.



When you press F11 (Display additional DSNX information), a display listing the DSNX/PC resource for each plan/phase is shown.

Whenever you select queue entries to be deleted, a confirmation display similar to the following is shown to allow you to confirm or change those entries you selected to be deleted.

```
Confirm Delete of DSNX/PC Queue Entries

DSNX/PC node . . . . . : PC2
Status . . . . . : Sending

Press Enter to confirm your choices for 4=Delete.
Press F12 to return to change your choices.

-----NetView DM----- Remote LU DSNX DSNX/PC
Opt Plan/Phase Date Time Location Nbr Origin Action
4 K2180920/PHASE1 07/17/87 12:50 DSXALT 01 RCH38329 Inform
4 A3031434/PHASE1 03/03/88 14:35 DSXMVS1 06 RCH36BL2 Send
4 A3031434/PHASE2 03/03/88 14:35 DSXMVS1 06 RCH36BL2 Retrieve
4 A3031434/PHASE3 03/03/88 14:35 DSXMVS1 06 RCH36BL2 Send

F11=Display additional DSNX information F12=Cancel Bottom
```

By pressing F11 (Display additional DSNX information), a display listing the DSNX/PC resource for each plan/phase is shown.



---

## Chapter 4. DSNX Considerations

This chapter discusses the following considerations when using DSNX:

- DSNX naming conventions
- DSNX scheduling
- DSNX security considerations
- Coexistence considerations
- Sending the same resource to several nodes
- DSNX processing considerations
- NetView DM session and node considerations
- Data considerations using NetView DM

---

### DSNX Naming Conventions

Certain conventions must be followed when naming AS/400 libraries, objects, and members that are to be used with NetView DM. These conventions are to be used when supplying the *name at node* on GIX.

### Object Naming

The name at node resource qualifiers correspond to the object name on the AS/400 system in one of the following ways:

- library.object.type

**library**

The 1 through 8 character name of the AS/400 library.

**object**

The 1 through 8 character name of the AS/400 object.

**type**

The object type as used on the AS/400 system. The object type must be valid for the OBJTYPE parameter in the Save Object (SAVOBJ) command.

- library.file.member.MEM

**library**

The 1 through 8 character name of the AS/400 library.

**file**

Source physical file (database file).

**member**

Member in the database file.

**MEM**

Indicates this is a database file member.

- library.file.MEM

This form can be used when retrieving, sending, or deleting date-differentiated file members. You may also use the library.file.member.MEM format described above for date-differentiated file operations, but the member name must always be specified. If you choose to use the library.file.MEM format, AS/400 DSNX assigns a member name as follows (yymmdd is the current date on the AS/400 system):

NetView DM Request	AS/400 DSNX Action
RTV lib/file MEM	Return member lib.file (*LAST) to NetView DM.
SEND lib/file MEM (add option)	Check for the existence of member lib.file.(Myymmdd). If the member exists, do not replace it; return an error to NetView DM. If the member does not exist, add member lib.file.(Myymmdd).
SEND lib/file MEM (replace option)	Check for the existence of member lib.file.(Myymmdd). If the member exists, replace it. If the member does not exist, add member lib.file.(Myymmdd).
DELETE lib/file MEM	Delete member lib.file.(Myymmdd).

#### Notes:

1. If MEM is not specified as the last qualifier, the last qualifier is ignored, and the object name uses the format library.object.type, as the following example shows:

If the name at node specified on NetView DM is:  
MYLIB.MYFILE.MYMEM.FILE  
the AS/400 system uses:  
MYLIB/MYFILE and a type of \*MYMEM  
\*MYMEM is not a valid type and an error  
message is sent to NetView DM.

2. A GIX restriction specifies that resource qualifiers must be 8 or fewer characters.
3. If the destination of the file is a system other than an AS/400 system, the last qualifier (MEM) must be used. This prevents OS/400 DSNX from saving the object as a save file. The only AS/400 object that can be sent to a destination that is not an AS/400 system is a database file member. To retrieve a database file member, the last qualifier must be MEM.
4. To retrieve a file that is already in the save file format, the three-part name (library.object.SAVF) should be used.

---

## DSNX Scheduling

To make the most efficient use of system resources, AS/400 users should work closely with the NetView DM coordinator at the host site to schedule data transmissions. The AS/400 system operator can also control when a request is sent to the AS/400 system by using the Release Remote Phase (RLSRMTPHS) command.

## Releasing a Phase

A NetView DM transmission plan must be prepared and submitted at the host system before resources can be transmitted. A *plan* groups together transmission activities. An individual transmission activity for a specific destination is called a *phase*. A transmission plan can contain one or more phases.

Each phase of a transmission plan can be assigned a day and a time when the transmission activity is to take place. For example, new programs can be sent and installed on the AS/400 system during nighttime hours so that daily work will not be interrupted. You can release phases from NetView DM by typing the Release Remote Phase (RLSRMTPHS) command at the directly-attached AS/400 system.

When you enter the RLSRMTPHS command and press F4 (Prompt), a display similar to the following is shown:

```
Release Remote Phase (RLSRMTPHS)

Type choices, press Enter.

Phase . . . . . THEPHASE name
Plan . . . . . THEPLAN name
Application identifier . . . . . APPL1 name
Remote location . . . . . SYSB name
Device . . . . . DEVICE1 name
```

The following parameters are supported for the RLSRMTPHS command.

**PHASE Parameter**

Specifies the name of the NetView DM phase to be released. The phase must exist on the NetView DM host system as part of the plan specified by the PLAN parameter. The phase must exist on the NetView DM host with a HELD status. The maximum length is 8 characters.

**PLAN Parameter**

Specifies the name of the NetView DM plan that contains the phase to be released. The maximum length is 8 characters.

**APPID Parameter**

Specifies the name of the NetView DM application under which the plan name specified in the PLAN parameter was submitted. The application ID should be provided by the NetView DM coordinator at the host site. The maximum length is 8 characters.

**RMTLOCNAME Parameter**

Specifies the name of the remote location associated with the remote system with which this device communicates. This parameter must match the DEVD parameter in the Create Device SNUF (CRTDEVSNUF) command. Usually the remote location name is the name of the logical unit on the host system. The maximum length is 8 characters.

**DEV Parameter**

Specifies the name of the AS/400 device that will be used for the communications session started as a result of this command. The device must be a SNUF device and be able to accept a program start request. The maximum length is 10 characters.

Following are some general considerations for using the RLSRMTPHS command:

- Uses the same device as other DSNX processing.
- Can only release phases that use the same logical unit (LU) as specified in the LOCADR parameter in the CRTDEVSNUF command.
- The Release Remote Phase (RLSRMTPHS) command can only be run on a node that is directly attached to the host. You can release phases for other nodes from the node that is the host interface for those nodes.
- End nodes must use the same host-attached node.

- QSYSOPR and QPGMR have \*USE authority; other users have \*EXCLUDE authority. To grant individual users authority to this command, use the Grant Object Authority (GRTOBJAUT) command.
- Message CPC8889 indicates that the specified phase was successfully released.

Message CPF8880 is an escape message indicating that the phase was not released. Batch programs and CL programs should monitor for this message.

An example of creating a SNUF device used by the RLSRMTPHS command is provided on page 3-6.

---

## DSNX Security Considerations

All DSNX objects are managed by the OS/400 security support. The AS/400 programmer should be aware of the following security considerations:

- Header information received in transmissions from NetView DM to a DSNX receiver includes password information that is not encrypted. This password is not checked before being sent through intermediate nodes in the NetView DM network and then verified by the target DSNX receiver node. DSNX receiver systems must have user profiles created for each NetView DM user. The target system is responsible for maintaining the AS/400 user profile and password information for checking the NetView DM transmissions. The NetView DM user password should not exceed 6 characters.
- Because the personal computers are not password-protected, the DSNX/&pcs. has no password protection. No password is sent to personal computers in the NetView DM network.
- The DSNX request processor uses the user profile and password sent in the NetView DM header for running the NetView DM request. The AS/400 user profile must have sufficient authority to objects required for the NetView DM request.
- The DSNX request processor does not differentiate between an expired and an unexpired user password; therefore, the NetView DM request will be started even if the user password has expired.

---

## Coexistence Considerations

Within a NetView DM network, AS/400 systems using DSNX may coexist with System/36s using DSNX. AS/400 database members can be retrieved by NetView DM and then sent to a System/36 as a file. However, an AS/400 CLIST must be converted to System/36 OCL before it can be successfully run on a System/36.

All other types of AS/400 objects are retrieved by NetView DM in an AS/400 save file format. They should be sent only to other AS/400 systems in the NetView DM network. That is, the receiving node must be an AS/400 system. However, any intermediate nodes between the NetView DM host system and the receiving node can be AS/400 systems or System/36s.

If the logical record length of a data file sent to a System/36 does not match the logical record length of the existing System/36 file, the System/36 deletes the

existing file and creates a new file, using the logical record length of the file that was sent.

If the logical record length of a data file sent to an AS/400 system does not match the logical record length of the existing AS/400 file, an error message is sent to NetView DM.

---

## **Sending the Same Resource to Several Nodes**

When a resource is sent to more than one node with a type of intermediate, the requests can be chained for improved performance. This can be done by defining a node group and sending the resource to that node group with a specified user ID and password, or with the default node-dependent user ID and password. The requests having the same user ID, password, and resource name are sent from the host interface node to other nodes in the same SNADS distribution. The resource for a node group is transmitted only once from the NetView DM host to the intermediate node.

Only add, replace, and run requests are chained.

---

## **DSNX Processing Considerations**

DSNX functions include:

- Processing batch jobs submitted to the AS/400 system by the host
- Replacing objects on the AS/400 system

## **AS/400 Processing of CLISTs Sent by NetView DM**

On the host system, the INITIATE CLIST NetView DM control statement is used to submit batch jobs to the AS/400 system. The CLIST (batch job stream) can be created at the host or created at the AS/400 system and retrieved by NetView DM before INITIATE CLIST is started. These CLISTs (batch job streams) are processed by the OS/400 DSNX request processor as follows:

- The job stream is put into a temporary database member.

**Notes:**

1. The maximum record size is 32 766 bytes.
  2. The maximum CLIST size is 65 280 bytes.
- The job stream is started using the Submit Database Job (SBMDBJOB) command.
  - A check is made to ensure all the jobs in the job stream are submitted successfully.
  - The DSNX request processor waits for all jobs to finish running. Because the DSNX request processor handles requests from the host one at a time, the DSNX request processor does not run another request until all the submitted jobs complete.
  - Ending status is not presented to NetView DM until the batch job is complete.
  - After waiting approximately an hour, the DSNX request processor gives a warning message (CPD8869) to indicate the job is still running.

- After waiting about three and a half hours, a second message is given (CPD8870) to indicate the job has been running a long time and that some action should be taken to end the job.
  - Regardless of the messages, the DSNX request processor continues to wait until the job completes.
- Note:** Whenever possible, an automatic response should be given to all inquiry messages resulting from CLISTs being processed by the AS/400 system. For those messages not allowing automatic response, DSNX processing stops until the system operator responds to the message. This may occur, for example, if there is a programming error in one of the programs called by a job in the job stream.
- When the jobs complete, the DSNX request processor checks to see if any *user ending status* was given. If none is found, a primary return code of zero is sent back to the host in the DSNX response. The DSNX request processor does not evaluate the success or failure of the submitted job. To provide user ending status you must:
    - Send program message CPC8888 to the QGPL/QDSNX message queue. The message data provided with CPC8888 is a 4-character value that should represent the primary return code to be sent back to the host. This number should be given in ZONED(4,0) format (that is, 4 digits with no decimal point). If a value that is not valid is specified, the value is ignored. Examples of values that are not valid are:  
'8', '8bbb', '8!'.
    - To send a program message, the job stream must call a command language program that uses the Send Program Message (SNDPGMMSG) command as follows:

```
SNDPGMMSG MSGID(CPC8888) MSGF(QCPFMSG) MSGDTA('0008')
          TOMSGQ(QGPL/QDSNX)
```

In this example, a return code of 8 would be sent as the primary return code to the host system (GIX). The primary return code is presented to the host (interactive operator facility (IOF)) operator. The number specified in the program message (MSGDTA) is converted to hexadecimal when sent to NetView DM.

**Notes:**

1. Use of user ending status is an optional function.
  2. AS/400 batch job streams do not have the function to monitor for messages (MONMSG). The batch job stream can call a CL program which can use MONMSG.
- If more than one CPC8888 message is found on the message queue, the largest return code is sent back to the host.

See Chapter 5, “Sample Procedures for DSNX” for examples of CLISTs.



## Replacing Objects on an AS/400 System

The DSNX request processor replaces objects on your AS/400 system in the following manner:

- If the object is a member:
  - A temporary member (QDSNX\$TMBR) is created in the database file where the member exists.  
**Note:** If the file where the temporary member is created has the MAXMBRS parameter set to 1, an error message is returned. Set the MAXMBRS parameter to \*NOMAX using one of the following commands:
    - Change Source Physical File (CHGSRCPF)
    - Change Physical File (CHGPF)
  - The temporary member is filled with the data in the request.
  - The actual member is deleted if it exists.
  - The temporary member is renamed to the actual name given in the request.  
**Note:** The member *type* and *text* information is not transferred when a member is sent to a host system and then replaced on the AS/400 system. After the member is replaced, the type and text areas will be blank.
- If the object is a save file:
  - A temporary save file (QDSNXSAV) is created in the library where the save file exists.
  - The temporary save file is filled with the data in the request.
  - The actual save file is deleted if it exists.
  - The temporary save file is renamed to the actual name given in the request.
- For any other object:
  - A temporary save file (QDSNXSAV) is created in the QTEMP library.
  - The temporary save file is filled with the data in the request.
  - The actual object is deleted if it exists.
  - The object is created in the library that is specified in the request using the Restore Object (RSTOBJ) command.

## Retrieving Objects Other Than Members or Save Files

The DSNX request processor retrieves objects on your AS/400 system in the following manner:

- If the object is a member or a save file:
  - The member or file is opened.
  - The data is read from the member or file and placed in the NetView DM response.
- For all other objects:
  - A temporary save file (QDSNXSAV) is created in the QTEMP library.
  - The temporary save file is filled with the object data by using the Save Object (SAVOBJ) command with the DTACPR parameter set to \*Yes.
  - The temporary save file (QDSNXSAV) is opened.

- The object data is read from the save file and placed in the NetView DM response.

---

## NetView DM Session and Node Considerations

Communications between an AS/400 system and a host system requires multiple sessions to be established. These sessions are started by commands in the job stream.

The AS/400 system can be defined to NetView DM as either direct node type or an intermediate node type. Each node type has advantages and disadvantages at a session level.

## NetView DM Sessions

There are three types of sessions involved when NetView DM and a System/370 system communicates with DSNX on an AS/400 system using the SNA Upline Facility (SNUF) protocol:

- **System services control point (SSCP)**<sup>1</sup> to physical unit (PU). This session is started by the Activate Physical Unit (ACTPU) command. This allows a VTAM system running on a System/370 host to communicate with an AS/400 controller (PU).
- SSCP to logical unit (LU). This session is started by the Activate Logical Unit (ACTLU) command. This allows a VTAM system running on a System/370 host to communicate with an AS/400 device (LU).
- LU to LU. This session is started by the BIND command. It allows a VTAM Logical Unit to communicate with an AS/400 device (LU).

All NetView DM to DSNX requests and responses are carried on an LU to LU session except for a LOGON which is carried on the SSCP to LU session.

## Direct Node

An AS/400 system defined as a direct node does not use multiple sessions as a system defined as an intermediate node does. Each function within a phase will run in a single session. The session will remain active while the following occurs:

- The data is sent to the AS/400 host interface.
- The request processor is running (at this time, no DSNX data is being transferred over the communications line to the host).
- The response is sent to NetView DM.

