



DOCUMENT INTERCHANGE ARCHITECTURE:
TECHNICAL REFERENCE

Program Number 5743-DIA

SC23-0781-0



DOCUMENT INTERCHANGE ARCHITECTURE: TECHNICAL REFERENCE

Program Number 5743-DIA

SC23-0781-0

First Edition (May 1985)

This manual is a total replacement for the following publications that are now obsolete: SC23-0759-0, SC23-0762-0, SC23-0760-0, and SC23-0761-0. It is a major revision and should be reviewed in its entirety. Changes are made periodically to the information herein; any such changes will be reported in subsequent revisions.

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 program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Department D33, 11400 Burnet Road, Austin, Texas 78758. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

ABOUT THIS BOOK

Document Interchange Architecture (DIA) is a program-to-program communication architecture that provides document interchange capabilities across a broad spectrum of IBM office systems.

Specifically, DIA defines the protocols and data structures that enable programs to communicate processing intentions and to interchange data in an IBM office system network. DIA is composed of several parts: an information interchange base and various DIA application services—Document Distribution Services, Document Library Services, File Transfer, and Application Processing Services.

- Document Distribution Services encompasses the protocols and commands to distribute a document and messages to a set of one or more recipients in a network.
- Document Library Services encompasses the protocols and commands to file, retrieve, search, and delete documents in a Document Interchange Architecture (DIA) document library.
- File Transfer Services encompasses the protocols and commands to file, retrieve, and deliver documents to and from user document libraries.
- Application Processing Services encompasses the protocols and commands to transform documents from one format to another, to modify document descriptors, and to execute user-supplied programs.

This manual describes the DIA information interchange base (DIA concepts, protocols, data structures, and session services) and DIA application services. This book also covers other common topics, such as function subsetting, session control, and error recovery. Restrictions and limitations imposed by product implementation of the architecture are not covered.

WHO SHOULD READ THIS BOOK

DIA Technical Reference is organized according to the information needs of the audience.

- Chapters 1 through 3 are intended primarily for data processing managers who need a detailed overview of DIA.
- Chapters 4 through 6 are intended for designers, systems analysts, system engineers and others who need to understand DIA protocols and data stream structured fields.
- Chapters 7 through 12 are written for systems programmers and application programmers who actually implement the commands.
- The appendixes are for those persons responsible for coding DIA applications.

HOW THIS BOOK IS ORGANIZED

Chapter 1, "Introduction" introduces IBM's office information architectures, identifies the needs of today's office, and explains how DIA addresses those needs.

Chapter 2, "Concepts" presents DIA concepts.

Chapter 3, "Services" introduces Document Interchange Architecture Services.

Chapter 4, "Request/Reply Protocols" details DIA request/reply protocols.

Chapter 5, "Document Interchange Unit" introduces the document interchange unit, its components and subcomponents.

Chapter 6, "Exception Detection, Classification, and Reporting" discusses exception detection, classification, and reporting.

Chapter 7, "Function Sets and Commands: Introduction" introduces function sets and commands.

Chapter 8, "Function Sets and Commands: DIA Session Services" discusses the commands for DIA Session Services.

Chapter 9, "Function Sets and Commands: Document Library Services" discusses the commands for Document Library Services.

Chapter 10, "Function Sets and Commands: File Transfer Services" discusses the commands for File Transfer Services.

Chapter 11, "Function Sets and Commands: Document Distribution Services" discusses the commands for Document Distribution Services.

Chapter 12, "Application Services" discusses the commands for Application Processing Services.

Appendix A, "Operands" provides a reference for operands defined by DIA.

Appendix B, "Code Points" provides a reference for code points defined for the DIA constructs.

Appendix C, "DIA Document Types" provides a reference for DIA document types.

Appendix D, "Graphic Character Sets" provides a reference for the graphic character sets.

Appendix E, "Character Set Translation" provides a reference for the graphic character set translation.

Appendix F, "DIU General Exception Conditions" provides a reference for the general exception conditions.

WHAT ELSE TO READ

- Office Information Architectures: Concepts, GC23-0765
- Document Interchange Architecture: Interchange Document Profile Reference, SC23-0764
- Document Interchange Architecture: Transaction Programmer's Guide, SC23-0763
- Document Content Architecture: Revisable-Form-Text Reference, SC23-0758
- Document Content Architecture: Final-Form-Text Reference, SC23-0757
- Registry of Graphic Character Sets and Codepages, IBM Corporate Specification C-H 3-3220-050
- Systems Network Architecture Concepts and Products, GC30-3072
- Systems Network Architecture Technical Reference, GC30-3073
- Systems Network Architecture Format and Protocol Reference Manual: Distribution Services, SC30-3098
- Systems Network Architecture Transaction Programmers Reference Manual for LU Type 6.2, GC30-3084.

SUMMARY OF AMENDMENTS

SC23-0769-1

This edition of Document Interchange Architecture: Technical Reference reflects the DIA Version 1.1 level of the architecture. This document is a consolidation of the reference manuals for DIA Concepts and Structures, Document Distribution Services, Document Library Services, and Application Processing Services. This reorganization was done to make the information more accessible and easier to understand. The changes consist of enhancements to DIA, architectural clarifications, and editorial corrections.

The enhancements include a new function set, inclusion of provisions for Systems Network Architecture Distribution Services interface, and registering of new document types and subprofiles. The new function set defines commands to provide the capability for transferring information into and out-of private user libraries. Provision for SNADS includes registering code points, defining the mapping between DIA capabilities and SNADS capabilities, and the redefinition of the character set rules. Code points were registered for new document types and subprofiles. Finally, another enhancement allows for one end user to initiate multiple DIA sessions.

In this edition, the acronym GCID has been replaced with CGCSGID.

CONTENTS

Chapter 1. Introduction	1
Office