Direct node type would probably result in longer session times but considerably fewer dial operations when using a switched line.

If direct node type is used, the request processor must be available when the host interface job is started or the request cannot be completed.

---

<sup>1</sup> A focal point within an SNA network for managing the other systems and devices, coordinating network operator requests and problem analysis requests, and providing directory routing and other session services for network users.

## Intermediate Node

If the request processor is not on the same AS/400 system as the host interface, intermediate node support must be used.

The request processor may take a considerable amount of time to process a request once it has been sent from NetView DM to the host interface. When this type of delay occurs for an AS/400 system defined as an intermediate node, the session is ended by NetView DM. When the DSNX request processor completes the function, the DSNX host interface will log on to the System/370 host to notify NetView DM that the response is available. Because the session is ended and then restarted when the function completes, long session times are avoided for long running functions.

Multiple sessions may be an advantage on long-running functions. However, functions that run quickly may not run efficiently on multiple sessions. If the processing delay is long enough to cause NetView DM to end the session, at least three sessions will occur to complete the request:

1. NetView DM sends a request which causes the VTAM program to start a session.
2. The AS/400 host interface sends a logon command to the VTAM program running on the System/370 host.
3. NetView DM, through the VTAM program, starts a third session to get the response from the request processor.

Functions that run quickly may cause these multiple sessions because of the timing between the AS/400 system and the System/370 host. Switched line users should consider defining the AS/400 system as a direct node to reduce the number of dialing operations. Automatic dial and automatic answer configurations should also be considered.

It is the responsibility of the AS/400 system to notify NetView DM when a response is available. The AS/400 system will continue to try to log on until the logon is sent to NetView DM. The time interval between logon attempts can be controlled by using the Change ICF File (CHGICFF) command to update the WAITFILE parameter on the QDXHCMNF communications file. Approximately 25 seconds is added to the value of the WAITFILE parameter to set the interval between logon attempts. The host interface job must remain active (not HELD or ENDED) for the logon attempts to continue.

---

## Data Considerations Using NetView DM

NetView DM has functions available to prepare a Multiple Virtual Storage (MVS) data set so that it is compatible with AS/400 database files. The prepared MVS data set is held in NetView DM resource storage. The following applies to preparing data for the AS/400 system:

- Only data that will be sent (Add, Replace, or Decompress requests) to AS/400 database members can be prepared. All other object types that are sent to the AS/400 system must be in Save File format in NetView DM storage (the data was retrieved from an AS/400 system).
- When data is prepared from MVS variable record length data sets the data is sent to the AS/400 system as a continuous stream of data. Because the

AS/400 system does not support variable length records, the data will be assumed to be fixed record length data. For example, if the MVS data set has 92 byte records, the AS/400 system will break up the data stream into 92 byte pieces and store the data in a database member having a 92 byte record length. If there is less than 92 bytes in the final record, it is filled with blanks.

A NetView DM prepare function should not be done on variable record length data sets unless the receiving node is expecting the database to contain the resulting broken stream of data.

- NetView DM allows users to specify whether the data being retrieved should be compressed by the AS/400 system. The AS/400 system ignores this option for all object types except for database members.
- If a data set in the NetView DM resource storage is filled with compressed data, the data can be sent to an AS/400 system, but only with a *Replace* function with *Decompress* option. If compressed data is sent without the decompress option, an error message is issued by the host interface and sent to NetView DM.
- The amount of data to be prepared is indicated in the Prepare statement by the following parameters:
  - Logical record length (LRLEN)
  - Logical block length (LBLEN)
  - Primary allocation (NLBLR)
  - Secondary allocation (EXT)

If the amount of data to be prepared exceeds the data amount indicated in the Prepare statement, the AS/400 system sends an error message to NetView DM.

- When data is prepared for a source physical file, a 12-byte sequence number must be included at the beginning of each record. The format for the sequence number is **nnnnnyymmdd**, where **nnnnnn** is the line number and **yymmdd** is the date when the record was last changed (the date can be all zero characters).

For example, **000001900215** indicates line one and a record change date of February 15, 1990.

Approximately 2 gigabytes ( $2^{31}$ ) of data can be added or replaced to, or retrieved from an AS/400 system (if the AS/400 system has sufficient storage). The maximum number of bytes may be slightly less than 2 gigabytes because of space used by DSNX request or response headers.

## Considerations Using DSNX on the AS/400 System

The following considerations apply to DSNX used on the AS/400 system:

- Resource group name  
NetView DM allows one name to be used for several resources (sometimes called a resource group). When using NetView DM functions such as Add and Replace, a resource group name can be used wherever a resource name would usually be specified. However, an AS/400 system configured as a direct node does not support the use of resource group names for the Retrieve (compressed data) or the Replace (decompress data) functions.
- Application ID or LU number

AS/400 information about outstanding requests is available only for requests submitted under sessions using the same application ID and LU number as defined in the current session.

- Compressed data

Compressed data cannot be stored in AS/400 database members. When sending compressed data to an AS/400 database member, the Decompress option must be specified in the NetView DM Send function. Also, compression only compresses AS/400 file members, it does not compress the AS/400 file.



---

## Chapter 5. Sample Procedures for DSNX

The procedures described in this chapter allow you to perform functions not directly supported by OS/400 DSNX. These procedures describe how to:

- Distribute application programs to AS/400 systems that use a previous release of the OS/400 licensed program.
- Distribute an entire application program library from one AS/400 system to another.
- Retrieve the history file from one AS/400 system and print it at another AS/400 system.
- Transfer **program temporary fixes**<sup>1</sup> (PTFs).
- Transfer spooled file entries.

---

### Distributing Programs to Systems that Use a Previous Release

The following procedure shows how you can transfer an application program to AS/400 systems that use a previous release of the OS/400 licensed program. Although DSNX allows individual objects to be transferred one at a time, it does not directly support transferring objects to a system with a previous OS/400 release.

A single AS/400 system can be used as the development source system for several AS/400 sites (all the AS/400 systems are connected to a NetView DM network). When a set of application programs is developed, those programs can be sent to the other AS/400 systems using DSNX. The target systems can be at the current OS/400 release or a previous release. It should be stated in the program if the program is intended to be sent to a system at a previous release.

Considerations that apply to the following example are:

**Note:** The program in the following example must have been created for the previous OS/400 release.

The user ID under which the requests in the following example run on the AS/400 system must be authorized to `TEMPLIB/SAVEFILE`.

1. The host system NetView DM operator initiates a CLIST (batch job) at the source AS/400 system to save the program into a save file. Use the Grant Object Authority (`GRTOBJAUT`) command to authorize the user ID, under which DSNX will run on the AS/400 system, to the `SAVEFILE`.

---

<sup>1</sup> A temporary solution to or bypass of a problem diagnosed by IBM as resulting from a defect in a current unaltered release of a licensed program.

For example, the CLIST could be like the following:

```
//BCHJOB JOB(DSNXJOB)
CRTLIB TEMPLIB
CRTSAVF TEMPLIB/SAVEFILE
GRTOBJAUT OBJ(TEMPLIB/SAVEFILE) +
           OBJTYPE(*FILE)         +
           USER(user ID)          +
           AUT(*ALL)
SAVOBJ OBJ(objname) LIB(libname) DEV(*SAVF) OBJTYPE(*PGM) +
        SAVF(TEMPLIB/SAVEFILE) TGTRLS(*PRV)
//ENDBCHJOB
```

2. The host system NetView DM operator retrieves the save file from the source AS/400 system. The GIX name at node is:

```
TEMPLIB.SAVEFILE.SAVF
```

3. The host system NetView DM operator initiates a CLIST at the target AS/400 systems to create a temporary library. For example:

```
//BCHJOB JOB(DSNXJOB)
CRTLIB TEMPLIB
//ENDBCHJOB
```

4. The host system NetView DM operator adds the save file to the target AS/400 systems. The ID of the requesting user sent with the request must be authorized to the TEMPLIB created in step 3. The Grant Object Authority (GRTOBJAUT) command can be used to grant authority, or have the job in step 3 run under the ID of the requesting user.

5. The host system NetView DM operator initiates a CLIST at the target AS/400 systems to update the target library. For example:

```
//BCHJOB JOB(DSNXJOB)
RSTOBJ OBJ(objname) SAVLIB(libname) DEV(*SAVF) OBJTYPE(*PGM) +
        SAVF(TEMPLIB/SAVEFILE)
DLTLIB TEMPLIB
//ENDBCHJOB
```

---

## Distributing an AS/400 Application Library

The following procedure shows how you can transfer an entire library using DSNX. Although DSNX allows individual objects within a library to be transferred one at a time, it does not directly support transferring an entire library at one time.

A single AS/400 system can be used as the development source system for several AS/400 sites (all the AS/400 systems are connected to a NetView DM network). When a set of application programs is developed, those programs can be sent to the other target AS/400 systems using DSNX.

The NetView DM administrator creates plans to initiate the CLISTs (batch jobs) and submits the plans to NetView DM. The NetView DM operator starts the plans. The CLISTs must begin with //BCHJOB and end with //ENDBCHJOB.



Considerations that apply to the following example are:

**Note:** The user ID under which the requests in the following example will run on the AS/400 system must be authorized to TEMPLIB/SAVEFILE, (libname), and to the objects in (libname).

CLISTs may be submitted from the host system without operator action at the source or target AS/400 systems.

1. The host system NetView DM operator initiates a CLIST (batch job) at the source system to save the library into a save file. Use the Grant Object Authority (GRTOBJAUT) command to authorize the user ID under which DSNX will run on the AS/400 system. For example, the CLIST could be like the following:

```
//BCHJOB JOB(DSNXJOB)
CRTLIB TEMPLIB
CRTSAVF TEMPLIB/SAVEFILE
GRTOBJAUT OBJ(TEMPLIB/SAVEFILE) +
           OBJTYPE(*FILE)         +
           USER(user ID)          +
           AUT(*ALL)
SAVLIB LIB(libname) DEV(*SAVF) SAVF(TEMPLIB/SAVEFILE)
//ENDBCHJOB
```

2. The host system NetView DM operator retrieves the save file from the source AS/400 system. The GIX name at node is:

```
TEMPLIB.SAVEFILE.SAVF
```

3. The host system NetView DM operator initiates a CLIST at the target system to create a temporary library. For example:

```
//BCHJOB JOB(DSNXJOB)
CRTLIB TEMPLIB
//ENDBCHJOB
```

4. The host system NetView DM operator adds the save file to the target AS/400 systems. The ID of the requesting user sent with the request must be authorized to the TEMPLIB created in step 3. The Grant Object Authority (GRTOBJAUT) command can be used to grant authority, or have the job in step 3 run under the ID of the requesting user.

5. The host system NetView DM operator initiates a CLIST at the target system to update the target library. For example:

```
//BCHJOB JOB(DSNXJOB)
CLRLIB libname (libname is assumed to exist)
RSTLIB SAVLIB(libname) DEV(*SAVF) SAVF(TEMPLIB/SAVEFILE)
DLTLIB TEMPLIB
//ENDBCHJOB
```

---

## Retrieving and Printing a History File

The following procedure shows how you can retrieve a history file from one AS/400 system and spool it at another AS/400 system for printing.

**Note:** The user ID under which the requests in the following example run on the AS/400 system must be authorized to YOURLIB/FILEDATA.

1. The host system NetView DM operator initiates a CLIST (batch job) at the **source system**, the system that issues a request to establish communications with another system, and copy the history file into a file. Use the Grant Object Authority (GRTOBJAUT) command to authorize the user ID under which DSNX will run on the AS/400 system. For example, the CLIST could be like the following:

```
//BCHJOB JOB(SPLFGET)
CRTLIB YOURLIB
CRTPF FILE(YOURLIB/FILEDATA) RCDLEN(133) MBR(*NONE)
DSPLOG OUTPUT(*PRINT)
GRTOBJAUT OBJ(YOURLIB/FILEDATA) +
          OBJTYPE(*FILE) +
          USER(user ID) +
          AUT(*ALL)
CPYSPLF FILE(QPDSPLG) TOFILE(YOURLIB/FILEDATA) SPLNBR(*LAST) +
        TOMBR(MEMBER)
//ENDBCHJOB
```

**Note:** The **CPYSPLF** command copies the textual data in the spooled file. It does not copy the advanced function attributes, such as graphics and variable fonts.

2. The host system NetView DM operator retrieves the log file from the source AS/400 system. The GIX name at node is:

```
YOURLIB.FILEDATA.FILE
```

3. The host system NetView DM operator initiates a CLIST to create a library on the target AS/400 system. For example:

```
//BCHJOB JOB(DSNXJOB)
CRTLIB YOURLIB
//ENDBCHJOB
```

4. The host system NetView DM operator adds the log file to the target AS/400 system. The ID of the requesting user sent with the request must be authorized to the YOURLIB created in step 3. The Grant Object Authority (GRTOBJAUT) command can be used to grant authority, or have the job in step 3 run under the ID of the requesting user.

5. The host system NetView DM operator initiates a CLIST to create the spooled file on the target AS/400 system. For example:

```
//BCHJOB JOB(DSNXJOB)
OVRPRTF QSYSVRT CTLCHAR(*FCFC)
CPYF FROMFILE(YOURLIB/FILEDATA) +
     TOFILE(QSYSVRT) FROMMBR(MEMBER)
DLTLIB YOURLIB
//ENDBCHJOB
```

For more information about saving and restoring spooled files, see the *Backup and Recovery* book.

---

## Transferring Program Temporary Fixes (PTFs)

The following procedure shows how you can transfer the IBM-supplied PTFs from one AS/400 system to other AS/400 systems. However, this procedure *cannot* be used to upgrade your system to a new release because Licensed Internal Code is not in a \*SAVF (save file) format.

1. The PTF should already be copied (CPYPTF) to the source system and in a save file (\*SAVF) format.
2. The host system NetView DM operator retrieves the object from the source AS/400 system. Use lib.file.SAVF. for the name at node.
3. The host system NetView DM operator adds the object to the target AS/400 system. Use lib.file.SAVF. for the name at node.
4. A system operator must sign on the target AS/400 system and do the following:
  - Load the PTF (LODPTF command)
  - Apply the PTF (APYPTF command)

**Note:** Licensed Internal Code fixes<sup>2</sup> may be distributed in this same way if received at the source system in \*SAVF format (for example, through electronic customer support).

---

## Transferring Spooled File Entries

The following procedure shows how you can retrieve a spooled file entry from one AS/400 system and add it to the spooled file on another AS/400 system.

**Note:** The user ID under which the requests in the following example run on the AS/400 system must be authorized to YOURLIB/FILEDATA.

1. The host system NetView DM operator initiates a CLIST (batch job) to copy the spooled file to a temporary database member on the source AS/400 system. Use the Grant Object Authority (GRTOBJAUT) command to authorize the user ID under which DSNX will run on the AS/400 system. For example, the CLIST could be like the following:

```
//BCHJOB JOB(DSNXJOB)
CRTLIB YOURLIB
DSPLIB QGPL OUTPUT(*PRINT)
CRTPF FILE(YOURLIB/FILEDATA) RCDLEN(132) MBR(*NONE)
GRTOBJAUT OBJ(YOURLIB/FILEDATA) +
          OBJTYPE(*FILE)          +
          USER(user ID)           +
          AUT(*ALL)
CPYSPLF FILE(QPDSPLIB) TOFILE(YOURLIB/FILEDATA) SPLNBR(*LAST) +
          TOMBR(MEMBER) CTLCHAR(*FCFC)
//ENDBCHJOB
```

---

<sup>2</sup> Licensed Internal Code fixes are temporary solutions to, or methods of bypassing, a defect in a current release of the Licensed Internal Code.

2. The host system NetView DM operator retrieves the file from the source AS/400 system. The GIX name at node is:

```
YOURLIB.FILEDATA.MEMBER.MEM
```

3. The host system NetView DM operator initiates a CLIST to create a library on the target AS/400 system. For example:

```
//BCHJOB JOB(DSNXJOB)
CRTLIB YOURLIB
CRTPF FILE(YOURLIB/FILEDATA) RCDLEN(132) MBR(*NONE)
//ENDBCHJOB
```

4. The host system NetView DM operator adds the log file to the target AS/400 system. The ID of the requesting user sent with the request must be authorized to the YOURLIB created in step 3. The Grant Object Authority (GRTOBJAUT) command can be used to grant authority, or have the job in step 3 run under the ID of the requesting user.

5. The host system NetView DM operator initiates a CLIST to create the spooled file on the target AS/400 system. For example:

```
//BCHJOB JOB(DSNXJOB)
CPYF FROMFILE(YOURLIB/FILEDATA) +
      TOFILE(QSYSPRT) FROMMBR(MEMBER)
DLTLIB YOURLIB
//ENDBCHJOB
```

**Note:** The **CPYSPLF** command copies the textual data in the spooled file. It does not copy the advanced function attributes, such as graphics and variable fonts.

---

## Chapter 6. DSNX Differences

This chapter describes the differences in DSNX support between the AS/400 system and the System/36.

---

### Differences from System/36 DSNX Support

In general, OS/400 DSNX support is an equivalent to System/36 DSNX support. However, in the following areas, the OS/400 DSNX support is significantly different.

- The AS/400 system can be configured as an intermediate connection or as a direct connection type to NetView DM. The AS/400 system supports all functions on either connection type. The System/36 does not support data compression or decompression when configured as a direct connection.
- OS/400 DSNX messages and status presented to the host interactive operator facility (IOF) operator are different from System/36 messages and status. In most cases, NetView DM functions remain in a PEND status until OS/400 DSNX sends a logon to report status.
- The OS/400 Work with DSNX/PC Queues (WRKDPCQ) command allows a user only to display or delete queue entries. The queue management function of System/36 DSNX additionally allows the user to hold or release an individual queue entry or an entire queue.
- OS/400 DSNX performs a logon to the host whenever replies are received from previous requests, if no active session currently exists. This is managed internally by OS/400 DSNX. System/36 DSNX additionally provides access to the logon support to allow a user on System/36 to start the session with the host.
- Except for files, there is no object compatibility between OS/400 DSNX and System/36 DSNX. System/36 files may be retrieved by the host and sent to an AS/400 system, where they are received as database members. OS/400 database members may be retrieved by the host and sent to System/36, where they are received as files. Notice that there are differences in the naming conventions (in terms of qualifiers) between files in System/36 DSNX and database members in OS/400 DSNX. Thus, you must rename the retrieved file or database member at the host before sending it.
- When a file is sent from NetView DM to the System/36 with the Add option specified, System/36 DSNX creates the file and places the data in it. If the Replace option is specified, the file already exists on the System/36, and the file attributes match the file attributes specified in the NetView DM request, then System/36 DSNX replaces all of the data in the file on the System/36. If the file attributes do not match those specified by NetView DM, the existing System/36 file is deleted and a new file is created with the attributes specified by NetView DM.

Before any member can be sent from NetView DM to a file on an AS/400 system, the file must already exist on the AS/400 system. The file attributes specified for the existing AS/400 file will apply to all members that are stored in the file.

When a member is sent from NetView DM to the AS/400 system with the Add option specified, AS/400 DSNX creates the member and stores it in the file.

When a member is sent from NetView DM to the AS/400 system with the Replace option specified and the member already exists in the AS/400 file, AS/400 DSNX replaces all of the data in the existing member. If the Replace option is specified and the member does not exist in the AS/400 file, AS/400 DSNX creates the member.

If the NetView DM file attributes do not match those of the AS/400 file, AS/400 DSNX does not delete and recreate the file with new attributes, because this would destroy other members in the file.

- If the record length of the member sent by NetView DM is smaller than the record length of the AS/400 file, AS/400 DSNX stores the data in the member. DSNX uses the attributes defined for the AS/400 file and pads individual records.
- If the record length of the member sent by NetView DM is larger than the record length of the AS/400 file, DSNX does not store the data and an error message is sent to NetView DM.

---

## Appendix A. NetView DM to OS/400 DSNX Configuration Example

This example provides the information needed to use OS/400 DSNX with NetView DM and the following:

- System/370 host
- 3725 Communications Controller
- AS/400 host-attached node (named HANODE)
- AS/400 node (named DALB60)
- Personal computer (named TU1021)

In this example distributions are routed through HANODE to DALB60 and to the personal computer attached to DALB60. Figure A-1 shows the example network.

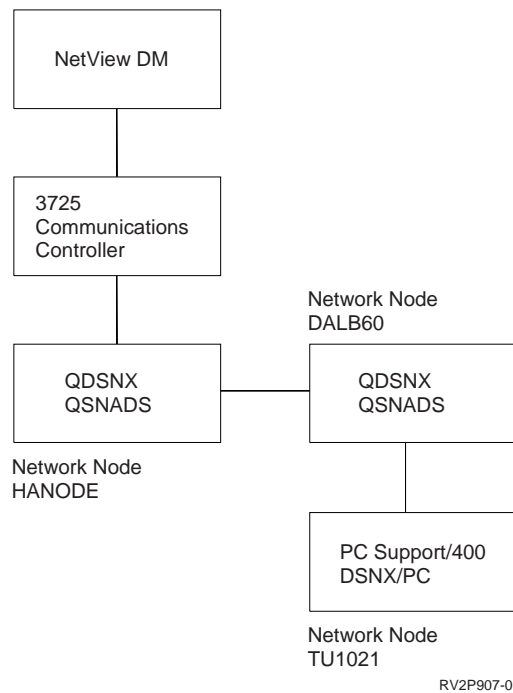


Figure A-1. DSNX Example Network

---

## System/370 Host Considerations

### VTAM Definitions

```
SWAS40  VBUILD TYPE=SWNET
AS40PU  PU  ADDR=C1, X
          DISCNT=NO, X
          IDBLK=056, X
          IDNUM=80040, X
          MAXOUT=7, X
          MAXDATA=265, X
          PASSLIM=7, X
          PUTYPE=2, X
          PACING=(1,1), X
          VPACING=2, X
          ISTATUS=ACTIVE

*****
*****  NetView DM Logical Unit
*****
AS40LU1  LU  LOCADDR=1, X
          ISTATUS=ACTIVE, X
          MODETAB=S36MODT, X
          DLOGMOD=DSX256
```

### Logmode Definition

```
DSX256  MODEENT LOGMODE=DSX256, LOGMODE FOR S/3X X
          FMPROF=X'03', NVDM SUPPORT X
          TSPROF=X'03', X
          PRIPROT=X'B0', X
          SECPROT=X'B0', X
          COMPROT=X'0000', X
          PSNDPAC=X'07', X
          SSNDPAC=X'07', X
          SRCVPAC=X'07', X
          RUSIZES=X'8585' 256 BYTES
```

**Note:** The parameters shown represent those known for an actual working network. However, better performance may be achieved by tuning.



## NetView DM Node Definitions

Since HANODE acts as the intermediate node for DALB60 and the personal computer attached to DALB60, the logical unit name for HANODE must be specified in both DALB60 and TU1021 node definitions. The directory name of the node type PDOS must be the name of the system to which the personal computer is attached (DALB60).

### Node HANODE

Node Name: HANODE  
Node Type: SSP  
Status: Test  
Logon ID: NVDM  
Password: NVDM  
Logon Mode: DSX256  
Connection: Intermediate  
Node Class: A0  
Logical Unit: AS40LU1  
Linetype: Switched

### Node DALB60

Node Name: DALB60  
Node Type: SSP  
Status: Test  
Logon ID: NVDM  
Password: NVDM  
Logon Mode: DSX256  
Connection: Intermediate  
Node Class: A0  
Logical Unit: AS40LU1  
Linetype: Switched

### Node TU1021

Node Name: TU1021  
Node Type: PDOS  
Status: Test  
Logical Unit: AS40LU1  
Logon Mode: DSX256  
Connection: Intermediate  
Directory: DALB60  
Linetype: Switched

---

## Definitions for HANODE Node

HANODE token-ring line definition:

```
CRTLINTRN LIND(TRNLIN) RSRNAME(LIN031) MAXCTL(50) MAXFRAME(1994)
ADPTADR(400001553072) EXCHID(05680040)
TEXT('TOKEN-RING LINE DESCRIPTION')
```

System/370 host controller definition:

```
CRTCTLHOST CTLD(TRLANHOST) LINKTYPE(*LAN) APPN(*NO)
SWTLINLST(TRNLIN) ADPTADR(400000000031)
SSCPID(050000000001) TEXT('HOST CONTROLLER')
```

HANODE DSNX device definition:

```
CRTDEVSNUF DEVD(DSNXDEV) LOCADR(01) RMTLOCNAME(DSNXLOC)
CTL(TRLANHOST) PGMSTRRQS(*YES) DFTPGM(QDXHRTR)
RCDLRN(32761) BLKLEN(32761) TEXT('DSNX DEVICE')
```

DALB60 APPN controller definition:

```
CRTCTLAPPC CTLD(DALB60) LINKTYPE(*LAN) APPN(*YES)
SWTLINLST(TRNLINE) RMTNETID(DALNET) RMTCPNAME(DALB60)
EXCHID(05600002) ADPTADR(400001131021) NODETYPE(*NETNODE)
TEXT('DALB60 APPN CONTROLLER')
```

NetView DM user profile:

```
CRTUSRPRF USRPRF(NVDM) PASSWORD(NVDM)
```

QDSNX subsystem communications entry:

```
ADDCMNE SBSDB(QGPL/QDSNX) DEV(DSNXDEV) JOBD(QGPL/QDSNX) DFTUSR(QDSNX)
```

QDSNX subsystem routing entry:

```
ADDRTE SBSDB(QGPL/QDSNX) SEQNBR(20) CMPVAL('PGMEVOKE' 29) PGM(*RTGDTA)
CLS(QGPL/QDSNX)
```

HANODE directory entries:

```
ADDIRE USRID(QDSNX HANODE) USRD('QDSNX on HANODE') USER(QDSNX)
SYSNAME(HANODE)
```

```
ADDIRE USRID(QDSNX DALB60) USRD('QDSNX on DALB60') USER(*NONE)
SYSNAME(DALB60)
```

```
ADDIRE USRID(TU1021 DALB60) USRD('TU1021 on DALB60') USER(*NONE)
SYSNAME(*PC)
```

To configure HANODE SNADS distribution queues, use the Configure Distribution Services (CFGDSTSRV) command. Select 1=Distribution Queues to configure a distribution services queue on HANODE for each remote system that receives distributions (DALB60 and TU1021).

```
Queue Name: DALB60
Queue Type: *SNADS
Remote Location Name: DALB60
Mode Name: DALGRP
Remote Network ID: DALNET
Local Location Name: *LOC
```

```
Queue Name: TU1021
Queue Type: *SNADS
Remote Location Name: TU1021
Mode Name: DALGRP
Remote Network ID: DALNET
Local Location Name: *LOC
```

#### HANODE SNADS routing table:

```
System Name:          DALB60
Service Level:
Fast:
  Queue Name:         DALB60
  Max Hops:           *DFT
Status:
  Queue Name:         DALB60
  Max Hops:           *DFT
Data High:
  Queue Name:         DALB60
  Max Hops:           *DFT
Data Low:
  Queue Name:         DALB60
  Max Hops:           *DFT
```

---

## Definitions for DALB60 Node

#### DALB60 token-ring line definition:

```
CRTLINTRN  LIND(TRNLIN)  RSRNAME(LIN021)  MAXCTL(50)  MAXFRAME(1994)
           ADPTADR(400001131021)  EXCHID(05600002)
           TEXT('TOKEN-RING LINE DESCRIPTION')
```

#### HANODE APPN controller definition:

```
CRTCTLAPPC  CTLD(HANODE)  LINKTYPE(*LAN)  APPN(*YES)
           SWTLINLST(TRNLIN)  RMTNETID(DALNET)  RMTCPNAME(HANODE)
           EXCHID(05680040)  ADPTADR(400001553072)  NODETYPE(*NETNODE)
           TEXT('HANODE APPN CONTROLLER')
```

#### Client Access controller definition:

```
CRTCTLAPPC  CTLD(TU1021)  LINKTYPE(*LAN)  APPN(*YES)
           SWTLINLST(TRNLIN)  RMTNETID(DALNET)  RMTCPNAME(TU1021)
           ADPTADR(400001131021)  NODETYPE(*ENDNODE)  CPSSN(*NO)
           TEXT('TU1021 PCS CONTROLLER')
```

#### NetView DM user profile:

```
CRTUSRPRF  USRPRF(NVDM)  PASSWORD(NVDM)
```

#### DALB60 directory entries:

```
ADDIRE  USRID(TU1021 DALB60)  USRD('TU1021 on DALB60')  USER(*NONE)
        SYSNAME(*PC)

ADDIRE  USRID(QDSNX DALB60)  USRD('QDSNX on DALB60')  USER(QDSNX)
        SYSNAME(DALB60)

ADDIRE  USRID(QDSNX HANODE)  USRD('QDSNX on HANODE')  USER(*NONE)
        SYSNAME(HANODE)
```

**Note:** You must add a directory entry for each personal computer attached to DALB60.

DALB60 SNADS distribution queues:

Queue Name: HANODE  
Queue Type: \*SNADS  
Remote Location Name: HANODE  
Mode Name: DALGRP  
Remote Network ID: DALNET  
Local Location Name: \*LOC

DALB60 SNADS routing table:

System Name: HANODE  
Service Level:  
Fast:  
Queue Name: HANODE  
Max Hops: \*DFT  
Status:  
Queue Name: HANODE  
Max Hops: \*DFT  
Data High:  
Queue Name: HANODE  
Max Hops: \*DFT  
Data Low:  
Queue Name: HANODE  
Max Hops: \*DFT

### Subsystem Considerations

In addition to the subsystems that are normally active, both the QDSNX and QSNADS subsystems must be active. Before the personal computer can send or receive distributions from NetView DM, both Client Access and DSNX/PC must be active.

---

## TU1021 Definitions

Client Access CONFIG.PCS file:

```
RTYP ITRN
MDEF DALB60,2
RTLN DALNET.TU1021
TRLI DALB60,400001131021
```

DSNX/PC transmission parameter table:

1. Routing Element Name .....TU1021
2. Remote Logical Unit Name (RLUN).....DALB60

**Note:** This table contains the SNADS parameters used to communicate with NetView DM through an intermediate node. This table must show the name of the PC node and the node to which it is attached.

---

## Appendix B. DSNX Problem Analysis

To describe DSNX logging and problem analysis, DSNX processing can be divided into four areas:

- DSNX host interface
- DSNX request processor
- DSNX/Client Access
- Local DSNX/PC queue management

These areas interact with each other as shown in Figure B-1 on page B-2, Figure B-2 on page B-3, and Figure B-3 on page B-4.

All four areas make entries in the DSNX log (QUSRSYS/QDSNX), both for successful completion of functions and error conditions. If the logging operation itself fails and an area cannot make an entry, then the affected area sends a CPI8854 message to the QSYSOPR message queue and ends its processing. Log entries showing a successful completion of a function are not made until after the function is completed. Thus, in the event of a logging failure after a successfully completed function, there may not be an entry in the log for the function just completed.

For information on how to display and interpret log entries, see the topic “DSNX Journal Analysis” on page B-6.

In each **journal entry**<sup>1</sup>, a hexadecimal value is used to indicate the function or error that has occurred. These values are shown following the explanation of each area. The values for normal entries are shown in parentheses ( ) and the values for error entries are shown in square brackets [ ]. The first 125 bytes of the journal entry are formatted. The entry-specific data starts at the program name, or byte 126 of the journal entry. The hexadecimal value used to indicate the function is in byte 35 of the entry-specific data.

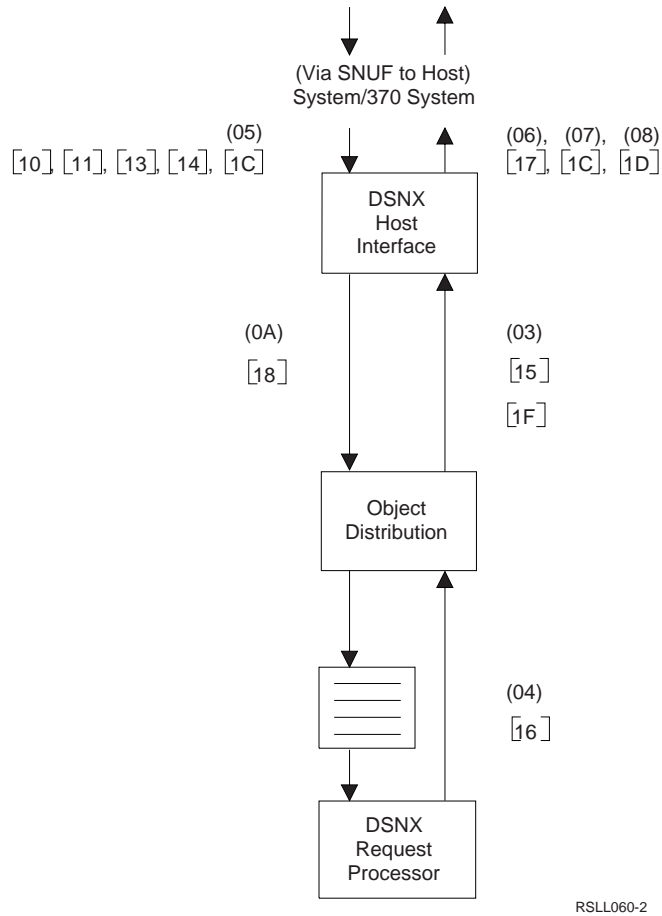
See Appendix D, “NetView DM to DSNX Data Flow” for examples of data flow between NetView DM and DSNX with corresponding journal entries.

---

<sup>1</sup> A record in the journal receiver that contains information about database files.

## DSNX Logging at Local Request Processor

Figure B-1 shows some of the DSNX entries made during an operation with a local request processor.



RSLL060-2

Figure B-1. DSNX Logging with Local Request Processor

## DSNX Logging at Remote Request Processor

Figure B-2 shows some of the DSNX entries made during an operation with a remote request processor.

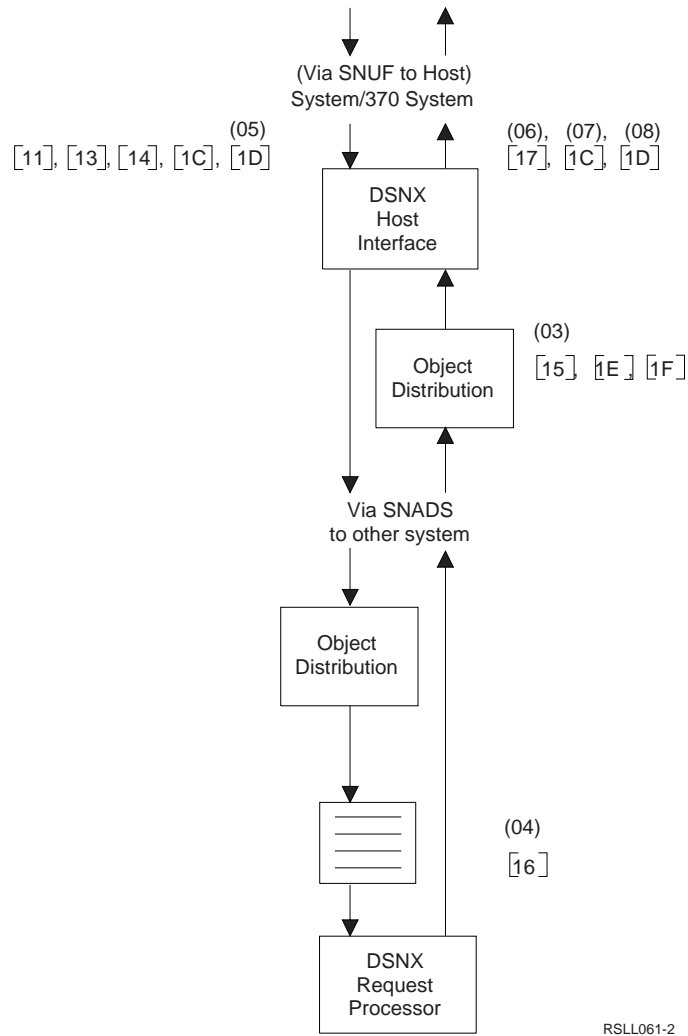


Figure B-2. DSNX Logging with Remote Request Processor

## DSNX Logging with DSNX/Client Access

Figure B-3 shows some of the DSNX entries made during an operation with DSNX/Client Access and DSNX/PC queue management.

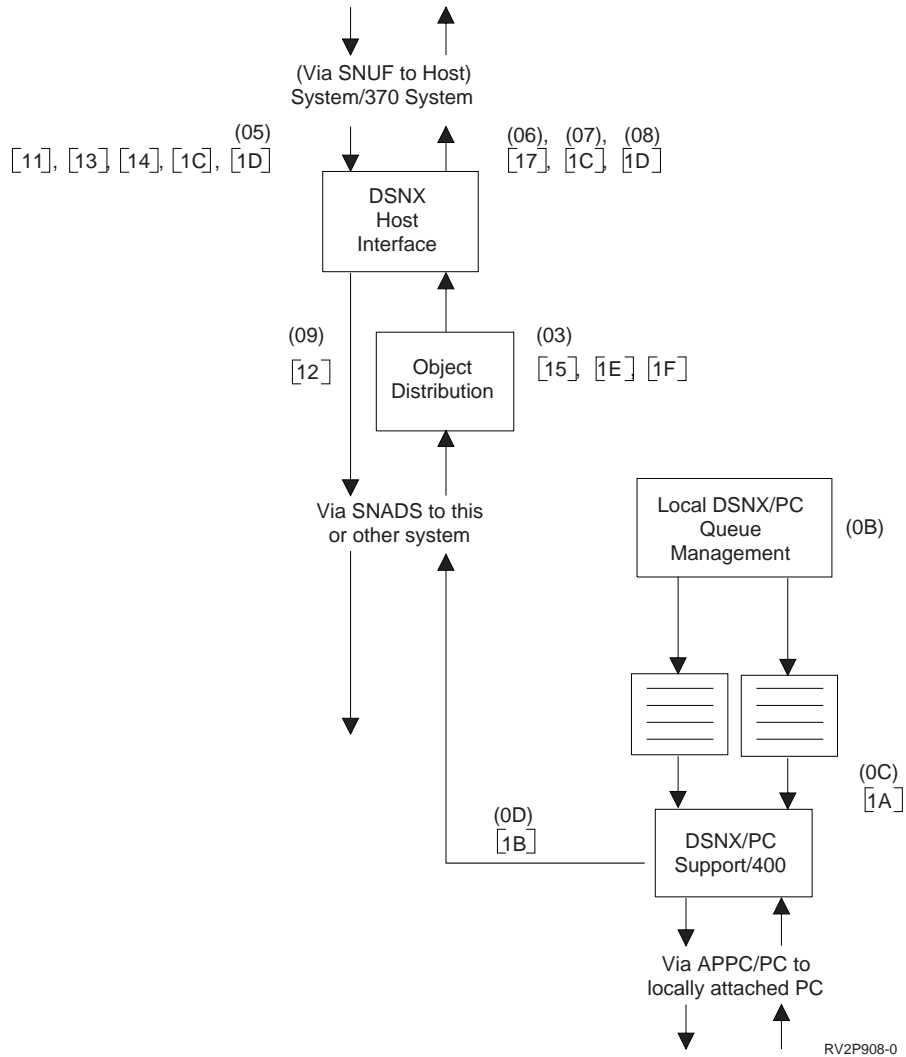


Figure B-3. DSNX Logging with DSNX/Client Access



---

## Area 1: DSNX Host Interface

This area contains the support to handle receipt and forwarding of NetView DM requests and replies between systems. Requests are received from the NetView DM host. Each request received is examined to determine if it is directed to this node or to another node in the network. If the request is not directed to this node, it is forwarded via SNADS to the next node in the direction of its ultimate destination.

When a request is found to be directed to this node, it is routed to the DSNX request processor or to DSNX/Client Access. If the request affects objects or starts a task on this system, it is routed to the DSNX request processor (Area 2) via an internal DSNX queue. If the request is directed to a locally attached personal computer, SNADS is used to place the distribution on an internal DSNX/PC queue to await processing by DSNX/Client Access (Area 3).

Replies flow in the opposite direction. If this system is a host interface node, the replies are correlated with the original request and held until the NetView DM host asks for them.

As functions are successfully performed, log entries (XL) are made to indicate this:

(05)	Request received
(03)	Reply received
(06)(07)	Reply returned to NetView DM host
(08)	Resynchronization
(09)	Distribute request to remote system
(0A)	Distribute request to local request processor

When errors occur during processing, error entries (XE) are made as follows, and a message is sent to the job log.

[1C] [1D] [11]	Error detected by host interface
[13] [17]	Error detected by host interface
[12] [18]	Distribution error
[15]	Error detected in response
[14]	Host interface initialization error
[19]	Resynchronization error
[1E]	SNADS error previously detected
[1F]	Outstanding request not found

---

## Area 2: DSNX Request Processor

The DSNX request processor handles requests affecting objects on this system, or tasks to be run on this system. A reply to each request is routed back to the communications interface (Area 1) indicating either successful completion or failure.

If a function has been successfully completed, this is indicated by making a log entry of (04). If a function cannot be performed, an error entry of [16] is made instead, and a message is sent to the QDSNX job log describing the specific error.

---

### Area 3: DSNX/Client Access

The DSNX/Client Access is started by a locally attached personal computer, which asks for any NetView DM requests that affect that particular personal computer. DSNX/Client Access has access to a set of internal DSNX/PC queues (one for each locally attached personal computer). It examines the queue for the particular personal computer and, if any requests are on the queue, sends the requests to the personal computer to be handled. Replies are received from the personal computer for each request, and are forwarded back in the direction of the NetView DM host.

DSNX/Client Access makes a log entry of (0C) when a request is sent successfully to a personal computer, and an entry of (0D) when a reply is received. An error log entry of [1A] is made if an error occurs while sending, and an entry of [1B] if an error occurs while receiving. Also, when an error occurs, a message is sent to the job log of the job under which this area is running.

---

### Area 4: Local DSNX/PC Queue Management

DSNX provides a user interface for management of the internal DSNX/PC queues used by DSNX/Client Access (Area 3). Using the WRKDPCQ command, a user may display the contents of DSNX/PC queues and also delete requests from those queues.

The only log entry made by this area is (0B), which indicates successful completion of a delete operation from a DSNX/PC queue. A CPI8857 message is also sent to the job log of the job under which the WRKDPCQ command was issued.

No error entries are made by this area.

---

## DSNX Journal Analysis

The Display Journal (DSPJRN) command displays the DSNX journal entries. These journal entries provide you with a record of completed and outstanding NetView DM activity on each destination DSNX node. The journal entries also show a record of the NetView DM requests received and sent to a NetView DM host node on the DSNX node that communicates directly with the NetView DM host.

The following examples show the format of the journal displays. Following the display examples are tables with descriptions of the journal entries for each type of entry. All journal entry fields before the name of job on these displays are used to create the header information on the Journal Entry display.

There are two types of DSNX journal entries:

XL            Normal entries, such as receiving a NetView DM request from a NetView DM host.

XE The DSNX request was not completed because of either a system or an administrative error, such as an incorrect user ID or password.

Other entry types may also appear in the DSNX journal. You may ignore them for purposes of DSNX problem analysis.

The following is an example of the journal display shown by entering the Display Journal (DSPJRN) command:

DSPJRN QDSNX

```

                                Display Journal Entries
Journal . . . . . : QDSNX          Library . . . . . : QUSRSYS

Type options, press Enter.
  5=Display entire entry

Opt Sequence  Code  Type  Object      Library      Job           Time
-----
  5   39   S   XL          DSNX03       DSNX03       14:51:22
  5   40   S   XL          DSNX03       DSNX03       14:51:24
  -   41   S   XL          QDSNX        DSNX03       14:51:24
  -   42   S   XL          DSNX03       DSNX03       14:51:25
  -   43   S   XL          DSNX03       DSNX03       14:51:26
  -   44   S   XL          QDSNX        DSNX03       15:25:47
  -   45   S   XL          DSNX04       DSNX04       15:29:08
  -   46   S   XL          QDSNX        DSNX04       15:29:17
  -   47   S   XL          DSNX04       DSNX04       15:29:19
  -   48   S   XL          DSNX04       DSNX04       15:29:23
  -   49   S   XL          QDSNX        DSNX04       15:29:24
  -   50   S   XL          DSNX03       DSNX03       15:43:42   +

F3=Exit   F12=Cancel

```

The following sample journal entry was shown by entering a 5 in the *Opt* column before 40 (the second item in the *Sequence* column) and pressing the Enter key. Information from all fields through the Library column is used on the next display.

```

                                Display Journal Entry

Journal . . . . . : QDSNX          Library:  QUSRSYS
Sequence . . . . . : 40

Code . . . . . : S - SNA services entry
Type . . . . . : XL - DSNX logging information

Object . . . . . :
Member . . . . . :

Position to . . . . . _____ (Column)

Entry specific data
Column *...+...1...+...2...+...3...+...4...+...5
00001 'QDXHRCV DSNX03 QDSNX 018406 0SHORTTSTPHASE1'
00051 ' 900227 1442 DSXNDM j MEM CTSMEM CT'
00101 'SFIL CTLIB DEPT555 QDSNX RCHAS123 '

More...

Press Enter to continue.

F3=Exit F6=Display only entry specific data
F10=Display only entry details F12=Cancel F24=More keys

```

The Display Journal Entry display usually does not have enough space to display all of the journal entry. Press F6 to show all of the entry on the Display Entry Specific Data display. The entry specific data is all of the data about a DSNX log entry except the header information, which is shown on the top half of the display.

```

                                Display Entry Specific Data

Journal . . . . . : QDSNX          Library:  QUSRSYS
Sequence . . . . . : 40

Position to . . . . . _____ (Column)

Entry specific data
Column  *...+....1....+....2....+....3....+....4....+....5
00001  'QDXHRCV DSNX03   QDSNX   018406 0SHORTTSTPHASE1'
00051  ' 900227 1442     DSXNDM   j MEM   CTSMEM CT'
00101  'SFIL CTLIB      DEPT555 QDSNX   RCHAS123  '
00151  '

```

Bottom

Press Enter to continue.

F3=Exit F6=Display entry F11=Display hexadecimal format F12=Cancel  
F14=Display previous entry F15=Display only entry details

You can also display the entry-specific data in hexadecimal format by pressing F11 (Hexadecimal format). The entry specific data changes to hexadecimal format as shown on the following display. To return to the character data, press F11 again.

```

                                Display Entry Specific Data

Journal . . . . . : QDSNX          Library:  QUSRSYS
Sequence . . . . . : 40

Position to . . . . . _____ (Column)

Entry specific data
Column  *...+....1....+....2....+
00001  'D8C4E7C8D9C3E540C4E2D5E7F0F340404040D8C4E2D5E74040'
00026  '404040F0F1F8F4F0F605F0E2C8D6D9E3E3E2E3D7C8C1E2C5F1'
00051  '4040F9F0F0F2F2F74040F1F4F4F2000100010001C4E2E7D5C4'
00076  'D4404003039104D4C5D44040404040C3E3E2D4C5D44040C3E3'
00101  'E2C6C9D34040C3E3D3C9C240404040404040404040C4C5D7'
00126  'E3F5F5F540D8C4E2D5E7404040D9C3C8C1E2F1F2F300000000'
00151  '000000000000000000000000000000000000000000000000000'
00176  '00000000'

```

Bottom

Press Enter to continue.

F3=Exit F6=Display entry F11=Display character format F12=Cancel  
F14=Display previous entry F15=Display only entry details

## DSNX Journal Formats from DSPJRN Command

The following sections define the formats of the information generated for each of the DSNX journal entry types by the Display Journal (DSPJRN) command.

### Format for Normal DSNX Log Entries (XL)

This table defines and explains the fields of a DSNX log entry, which represents completed DSNX functions.

**Note:** The following is the database description of the record QDXLGLOG (only description, no data) for the journal log entry type XL, which represents the distribution log entry. This record is contained in physical file QADXJRNL, which is shipped as part of QDSNX and could be used by a programmer to create a utility that formats log entries. Add 125 to the log entry to set the offset into QDXLGLOG.

Offset	Display Column	Field	Format	Description																								
0		Length of Entry	Zoned(5,0)	Total length of the journal entry including the entry length field.																								
5		Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.																								
15		Journal Code	Char(1)	Always S.																								
16		Entry Type	Char(2)	Always XL for DSNX-logged event.																								
18		Date of Entry	Char(6)	The system date that the entry was made.																								
24		Time of Entry	Zoned(6,0)	The system time that the entry was made.																								
30		(Reserved Area)	Char(95)																									
125	1	Program Name	Char(8)	The name of the DSNX program that made the journal entry.																								
133	9	Name of Job	Char(10)	The name of the job that caused the entry to be generated.																								
143	19	User Name	Char(10)	The user profile name associated with the job.																								
153	29	Job Number	Zoned(6,0)	The job number.																								
159	35	Function	Char(1)	DSNX function that was being performed when the logged entry was made. The possible values are:  <table border="0"> <thead> <tr> <th>Hex</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>03</td> <td>A DSNX reply distribution was received at the host interface.</td> </tr> <tr> <td>04</td> <td>Request processor ran the NetView DM request.</td> </tr> <tr> <td>05</td> <td>Value for host interface receiver entry when a NetView DM request header is received.</td> </tr> <tr> <td>06</td> <td>Value for host interface query response when a delayed ACK response is completed to the NetView DM host.</td> </tr> <tr> <td>07</td> <td>Value for host interface response when a data set ready response is completed to the NetView DM host.</td> </tr> <tr> <td>08</td> <td>Value for host interface response when a NetView DM resynchronization is completed.</td> </tr> <tr> <td>09</td> <td>Value for host interface remote distribution.</td> </tr> <tr> <td>0A</td> <td>Value for host interface local distribution.</td> </tr> <tr> <td>0B</td> <td>Entry deleted from DSNX/PC queue via WRKDPCQ command.</td> </tr> <tr> <td>0C</td> <td>Value for DSNX/Client Access in send mode.</td> </tr> <tr> <td>0D</td> <td>Value for DSNX/Client Access in receive mode.</td> </tr> </tbody> </table>	Hex	Function	03	A DSNX reply distribution was received at the host interface.	04	Request processor ran the NetView DM request.	05	Value for host interface receiver entry when a NetView DM request header is received.	06	Value for host interface query response when a delayed ACK response is completed to the NetView DM host.	07	Value for host interface response when a data set ready response is completed to the NetView DM host.	08	Value for host interface response when a NetView DM resynchronization is completed.	09	Value for host interface remote distribution.	0A	Value for host interface local distribution.	0B	Entry deleted from DSNX/PC queue via WRKDPCQ command.	0C	Value for DSNX/Client Access in send mode.	0D	Value for DSNX/Client Access in receive mode.
Hex	Function																											
03	A DSNX reply distribution was received at the host interface.																											
04	Request processor ran the NetView DM request.																											
05	Value for host interface receiver entry when a NetView DM request header is received.																											
06	Value for host interface query response when a delayed ACK response is completed to the NetView DM host.																											
07	Value for host interface response when a data set ready response is completed to the NetView DM host.																											
08	Value for host interface response when a NetView DM resynchronization is completed.																											
09	Value for host interface remote distribution.																											
0A	Value for host interface local distribution.																											
0B	Entry deleted from DSNX/PC queue via WRKDPCQ command.																											
0C	Value for DSNX/Client Access in send mode.																											
0D	Value for DSNX/Client Access in receive mode.																											
160	36	Correlation ID <sup>1</sup>	Char(44)	Identifier of the logged DSNX distribution.																								
204	80	Logged Data <sup>2</sup>	Char(100)																									

<sup>1</sup> For a list of the correlation ID values, see "Format for Correlation ID Entries" on page B-11.

<sup>2</sup> For a list of the logged data entries, see "Format for Logged Data Entries" on page B-11.

## Format for Distribution Errors (XE)

This table defines and explains the fields of a log entry for error information received.

**Note:** The following is the database description of the record QDXERLOG (only description, no data) for the journal log entry type XE, which represents error information log entry. This record is contained in physical file QADXERLG, which is shipped as part of QDSNX and could be used by a programmer to create a utility for log entries.

Offset	Display Column	Field	Format	Description																														
0		Length of Entry	Zoned(5,0)	Total length of the journal entry including the entry length field.																														
5		Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.																														
15		Journal Code	Char(1)	Always S.																														
16		Entry Type	Char(2)	Always XE for DSNX-logged errors.																														
18		Date of Entry	Char(6)	The system date that the entry was made.																														
24		Time of Entry	Zoned(6,0)	The system time that the entry was made.																														
30		(Reserved Area)	Char(95)																															
125	1	Program Name	Char(8)	The name of the DSNX program that made the journal entry.																														
133	9	Name of Job	Char(10)	The name of the job that caused the entry to be generated.																														
143	19	User Name	Char(10)	The user profile name associated with the job.																														
153	29	Job Number	Zoned(6,0)	The job number.																														
159	35	Function	Char(1)	DSNX function that was being performed when the logged entry was made. The possible values are:  <table border="0"> <thead> <tr> <th>Hex</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>11</td> <td>Host interface receive.</td> </tr> <tr> <td>12</td> <td>Object distribution error, host interface distribute through SNADS to destination node.</td> </tr> <tr> <td>13</td> <td>NetView DM protocol error, host interface receive.</td> </tr> <tr> <td>14</td> <td>Host interface initialization or router errors.</td> </tr> <tr> <td>15</td> <td>Response indicates SNADS error detected.</td> </tr> <tr> <td>16</td> <td>Request processor could not run NetView DM function.</td> </tr> <tr> <td>17</td> <td>Host interface send.</td> </tr> <tr> <td>18</td> <td>Object distribution error for local distribution.</td> </tr> <tr> <td>1A</td> <td>DSNX/Client Access in send mode.</td> </tr> <tr> <td>1B</td> <td>DSNX/Client Access in receive mode.</td> </tr> <tr> <td>1C</td> <td>Unexpected condition detected in host interface.</td> </tr> <tr> <td>1D</td> <td>Unexpected SNUF major/minor return code detected in host interface.</td> </tr> <tr> <td>1E</td> <td>Object distribution detected a SNADS-generated error.</td> </tr> <tr> <td>1F</td> <td>Object distribution error detected by DSNX.</td> </tr> </tbody> </table>	Hex	Function	11	Host interface receive.	12	Object distribution error, host interface distribute through SNADS to destination node.	13	NetView DM protocol error, host interface receive.	14	Host interface initialization or router errors.	15	Response indicates SNADS error detected.	16	Request processor could not run NetView DM function.	17	Host interface send.	18	Object distribution error for local distribution.	1A	DSNX/Client Access in send mode.	1B	DSNX/Client Access in receive mode.	1C	Unexpected condition detected in host interface.	1D	Unexpected SNUF major/minor return code detected in host interface.	1E	Object distribution detected a SNADS-generated error.	1F	Object distribution error detected by DSNX.
Hex	Function																																	
11	Host interface receive.																																	
12	Object distribution error, host interface distribute through SNADS to destination node.																																	
13	NetView DM protocol error, host interface receive.																																	
14	Host interface initialization or router errors.																																	
15	Response indicates SNADS error detected.																																	
16	Request processor could not run NetView DM function.																																	
17	Host interface send.																																	
18	Object distribution error for local distribution.																																	
1A	DSNX/Client Access in send mode.																																	
1B	DSNX/Client Access in receive mode.																																	
1C	Unexpected condition detected in host interface.																																	
1D	Unexpected SNUF major/minor return code detected in host interface.																																	
1E	Object distribution detected a SNADS-generated error.																																	
1F	Object distribution error detected by DSNX.																																	
160	36	Correlation ID <sup>1</sup>	Char(44)	Identifier of the logged DSNX distribution.																														
204	80	Exception Data <sup>2</sup>	Char(297)																															

<sup>1</sup> For a list of the correlation ID values, see "Format for Correlation ID Entries" on page B-11.

<sup>2</sup> For a list of the exception data entries, see "Format for Exception Data Entries" on page B-12.

## Format for Correlation ID Entries

The Correlation ID can contain the following:

Field	Display Column	Format	Description
Reply/Request	36	Char(1)	Reply/request constant = 0 or 1 (0=request, 1=reply).
NetView DM Request Correlator Values	37	Char(34)	
Session ID	37	Char(16)	
Session Date	53	Char(8)	
Session Time	61	Char(4)	
Session Sequence Number	65	Bin(15)	
Session Function Number	67	Bin(15)	
Function/Resource Number	69	Bin(15)	
DSNX Location Name	71	Char(8)	
DSNX LU Number	79	Bin(8)	

## Format for Logged Data Entries

The following table contains the logged data for the specified hexadecimal value.

When Function Value Is	Display Column	Format	Contents
01, 02, 0C, 0D			No specific logged data.
03, 05, 06, 07, 08, 0A, 0B	80	Char(2)	NetView DM request type (hexadecimal value).
	82	Char(1)	Number of qualifiers in resource name (1-4). This may be blank for end of distribution header or a Query request.
	83	Char(40)	Resource name for NetView DM request. This may be blank for end of distribution header or a Query request.
	123	Char(8)	User ID of requester (not used for values 06 and 07). This may be blank for end of distribution header or a Query request.
	131	Char(8)	Destination of requester. For personal computer receivers, this is the REN or system name. For other receivers, this is 'QDSNX'. This may be blank for a Query request type.
	139	Char(8)	Destination qualifier of requester. For personal computer receivers, this is the directory name. For other receivers, this is the REN or system name of the DSNX target system. This will be blank for a Query request type.
04	80	Char(11)	Reserved (blanks).
	91	Char(40)	Resource name for request.
	131	Char(2)	Reserved (blanks).
	133	Char(8)	User ID of requestor.
09	80	Char(2)	NetView DM request type (hexadecimal value).
	82	Char(1)	Number of qualifiers in resource name (1-4).
	83	Char(40)	Resource name for NetView DM request.
	123	Char(8)	User ID of requestor.
	131	Char(8)	Destination of requester. For personal computer receivers, this is the REN or system name. For other receivers, this is 'QDSNX'.
	139	Char(8)	Destination qualifier of requester. For personal computer receivers, this is the directory name. For other receivers, this is the REN or system name of the DSNX target system.
	147	Bin(15)	Number of destinations in this distribution.
149	Char(8)	Internal SNADS origination date.	
157	Char(4)	SNADS sequence number.	

## Format for Exception Data Entries

The following table contains the exception data for the specified hexadecimal value.

When Value Is	Display Column	Format	Contents
14	80	Char(7)	Exception message ID received. This may be blank.
	87	Char(7)	Exception message ID sent.
	94	Bin(15)	Router phase.
11, 17	80	Char(7)	Exception message ID.
	87	Char(8)	Sense data when negative response received.
	95	Char(4)	Return code.
12	80	Char(2)	NetView DM request type (hexadecimal value).
	82	Char(1)	Number of qualifiers in fully qualified resource name.
	83	Char(40)	Resource name.
	123	Char(8)	User ID of requestor. This may be blank for the end of distribution header.
	131	Bin(15)	Distribute data return code.
	133	Bin(15)	Number of destinations in this distribution.
	135	Char(12)	SNADS request identifier.
13	80	Char(2)	NetView DM request type (hexadecimal value).
	82	Char(1)	Number of qualifiers in fully qualified resource name.
	83	Char(40)	Resource name.
	123	Char(8)	User ID of requestor.
	131	Char(7)	DSNX message ID.
	138	Char(1)	Reserved.
	139	Bin(15)	Message subcode.
	141	Char(16)	Destination user ID and system name.
15	80	Char(16)	Detecting SNADS system name.
	96	Char(2)	SNADS error status code.
	98	Char(16)	DSNX receiver user ID and system name.
16	80	Char(4)	Reserved (blanks).
	84	Char(7)	DSNX message ID.
	91	Char(40)	Resource name.
	131	Char(2)	Number of jobs not submitted (for starting jobs only).
	133	Char(8)	User ID of requestor.
18	80	Char(7)	DSNX message ID.
	87	Char(2)	NetView DM request type (hexadecimal value).
	89	Char(1)	Number of qualifiers in fully qualified resource name.
	90	Char(40)	Resource name.
	130	Char(8)	NetView DM user ID.
	138	Char(16)	Destination node identifier.
	154	Char(1)	Local distribution error or invalid receiver.
	155	Char(2)	ODIST return code. Possible values are:
			0101 Delivered
			0201 System error
		0301 Recipient not valid, local	
		0401 Document not valid	
		0501 Canceled	
		0502 Document deleted	
		0601 Maximum list exceeded	
		FFFF Exception condition	
1A, 1B	80	Char(7)	DSNX message ID.
	87	Char(2)	NetView DM request type (hexadecimal value).
	89	Char(1)	Number of qualifiers in fully qualified resource name.
	90	Char(40)	Resource name.
	130	Char(8)	PC node ID.
1C	80	Char(7)	DSNX message ID.
	87	Char(1)	Internal program code.
	88	Char(2)	Internal program error code.
	90	Bin(31)	Number of bytes of data sent and received



When Value Is	Display Column	Format	Contents
1D	80	Char(1)	Internal program code.
	81	Char(2)	Internal program error code.
	83	Char(4)	Unexpected SNUF major/minor code. See <i>SNA Upline Facility Programming</i> .
	87	Bin(31)	Number of bytes of data sent and received
1E	80	Char(7)	Exception message ID.
	87	Char(2)	SNADS error status code.
1F	80	Char(7)	Exception message ID.

## NetView DM Request and Response Type Values

The following table contains the list of request and response type values as contained in the NetView DM request type of the journal entry.

Hexadecimal Value	Description
0281	Add new object
0282	Replace object
0283	Delete
0284	Run CLIST
0285	Inform system operator
0286	Query for status
0391	Retrieve object
0491	Retrieve object response
4282	Decompress object
4286	Query response
4391	Compress dataset
8281	Recover and add new object
8282	Recover and replace object
8391	Recover and retrieve object
8491	Recover and retrieve object response
C282	Recover and decompress object

## Deleting Entries from the QDSNX Journal

You are responsible for deleting entries from the QDSNX journal. Do this when the journal receiver is full, or approximately every 30 days, with the Change Journal (CHGJRN) command. Never delete a journal receiver until it is at least 30 days old. When you delete it, do not delete the new receiver that is created.

Although the size of the journal receiver is limited only by the system capacity and the maximum size of files, a message is sent to the system operator when the journal receiver exceeds 10 megabytes. Even though this limit is reached, journaling to the receiver can continue until the file limits are reached.

A **journal receiver** is a system object that contains journal entries recorded when changes are made to the data in database files or the access paths associated with the database files.

For more information on journaling and changing journal receivers, see the journal management chapter in the *Backup and Recovery* book.

---

## Appendix C. DSNX Request Descriptions

The following table lists the functions that are supported by NetView DM. The object types listed are used in the NetView DM plan to indicate the AS/400 object type.

---

OS/400 Object	NetView DM Type <sup>1</sup>	Send	Retrieve	Delete	Initiate CLIST	Inform Operator
File Mbr (MEM) (sent in data file)	Data set or CLIST	X	X	X	X	
Message (sent as message)	Message					X
Save File (SAVF) (sent in data file)	Program	X	X	X		
Any other object valid for the OBJTYPE parameter of the SAVOBJ command (such as *FILE, *CMD, *PGM)	Program	X	X	X		

---

<sup>1</sup> Even though the NetView DM type given in the table is Program, you can also use Data set.

---

**Note:** When specifying the GIX *name at node*, the asterisk (\*) is not included.



---

## Appendix D. NetView DM to DSNX Data Flow

Table D-1 through Table D-18 represent how data is sent from NetView Distribution Manager (NetView DM) to distributed systems node executive (DSNX) on an AS/400 system. The intent of these diagrams is not to show all data flow, but to show only the information needed for problem analysis and configuration decisions.

The diagrams do not show data flows for levels of the communications functions below DSNX. SNA upline facility (SNUF) level information such as Bind, Start Data Traffic, Notify, and Unbind can be obtained from the *SNA Formats* book.

Each of the flow tables contain the following information:

<b>Table column heading</b>	<b>Meaning</b>
NetView DM Phase Status	This status is shown on the NetView DM Interactive Operator Facility (IOF) and changes as the phase progresses. The <i>NetView Distribution Manager User's</i> book has descriptions of the phases.
NetView DM Description	This is a high level description of the data that NetView DM is sending to DSNX.
Data and Direction	This is some of the data sent between NetView DM and DSNX. The arrow shows the direction of that data flow. Only the first 2 bytes of the request or response is given. These bytes represent the function type of the request or response.  See "NetView DM Request and Response Type Values" on page B-13 for a complete list of the request and response type values.
DSNX Description	This is a high level description of what DSNX is processing or sending to NetView DM.
DSNX Journal Entry	This is a letter code that points to a description of the AS/400 system journal entry made by DSNX. The meaning of the letter codes are shown in the table that follows each function table. For more information about journal entries, see Appendix B, "DSNX Problem Analysis."

Sample data flows are given for:

- "Intermediate Node Successful Add Function" on page D-2
- "Intermediate Node Successful Retrieve with Compress Function" on page D-4
- "Intermediate Node Unsuccessful Delete Function" on page D-6
- "Intermediate Node—Unsuccessful Replace Function" on page D-7
- "Direct Node Successful Replace Function" on page D-8
- "Direct Node Successful Retrieve with Compress Function" on page D-9
- "Direct Node Successful Replace with Decompress Function" on page D-11
- "Direct Node Successful Retrieve CLIST Function" on page D-12
- "Direct Node Unsuccessful Initiate Job Function" on page D-13

---

## Intermediate Node Successful Add Function

Table D-1 on page D-3 shows some of the data flow for an **Add** function. A typical occurrence for an **Add** request is that the data is not available from the request processor when the first query comes from NetView DM. This causes the session to be dropped and the AS/400 system must do a logon operation with the host system to inform NetView DM that a response is available.

A **Replace** request is the same except that the FIC with the request header would have a request code of ('0282') and the OIC that indicates the end of distribution is ('0282').

A **Decompress** request is the same except that the FIC with the request header would have a request code of ('4282') and the OIC that indicates the end of distribution is ('4282').

Table D-1. Intermediate Node—Add Data ('0281')

<i>NetView DM Phase Status</i>	<i>NetView DM Description</i>	<i>Data and Direction</i>	<i>DSNX Description</i>	<i>DSNX Journal Entry</i>
READY		Bind session		
EXEC	Add data	----- FIC '0281' ----->		<b>A</b>
	data	----- MIC ----->		
	data	----- MIC ----->		
	data	----- MIC ----->		
	data	----- LIC ----->		
PEND		<----- (+)Response -----		
	End of distribution	----- OIC '0281' ----->		<b>B</b>
		<----- (+)Response -----		<b>C</b>
	Query	----- OIC '0286' ----->		<b>D</b>
		<----- (-)Response 'F004'	No status	
		End session		
		<----- Logon -----	Request processed Start a session	<b>E</b> <b>F</b>
	Reject logon	----- NSPE ----->		
		End session		
		Bind session		
	Query	----- OIC '0286' ----->		<b>G</b>
		<----- (+)Response -----		
		<----- FIC '4286' -----	Query response	
		<----- LIC -----	Delayed ack	<b>H</b>
		----- (+)Response ----->		
COMPL		End session		

## Journal Entries for Intermediate Node—Add Data

Table D-2 shows the meaning of the journal entry letter codes used in Table D-1 on page D-3.

<b>Entry</b>	<b>Type</b>	<b>Program</b>	<b>Description</b>
<b>A</b>	'05'x	QDXHRCV	Add request header received (0281)
<b>B</b>	'05'x	QDXHRCV	End of distribution received (0281)
<b>C</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>D</b>	'05'x	QDXHRCV	Query request header received (0286)
<b>E</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>F</b>	'04'x	QDXDDOER	Function run by request processor
<b>G</b>	'05'x	QDXHRCV	Query request header received (0286)
<b>H</b>	'06'x	QDXHSEND	Delayed acknowledgement sent to host system

## Intermediate Node Successful Retrieve with Compress Function

Table D-3 on page D-5 shows some of the data flow for a **Retrieve** function with the **Compress** option. A typical occurrence for a **Compress** request is that the data is not available from the request processor when the first query comes from NetView DM. This causes the session to be dropped and the AS/400 system must do a logon operation with the host system to inform NetView DM that a response is available.

A **Retrieve** request is the same except that the OIC with the request header has a request code of ('0391') instead of ('4391').



Table D-3. Intermediate Node—Retrieve with Compress ('4391')

<i>NetView DM Phase Status</i>	<i>NetView DM Description</i>	<i>Data and Direction</i>	<i>DSNX Description</i>	<i>DSNX Journal Entry</i>
READY		Bind session		
EXEC	Compress	----- OIC '4391' ----->		<b>A</b>
PEND		<----- (+)Response -----		<b>B</b>
	Query	----- OIC '0286' -----> <----- (-)Response 'F004'	No status	<b>C</b>
		End session	Request processed	<b>D</b> <b>E</b>
		<----- Logon -----	Start a session	
	Reject logon	----- NSPE ----->		
		End session		
		Bind session		
	Query	----- OIC '0286' -----> <----- (+)Response -----		<b>F</b>
		<----- OIC '4286' ----- ----- (+)Response ----->	Dataset ready	
	Send data	----- OIC '0391' ----->		<b>G</b>
EXEC		<----- (+)Response -----		
		<----- FIC '0491' -----	Send data response data	
		<----- MIC -----	data	
		<----- MIC -----	data	
		<----- LIC -----	data	
		----- (+)Response ----->		<b>H</b>
COMPL		End session		

## Journal Entries for Intermediate Node—Retrieve with Compress

Table D-4 shows the meaning of the journal entry letter codes used in Table D-3 on page D-5.

Entry	Type	Program	Description
<b>A</b>	'05'x	QDXHRCV	Compress request header received (4391)
<b>B</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>C</b>	'05'x	QDXHRCV	Query request header received (0286)
<b>D</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>E</b>	'04'x	QDXDDOER	Function run by request processor
<b>F</b>	'05'x	QDXHRCV	Query request header received (0286)
<b>G</b>	'05'x	QDXHRCV	Request header for Send Data received (0391)
<b>H</b>	'07'x	QDXHSEND	Data sent to host system

## Intermediate Node Unsuccessful Delete Function

Table D-5 on page D-7 shows some of the data flow for a **Delete** function when the database member to be deleted is locked. This causes the host interface to send a delayed acknowledgement with an error message in it.

This flow differs from a successful **Delete** function as follows:

- The request processor would have a journal entry type of '04' (which indicates a successful request) instead of '16'.
- The delayed acknowledgement in an error flow contains an error message indicating the reason the **Delete** function failed. If no error had occurred, the delayed acknowledgement would have no error message and the function is considered successful by NetView DM.

A request processor can usually process a **Delete** function fast enough to have a response ready for the host interface when NetView DM sends the query.

NetView DM Phase Status	NetView DM Description	Data and Direction	DSNX Description	DSNX Journal Entry
READY		Bind session		
EXEC	Delete	----- OIC '0283' ----->		<b>A</b>
PEND		<----- (+)Response -----		<b>B</b>
			Error detected	<b>C</b> <b>D</b>
	Query	----- OIC '0286' -----> <----- (+) Response -----		<b>E</b>
EXEC		<----- FIC '4286' ----- <----- LIC -----  ----- (+)Response ----->	Query response Delayed ack	<b>F</b>
COMPL		End session		

## Journal Entries for Intermediate Node—Unsuccessful Delete Function

Table D-6 shows the meaning of the journal entry letter codes used in Table D-5.

Entry	Type	Program	Description
<b>A</b>	'05'x	QDXHRCV	Delete request header received (0283)
<b>B</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>C</b>	'16'x	QDXDDOER	Error detected by request processor
<b>D</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>E</b>	'05'x	QDXHRCV	Query request header received (0286)
<b>F</b>	'06'x	QDXHSEND	Delayed acknowledgement sent to host system

## Intermediate Node—Unsuccessful Replace Function

Table D-7 on page D-8 shows some of the data flow for a **Replace** function when the following problems are found:

- A link error occurs (such as a line failure) when NetView DM is sending data.
- After correcting the link problem, a **Replace** request with **Recover** option ('8282') is attempted but the AS/400 system has lost all the information about

the outstanding function. (For example, the information would be lost if the QDSNX library was deleted.)

<i>Table D-7. Intermediate Node—Unsuccessful Replace Function ('0282')</i>				
<i>NetView DM Phase Status</i>	<i>NetView DM Description</i>	<i>Data and Direction</i>	<i>DSNX Description</i>	<i>DSNX Journal Entry</i>
READY		Bind session		
EXEC	Replace data	----- FIC '0282'		<b>A</b>
	data	----->		
	data	----- MIC ----->		
	data	----- MIC ----->		
REST		*** Link Failure ***		<b>B</b>
REST		*** Link Restored ***		
		Bind session		
EXEC	Recover	----- OIC '8282'		<b>C</b>
		----->		
		<----- (-)Response	No information	<b>D</b>
		-----		
COMPL		End session		

## Journal Entries for Intermediate Node—Unsuccessful Replace

Table D-8 shows the meaning of the journal entry letter codes used in Table D-7.

<i>Table D-8. Journal Entries for Intermediate Node—Unsuccessful Replace</i>			
<i>Entry</i>	<i>Type</i>	<i>Program</i>	<i>Description</i>
<b>A</b>	'05'x	QDXHRCV	Replace request header received (0282)
<b>B</b>	'1C'x	QDXHRCV	Unexpected condition (link failure)
<b>C</b>	'05'x	QDXHRCV	Recover Replace request header received (8282)
<b>D</b>	'13'x	QDXHRCV	Protocol error because no information available

## Direct Node Successful Replace Function

Table D-9 on page D-9 shows some of the data flow for a **Replace** function. All data flows in the same Session in this direct node function.

A direct node **Add** request would be the same except that the FIC with the request header would have a request code of ('0281') instead of ('0282').

**Note:** A direct node **Replace** request with **decompress** ('4282') option is a different flow. See Table D-13 on page D-11.

<i>Table D-9. Direct Node—Replace ('0282')</i>				
<i>NetView DM Phase Status</i>	<i>NetView DM Description</i>	<i>Data and Direction</i>	<i>DSNX Description</i>	<i>DSNX Journal Entry</i>
READY		Bind session		
EXEC	Replace data	----- FIC '0282'		<b>A</b>
	data	----->		
	data	----- MIC ----->		
	data	----- MIC ----->		
	data	----- MIC ----->		
	data	----- LIC ----->		<b>B</b>
			Request processed	<b>C</b>
				<b>D</b>
		<----- (+)Response		<b>E</b>
		-----		
COMPL		End session		

## Journal Entries for Direct Node—Replace

Table D-10 shows the meaning of the journal entry letter codes used in Table D-9.

<i>Table D-10. Journal Entries for Direct Node—Replace</i>			
<i>Entry</i>	<i>Type</i>	<i>Program</i>	<i>Description</i>
<b>A</b>	'05'x	QDXHRCV	Replace request header received (0282)
<b>B</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>C</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>D</b>	'04'x	QDXDDOER	Function run by request processor
<b>E</b>	'06'x	QDXHRTR	Response sent to host system

## Direct Node Successful Retrieve with Compress Function

Table D-11 on page D-10 shows some of the data flow for a **Retrieve** function with the **Compress** option.

This flow is different from intermediate node support because the AS/400 system will always wait for the response from the request processor to be available before sending the positive (+) response to the **Compress** request header (OIC '4391'). Only one session is used to complete the request because the AS/400 system waits for the response.

Table D-11. Direct Node—Retrieve with Compress (4391)

<i>NetView DM Phase Status</i>	<i>NetView DM Description</i>	<i>Data and Direction</i>	<i>DSNX Description</i>	<i>DSNX Journal Entry</i>
READY		Bind session		
EXEC	Compress	----- OIC '4391' ----->		<b>A</b>
			Request processed	<b>B</b> <b>C</b> <b>D</b>
PEND		<----- (+)Response -----		
	Query	----- OIC '0286' -----> <----- (+)Response ----- <----- OIC '4286' -----	Dataset ready	<b>E</b>
	Send data	----- (+)Response ----->		
EXEC		----- OIC '0391' -----> <----- (+)Response ----- <----- FIC '0491' ----- <----- MIC ----- <----- MIC ----- <----- MIC ----- <----- LIC -----	Send data response data data data data	<b>F</b>
		----- (+)Response ----->		<b>G</b>
COMPL		End session		

## Journal Entries for Direct Node—Retrieve with Compress

Table D-12 on page D-11 shows the meaning of the journal entry letter codes used in Table D-11.

<i>Table D-12. Journal Entries for Direct Node—Retrieve with Compress</i>			
<b>Entry</b>	<b>Type</b>	<b>Program</b>	<b>Description</b>
<b>A</b>	'05'x	QDXHRCV	Compress request header received (4391)
<b>B</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>C</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>D</b>	'04'x	QDXDDOER	Function run by request processor
<b>E</b>	'05'x	QDXHRCV	Query request header received (0286)
<b>F</b>	'05'x	QDXHRCV	Request header for Send Data received (0391)
<b>G</b>	'07'x	QDXHSEND	Data sent to host system

## Direct Node Successful Replace with Decompress Function

Table D-13 shows some of the data flow for a **Replace** function with the **Decompress** option. All data flows in the same session in this direct node function.

<i>Table D-13. Direct Node—Replace with Decompress (4282)</i>				
<b>NetView DM Phase Status</b>	<b>NetView DM Description</b>	<b>Data and Direction</b>	<b>DSNX Description</b>	<b>DSNX Journal Entry</b>
READY		Bind session		
EXEC	Decompress data data data data	----- FIC '4282' -----> ----- MIC -----> ----- MIC -----> ----- MIC -----> ----- LIC ----->		<b>A</b>
			Request processed	<b>B</b> <b>C</b> <b>D</b>
PEND	Query	<----- (+)Response ----- ----- OIC '0286' -----> <----- (+)Response ----- <----- FIC '4286' ----- <----- LIC ----- ----- (+)Response ----->	Query response Delayed ack	<b>E</b> <b>F</b>
COMPL		End session		

## Journal Entries for Direct Node—Replace with Decompress

Table D-14 shows the meaning of the journal entry letter codes used in Table D-13 on page D-11.

Entry	Type	Program	Description
<b>A</b>	'05'x	QDXHRCV	Decompress request header received (4282)
<b>B</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>C</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>D</b>	'04'x	QDXDDOER	Function run by request processor
<b>E</b>	'05'x	QDXHRCV	Query request header received (0286)
<b>F</b>	'06'x	QDXHSEND	Delayed acknowledgement sent to host system

## Direct Node Successful Retrieve CLIST Function

Table D-15 shows some of the data flow for a **Retrieve** of a CLIST function. All data flows in the same session in this direct node function.

NetView DM Phase Status	NetView DM Description	Data and Direction	DSNX Description	DSNX Journal Entry
READY		Bind session		
EXEC	Retrieve	----- OIC '0391' ----->	Request processed	<b>A</b> <b>B</b> <b>C</b> <b>D</b>
		<----- (+)Response -----		
		<----- FIC '0491' -----	Retrieve response	
		<----- MIC -----	data	
		<----- MIC -----	data	
		<----- MIC -----	data	
		<----- LIC -----	data	
		----- (+)Response ----->		<b>E</b>
COMPL		End session		



## Journal Entries for Direct Node—Retrieve CLIST

Table D-16 shows the meaning of the journal entry letter codes used in Table D-15 on page D-12.

Entry	Type	Program	Description
<b>A</b>	'05'x	QDXHRCV	Retrieve request header received ('0391')
<b>B</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>C</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>D</b>	'04'x	QDXDDOER	Function run by request processor
<b>E</b>	'07'x	QDXHSEND	Data sent to host system

## Direct Node Unsuccessful Initiate Job Function

Table D-17 shows some of the data flow for the **Initiate Job** function when the data sent is not a valid AS/400 batch job stream. The error data will flow in a **Status** response unit (RU) following a negative (-) response.

NetView DM Phase Status	NetView DM Description	Data and Direction	DSNX Description	DSNX Journal Entry
READY		Bind session		
EXEC	Initiate data	----- FIC '0284' ----->		<b>A</b>
	data	----- MIC ----->		
	data	----- MIC ----->		
	data	----- MIC ----->		
	data	----- LIC ----->		<b>B</b>
			Error detected	<b>C</b>
				<b>D</b>
		<----- (-)Response '0846'		
		<----- OIC '05C0'	Status RU	<b>E</b>
		-----		
		----- (+)Response		
		----->		
COMPL		End session		

## Journal Entries for Direct Node—Initiate Job

Table D-18 shows the meaning of the journal entry letter codes used in Table D-17 on page D-13.

<b>Entry</b>	<b>Type</b>	<b>Program</b>	<b>Description</b>
<b>A</b>	'05'x	QDXHRCV	Initiate job request header received (0284)
<b>B</b>	'0A'x	QDXHRCV	Request distributed to local request processor
<b>C</b>	'16'x	QDXDDOER	Error trying to submit job
<b>D</b>	'03'x	QDXORCV	Response distributed to host interface through OD
<b>E</b>	'06'x	QDXHRTR	Response sent to host system

---

## Bibliography

The following publications provide additional information about the topics described or referred to in this book. The books are listed with their full titles and order numbers.

---

### IBM Publications

#### Communications and Programming

The following IBM AS/400 publications provide additional information about topics described or referred to in this book:

- *Backup and Recovery*, SC41-5304, provides information to help you become familiar with AS/400 functions, develop a backup plan, and recover from system failures.
  - *APPN Support*, SC41-5407, provides information about the concepts of AS/400 advanced peer-to-peer networking (APPN) and about planning APPN networks.
  - *APPC Programming*, SC41-5443, describes the advanced program-to-program communications (APPC) support for the AS/400 system and provides the information necessary for developing communications application programs.
  - *SNA Distribution Services*, SC41-5410, provides the information about using Systems Network Architecture distribution services (SNADS), object distribution, VM/MVS bridge, and the system distribution directory.
  - *ISDN Support*, SC41-5403, contains information on AS/400 connectivity to an integrated services digital network (ISDN) using AS/400 integrated communications adapter.
  - *LAN and Frame Relay Support*, SC41-5404, contains information on using an AS/400 system in a token-ring network, Ethernet network, or bridged network environment.
  - *X.25 Network Support*, SC41-5405, contains information on using AS/400 systems in an X.25 network.
  - *Communications Management*, SC41-5406, contains information about operating communications and handling communications errors.
  - *Communications Configuration*, SC41-5401, contains general configuration information, including descriptions of network interface, line, controller, device, modes and class-of-service descriptions.
- Information about configuration lists and connection lists is also included.
- *SNA Upline Facility Programming*, SC41-5446, contains the programming information for using the system network architecture (SNA) upline facility with the AS/400 system. This book describes how to set up the upline facility, how to write application programs for the SNA upline facility, and the return codes that the SNA upline facility can send to a program.
  - *OSI CS/400 Configuration*, SC41-3425, indicates how to gather information needed to identify the local node in the surrounding OSI environment, and communicate with the desired destination nodes, using relay nodes, if necessary. It provides worksheets for gathering this information, and instructs users on how to enter the information interactively using the Administrative Facility menu, list, and prompt panels, or using CL configuration commands.
  - *OSI CS/400 Operations*, SC41-3426, provides information about using OSI alerts.
  - *CL Programming*, SC41-5721, provides a discussion of AS/400 programming topics, such as a general discussion of objects and libraries, control language (CL) programming, messages and message handling, user-defined commands and menus, and application testing.
  - *CL Reference*, SC41-5722, provides a description of the AS/400 control language (CL) and its commands.
  - *Work Management*, SC41-5306, provides information on how to set up an initial work management environment and change work management objects.
  - *System API Reference*, SC41-5801, provides a description of the OS/400 application programming interfaces (APIs). Included in this book is information about using the alerts APIs: QALGENA, QALSNDNA, and QALRTVA.

#### NetView

- *Learning About NetView: Network Concepts*, SK2T-0292 (PC Diskette)
- *NetView Administration Reference*, SC30-3361
- *NetView Command Lists*, SC30-3423
- *NetView Command Summary*, SX27-3620
- *NetView Customization*, LY30-5586
- *NetView Diagnosis*, LY30-5587
- *NetView Hardware Problem Determination Reference*, SC30-3366
- *NetView Installation and Administration* book, SC30-3360

- *NetView Messages*, SC30-3365
- *NetView Operation*, SC30-3364
- *NetView Operation Primer*, SC30-3363
- *NetView Operation Scenarios*, SC30-3376
- *Network Program Products Bibliography and Master Index*, GC30-3353
- *Network Program Products General Information*, GC30-3350
- *Network Program Products Planning*, SC30-3351
- *Network Program Products Samples: NetView*, SC30-3352
- *Network Program Products Storage Estimates*, SC30-3403

## NetView Distribution Manager

- *NetView Distribution Manager Application Programming*, SH19-6591
- *NetView Distribution Manager Diagnosis*, LY19-6293
- *NetView Distribution Manager General Information*, GH19-6587
- *NetView Distribution Manager Installation*, SH19-6590
- *NetView Distribution Manager Messages and Codes*, SH19-6595
- *NetView Distribution Manager Operation*, SH19-6592
- *NetView Distribution Manager Planning*, SH19-6589
- *NetView Distribution Manager User's book*, SH19-6602

## Advanced Communications Function for Virtual Telecommunications Access Method (ACF/VTAM)

- *ACF/VTAM\* General Information*, GC38-0254
- *ACF/VTAM System Programmer's*, SC38-0258

## Systems Network Architecture (SNA)

- *Systems Network Architecture Technical Overview*, GC30-3072
- *Systems Network Architecture Formats*, GA27-3136
- *Systems Network Architecture Format and Protocol Reference*, SC30-3112

- *Systems Network Architecture—Sessions Between Logical Units*, GC20-1868

## Data Link Control

- *IBM Synchronous Data Link Control Concepts*, GA27-3093

## Communications Controllers

- *Book to Using the IBM 3704 Communications Controller Control Panel*, GA27-3086
- *Book to Using the IBM 3705 Communications Controller Control Panel*, GA27-3087
- *IBM 3704 and 3705 Control Program Generation and Utilities Guide and Reference*, GC30-3008
- *IBM 3704 and 3705 Program Reference Handbook*, GY30-3012
- *IBM 3725 Communications Controller, Principles of Operation*, GA33-0013
- *IBM 3725 Model 1 Communications Controller, Introduction*, GA33-0010
- *IBM 3725 Model 2 Communications Controller, Introduction*, GA33-0021
- *Introduction to the IBM 3704 and 3705 Communications Controllers*, GA27-3051

## Personal Computer

- *Client Access/400 for DOS and OS/2 Technical Reference*, SC41-3563
- *Client Access/400 for DOS with Extended Memory Setup*, SC41-3500
- *Client Access/400 for OS/2 Setup*, SC41-3520
- *Personal Computer/Distributed System Node Executive Installation and Operation*, SH19-6585

## System/38

- *Data Communications Programmer's Book*, SC21-7825

---

# Index

## A

- ACF/VTAM reference book**
  - list H-2
- ADD MASTER control statement**
  - LOGMODE parameter 3-7
- ADDR parameter**
  - PU macroinstruction 2-4
- address**
  - local
    - definition 2-5
  - station
    - definition 2-4
- advanced program-to-program communications (APPC)**
  - definition 1-2
- alert support**
  - configuration
    - matching values, work sheet for 2-6
- APPC (advanced program-to-program communications)**
  - definition 1-2
- application library**
  - non-DSNX supported procedure 5-1
  - transferring procedure 5-2
- AS/400 system**
  - application ID or node LU wrong 4-10
  - as development systems 5-2
  - automatic response to CLIST messages 4-6
  - change management 2-2
  - CLIST processing 4-5
  - compressed data 4-11
  - considerations using DSNX 4-10
  - data considerations using NetView DM 4-9
  - defining to NetView DM 3-7
  - differences from System/36 DSNX 6-1
  - distributed systems node executive (DSNX)
    - security considerations 4-4
    - supported operations 2-2, 3-1
  - host system work sheet 2-6
  - IBM NetView Distribution Manager (NetView DM)
    - description 2-2
    - network 2-1
    - sessions 4-8
  - performance considerations 4-2
  - processing CLISTs sent by NetView DM 4-5
  - replacing objects on 4-7
  - resource group name 4-10
  - retrieving objects on 4-7
  - terms 2-2
- automatic response**
  - use for CLIST processing 4-6

## B

- bind command**
  - definition 3-7
  - starting DSNX 3-7

## C

- change management**
  - description of DSNX 2-1
  - limitations 2-3
  - supported operations 2-2
- CHANGE MASTER control statement**
  - LOGMODE parameter 3-7
- CLIST**
  - automatic response to messages 4-6
  - processed by an AS/400 system 4-5, 4-6
- cluster**
  - See node
- command, CL**
  - Display Journal (DSPJRN) B-6
  - DSPJRN (Display Journal) B-6
  - Grant Object Authority (GRTOBJAUT) 5-1
  - GRTOBJAUT (Grant Object Authority) 5-1
  - Work with DSNX/PC Distribution Queues (WRKDPCQ) 3-13
  - WRKDPCQ (Work with DSNX/PC Distribution Queues) 3-13
- communications books**
  - list H-1
- communications controllers**
  - reference books, list H-2
- communications interface B-5**
- configuration**
  - required for DSNX support 3-5
- control statement**
  - ADD MASTER 3-7
  - CHANGE MASTER 3-7
  - INITIATE FUNCTION 4-5
- controller**
  - in NetView DM network 2-1
  - reference books H-2
- correlation ID entries, DSNX B-11**

## D

- data considerations using NetView DM 4-9**
- deleting**
  - entries from the QDSNX journal B-13
- description**
  - configuration for
    - DSNX 3-5
    - host interface support 3-5

## **differences**

between AS/400 system and System/36 DSNX 6-1

## **direct connection, AS/400 system to NetView**

**DM 6-1**

## **direct node**

node name 3-7

successful replace function D-8

successful replace with decompress function D-11

successful retrieve CLIST function D-12

successful retrieve with compress function D-9

unsuccessful initiate job function D-13

## **DISCNT (disconnect physical unit) parameter**

PU macroinstruction 2-4

## **disconnect physical unit (DISCNT) parameter**

PU macroinstruction 2-4

## **Display Journal (DSPJRN) command B-6**

## **displaying**

DSNX journal entries B-6

Display Journal (DSPJRN) command B-6

## **distributed systems node executive (DSNX Support )**

introduction 1-1

## **distributed systems node executive (DSNX)**

CLIST processing 4-5

coexistence considerations 4-4

object compatibility 4-4

components 3-1

compressed data 4-11

configuration for

DSNX/&pcs.-to-personal computer 3-12

host interface-to-DSNX/&pcs. 3-10

host interface-to-request processor 3-8

considerations on AS/400 system 4-10

correlation ID entries format B-11

data flow to NetView DM D-1

definition 1-1

differences between AS/400 system and System/36 6-1

DSNX/&pcs. subsystem 3-4

environment 2-1

example program A-1

exception data entries format B-12

file limitations 2-3

host interface subsystem 3-2

host system logon 3-7

host system programming considerations 2-1

introduction 1-1

journal entries

XE B-6

XL B-6

journal formats B-9

logged data entries format B-11

member limitations 2-3

NetView DM version requirements 2-1

object naming 4-1

problem analysis

DSNX/Client Access B-6

## **distributed systems node executive (DSNX) (continued)**

problem analysis (continued)

host interface B-5

local DSNX/PC queue management B-6

logging, defined B-1

logging, DSNX/Client Access B-4

logging, local request B-2

logging, remote request B-3

request processor B-5

releasing a phase 4-2

request descriptions C-1

request processor subsystem 3-4

resource group name 4-10

saved copy

when replacing objects 4-7

when retrieving objects 4-7

scheduling communications 4-2

security considerations 4-4

sending the same resource to several nodes 4-5

starting

ADD MASTER control statement 3-7

CHANGE MASTER control statement 3-7

supported operations 2-2, 3-1

terminology 2-2

wrong application ID or node LU 4-10

XE entries, list B-5

XE format B-10

XI entries, list B-5

XL format B-9

## **distributing**

objects from NetView DM host 4-7

programs to previous release systems 5-1

## **DLOGMOD parameter**

LU macroinstruction 2-5, 3-7

PU macroinstruction 3-7

## **DSNX (distributed systems node executive)**

*See also* distributed systems node executive (DSNX)

CLIST processing 4-5

coexistence considerations 4-4

object compatibility 4-4

components 3-1

compressed data 4-11

configuration for

DSNX/&pcs.-to-personal computer 3-12

host interface-to-DSNX/&pcs. 3-10

host interface-to-request processor 3-8

considerations on AS/400 system 4-10

correlation ID entries format B-11

data flow to NetView DM D-1

definition 1-1

differences between AS/400 system and System/36 6-1

DSNX/&pcs. subsystem 3-4

environment 2-1

example program A-1

## **DSNX (distributed systems node executive) (continued)**

- exception data entries format B-12
  - file limitations 2-3
  - host interface subsystem 3-2
  - host system logon 3-7
  - host system programming considerations 2-1
  - introduction 1-1
  - journal entries
    - XE B-6
    - XL B-6
  - journal formats B-9
  - logged data entries format B-11
  - member limitations 2-3
  - NetView DM version requirements 2-1
  - object naming 4-1
  - problem analysis
    - DSNX/Client Access B-6
    - host interface B-5
    - local DSNX/PC queue management B-6
    - logging, defined B-1
    - logging, DSNX/Client Access B-4
    - logging, local request B-2
    - logging, remote request B-3
    - request processor B-5
  - releasing a phase 4-2
  - request descriptions C-1
  - request processor subsystem 3-4
  - resource group name 4-10
  - saved copy
    - when replacing objects 4-7
    - when retrieving objects 4-7
  - scheduling communications 4-2
  - security considerations 4-4
  - sending the same resource to several nodes 4-5
  - starting
    - ADD MASTER control statement 3-7
    - CHANGE MASTER control statement 3-7
  - supported operations 3-1
  - wrong application ID or node LU 4-10
  - XE entries, list B-5
  - XE format B-10
  - XI entries, list B-5
  - XL format B-9
- DSNX host interface B-5**
- DSNX Support**
- host work sheet for configuration 2-6
- DSNX Support (distributed systems node executive)**
- introduction 1-1
- DSNX/&pcs. subsystem**
- DSNX support 3-4
  - starting the subsystem 3-4
- DSPJRN (Display Journal) command B-6**
- DSX**
- See IBM NetView Distribution Manager (NetView DM)

## **E**

- ENCR (encryption) parameter**
- LU macroinstruction 2-5
- encryption (ENCR) parameter**
- LU macroinstruction 2-5
- entry specific data B-8**
- error log format (XE), DSNX B-10**
- Ethernet network**
- definition 3-12
- examples**
- CL commands for NetView DM Host to DSNX Host Interface Configuration 3-5
  - Distributing an AS/400 Application Library 5-3
  - Distributing Programs to Systems that Use a Previous Release 5-1
  - DSNX Host Interface Subsystem
    - create separate subsystem 3-3
    - use existing subsystem 3-2
  - DSNX Logon Modes (via NetView DM) 3-7
  - DSNX/Client/400 Subsystem 3-4
  - NetView DM to OS/400 DSNX configuration A-1
  - Retrieving and Printing a History File 5-4
  - Transferring Program Temporary Fixes (PTFs) 5-5
  - Transferring Spooled File Entries 5-5
- exception data entries, DSNX B-12**
- exchange identifier (XID) parameter**
- PU macroinstruction 2-4
- existing object**
- replaced by NetView DM host 4-7
  - retrieved by AS/400 system 4-7

## **F**

- file**
- sending
  - size considerations for DSNX 2-5
- format**
- DSNX log B-9

## **G**

- generalized interactive executive (GIX)**
- definition 3-7
  - network configuration
    - NetView DM 3-7
- generating VTAM/NCP**
- caution, for DSNX 2-5
  - considerations for DSNX support 2-4
  - considerations for NetView DM host 2-1
- GIX (generalized interactive executive)**
- definition 3-7
  - network configuration
    - NetView DM 3-7
- Grant Object Authority (GRTOBJAUT)**
- command 5-1

**GRTOBJAUT (Grant Object Authority)**  
command 5-1

## H

### history file

retrieving and printing 5-4

### host interface subsystem

DSNX support 3-2

starting the subsystem 3-2

### host interface, DSNX B-5

### host system

bind command references

NetView DM 3-7

definition 1-1

for NetView DM network

host programming considerations 2-1

operating requirement 2-1

sending objects to an AS/400 system 4-7

generation

caution, for DSNX 2-5

matching configuration values for AS/400 2-6

replacing objects on an AS/400 system 4-7

retrieving objects on an AS/400 system 4-7

sending CLISTs to OS/400 DSNX 4-5

VTAM/NCP generation considerations 2-4

## I

### IBM NetView Distribution Manager (NetView DM)

*See also* NetView Distribution Manager (NetView DM)

controllers 2-1

definition 1-1

description 2-2

environment 2-1

network 2-1

nodes 2-1

phase 2-1

plan 2-1

processors 2-1

sessions 4-8

systems in network

AS/400 system 2-1

personal computer 2-1

System/36 system 2-1

terminology 2-2

version requirements for DSNX 2-1

### IDBLK parameter

PU macroinstruction 2-4

### IDNUM parameter

PU macroinstruction 2-4

### information

related printed information H-1

### INITIATE FUNCTION control statement 4-5

### intermediate node

node name 3-7

successful add function D-2

successful retrieve with compress function D-4

unsuccessful delete function D-6

unsuccessful replace function D-7

### ISTATUS parameter

LU macroinstruction 2-5

PU macroinstruction 2-4

## J

### journal entry

definition B-1

XE B-6

XL B-6

### journal format

DSNX B-9

### journal receiver

definition B-13

## L

### LAN (local area network)

definition 3-12

### library

naming conventions 4-1

### Licensed Internal Code fix

definition 5-5

### limitations

DSNX

files received 2-3

members received 2-3

### LOCADDR (local address) parameter

LU macroinstruction 2-5

### LOCADR parameter

LU macroinstruction

host work sheet 2-6

### local address (LOCADDR) parameter

definition 2-5

LU macroinstruction 2-5

### local area network (LAN)

definition 3-12

### log

correlation ID entries, DSNX B-11

error (XE) B-10

exception data entries, DSNX B-12

logged data entries, DSNX B-11

### log format

DSNX B-9

### logged data entry

DSNX B-11

### logical unit (LU)

host system 2-5

macroinstruction

DLOGMOD parameter 2-5

ENCR parameter 2-5



## **logical unit (LU)** *(continued)*

macroinstruction *(continued)*

  ISTATUS parameter 2-5

  LOCADDR parameter 2-5

  PACING parameter 2-5

  VTAM/NCP generation 2-5

## **LOGMODE parameter**

  ADD MASTER control statement 3-7

  CHANGE MASTER control statement 3-7

## **logon mode table**

  DLOGMOD parameter 3-7

  NetView DM host 3-7

## **LU (logical unit)**

  host system 2-5

  macroinstruction

    DLOGMOD parameter 2-5

    ENCR parameter 2-5

    ISTATUS parameter 2-5

    LOCADDR parameter 2-5

    PACING parameter 2-5

    VTAM/NCP generation 2-5

# **M**

## **manual, reference**

  ACF/VTAM H-2

  communications H-1

  communications controllers H-2

  NetView H-1

  NetView Distribution Manager H-2

  personal computer H-2

  SNA H-2

  System/38 H-2

## **matching value**

  during configuration and host generation 2-6

  SSCP identifier 2-6

## **MAXDATA (maximum amount of data) parameter**

  NCP generation macroinstruction 2-5

  PU macroinstruction 2-4

## **maximum amount of data (MAXDATA) parameter**

  NCP generation macroinstruction 2-5

  PU macroinstruction 2-4

## **MAXOUT parameter**

  PU macroinstruction 2-4

## **member**

  naming conventions 4-1

## **multiple node**

  sending the same resource to 4-5

# **N**

## **naming convention**

  libraries 4-1

  members 4-1

  objects 4-1

## **NCP generation macro**

  caution on MAXDATA parameter 2-5

## **NetView**

  reference books, list H-1

## **NetView Distribution Manager (NetView DM)**

  See also IBM NetView Distribution Manager (NetView DM)

### **CLISTs**

  replacing objects using 4-7

  retrieving objects using 4-7

  sending 4-5

  compressed data 4-11

  data considerations 4-9

  data flow to DSNX

    direct node, initiate job D-13

    direct node, replace D-8, D-11

    direct node, retrieve D-9

    direct node, retrieve CLIST D-12

    intermediate node, add D-2

    intermediate node, delete D-6

    intermediate node, replace D-7

    intermediate node, retrieve D-4

    introduction D-1

  defining

    AS/400 system to NetView DM 3-7

    personal computer to an AS/400 system 3-13

  host system bind references 3-7

  node considerations

    direct 4-8

    intermediate 4-9

  reference books, list H-2

  request/response format B-13

  resource group name 4-10

  sending bind command 3-7

  sessions 4-8

  to DSNX example program A-1

## **NetView DM**

  See IBM NetView Distribution Manager (NetView DM)

  See NetView Distribution Manager (NetView DM)

## **NetView DM host**

  generation considerations 2-1

  logon mode table 3-7

  programming considerations 2-1

## **NetView DM session and node considerations**

  direct node 4-8

  intermediate node 4-9

  sessions 4-8

## **node**

  defining PC to NetView DM 3-7

  definition 2-1

  direct 3-7, 4-8

  in NetView DM network 2-1

  intermediate 3-7, 4-9

## O

### object

- naming conventions 4-1
- replacing, on an AS/400 system 4-7
- retrieving, on an AS/400 system 4-7
- sending replacement objects 4-7

### object distribution

- definition 1-1

## P

### PACING parameter

- LU macroinstruction 2-5

### path

- definition 3-8

### PC node 3-13

### performance considerations

- DSNX communications scheduling 4-2

### personal computer

- manuals, list H-2
- NetView DM network 2-1

### phase 4-2

- NetView DM 2-1

### physical unit

- definition
- PU macroinstruction 2-4

### physical unit type (PUTYPE) parameter

- PU macroinstruction 2-5

### plan, transmission 4-2

### printing

- history file 5-4

### problem analysis

- DSNX B-6

### procedure

- distributing programs to previous release systems 5-1
- transferring AS/400 application libraries 5-2
- transferring spooled file entries 5-5

### processor

- in NetView DM network 2-1

### program temporary fix (PTF)

- definition 5-1
- distributing, procedure for 5-5
- non-DSNX supported procedure 5-1

### programming considerations

- NetView DM host 2-1
- replacing objects on an AS/400 system 4-7
- retrieving objects on an AS/400 system 4-7

### PTF (program temporary fix)

- definition 5-1
- distributing, procedure for 5-5
- non-DSNX supported procedure 5-1

### PU macroinstruction

- ADDR parameter 2-4
- disconnect (DISCNT) parameter 2-4

### PU macroinstruction (continued)

- exchange identifier 2-4
- IDBLK parameter 2-4
- IDNUM parameter 2-4
- ISTATUS parameter 2-4
- MAXDATA parameter 2-4
- MAXOUT parameter 2-4
- PUTYPE parameter 2-5
- SSCPFM parameter 2-5
- station address 2-4
- USSTAB parameter 2-5
- VTAM/NCP generation 2-4

### PUTYPE (physical unit type) parameter

- PU macroinstruction 2-5

## Q

### QDSNX journal

- deleting entries B-13
- displaying B-7

### queue

- DSNX/PC 3-13

## R

### receiving

- history file 5-4

### Release Remote Phase (RLSRMTPHS)

#### command 4-2

### releasing

- phase 4-2

### replacing

- files on an AS/400 system
  - types of files 2-3
- objects on an AS/400 system
  - considerations 4-7

### request description

- for DSNX C-1

### request processor subsystem

- DSNX support 3-4
- starting the subsystem 3-4

### retrieving

- history file
  - non-DSNX supported procedure 5-1
- objects on an AS/400 system
  - considerations 4-7

### RLSRMTPHS (Release Remote Phase)

#### command 4-2

### router

- definition 3-12

### routing entry

- definition 3-2

### RU size

- VTAM/NCP generation considerations 2-5

## S

### sample procedures

- distributing programs to previous release systems 5-1
- introduction to 5-1
- retrieving and printing the history file 5-4
- transferring application libraries 5-2
- transferring spooled file entries 5-5

### saved copy

- when replacing objects with DSNX 4-7

### scheduling communications, DSNX 4-2

### SNA distribution services (SNADS)

- definition 1-1

### SNA reference books, list H-2

### SNA upline facility (SNUF)

- definition 1-1

### SNADS (SNA distribution services)

- definition 1-1

### SNUF (SNA upline facility)

- definition 1-1

### SNUF subsystem

- matching configuration values for both systems work sheet for 2-6

### source system

- definition 5-4

### spooled file entry

- non-DSNX supported procedure 5-1
- transferring 5-5

### SSCP (system services control point)

- definition 4-8

### SSCP identifier

- on host work sheet 2-6

### SSCPFM parameter

- PU macroinstruction 2-5

### starting

#### DSNX

- ADD MASTER control statement 3-7
- bind command 3-7
- CHANGE MASTER control statement 3-7
- sessions 3-7

### statement

#### control

- ADD MASTER 3-7
- CHANGE MASTER 3-7
- INITIATE FUNCTION 4-5

### station address

- definition 2-4
- PU macroinstruction 2-4

### supported operations

- change management 2-2
- DSNX 2-2

### supported system 2-1

### system services control point (SSCP)

- definition 4-8

### System/36

- differences from AS/400 system DSNX 6-1
- NetView DM network 2-1

### System/38

- reference books, list H-2

### System/38 books, list H-2

## T

### target system

- definition 3-10

### TDLC (twinaxial data link control)

- definition 3-12

### token-ring network

- definition 3-12

### transferring

- application library 5-2
- programs to previous release systems 5-1
- spooled file entries 5-5

### transmission plan 4-2

### twinaxial data link control (TDLC)

- definition 3-12

## U

### user profile

- definition 3-3

### USS definition table (USSTAB) parameter

- PU macroinstruction 2-5

### USSTAB (USS definition table) parameter

- PU macroinstruction 2-5

## V

### Virtual Telecommunications Access

#### Method/Network Control Program (VTAM/NCP)

- generation considerations
  - DSNX file size 2-5
  - DSNX support 2-4
  - LU macroinstruction 2-5
  - PU macroinstruction 2-4
- reference books, list H-2

#### VTAM/NCP (Virtual Telecommunications Access

#### Method/Network Control Program)

- generation considerations
  - DSNX file size 2-5
  - DSNX support 2-4
  - LU macroinstruction 2-5
  - PU macroinstruction 2-4
- reference books, list H-2

## W

### Work with DSNX/PC Distribution Queues

#### (WRKDPCQ) command 3-13

### WRKDPCQ (Work with DSNX/PC Distribution

#### Queues) command 3-13

# **X**

## **XID (exchange identifier) parameter**

PU macroinstruction 2-4

---

# Reader Comments—We'd Like to Hear from You!

AS/400

DSNX Support

Version 4

Publication No. SC41-5409-00

Overall, how would you rate this manual?

	Very Satisfied	Satisfied	Dissatisfied	Very Dissatisfied
Overall satisfaction				

How satisfied are you that the information in this manual is:

Accurate				
Complete				
Easy to find				
Easy to understand				
Well organized				
Applicable to your tasks				

THANK YOU!

Please tell us how we can improve this manual:

---

---

---

---

---

May we contact you to discuss your responses?  Yes  No

Phone: (\_\_\_\_) \_\_\_\_\_ Fax: (\_\_\_\_) \_\_\_\_\_ Internet: \_\_\_\_\_

To return this form:

- Mail it
- Fax it
  - United States and Canada: **800+937-3430**
  - Other countries: **(+1)+507+253-5192**
- Hand it to your IBM representative.

Note that IBM may use or distribute the responses to this form without obligation.

\_\_\_\_\_  
Name

\_\_\_\_\_  
Address

\_\_\_\_\_  
Company or Organization

\_\_\_\_\_  
Phone No.



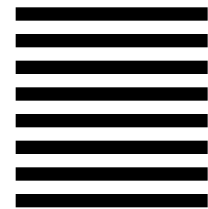
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES



# BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN DEPT 542 IDCLERK  
IBM CORPORATION  
3605 HWY 52 N  
ROCHESTER MN 55901-9986



Fold and Tape

Please do not staple

Fold and Tape





Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC41-5409-00





*Spine information:*



AS/400

**DSNX Support**

*Version 4*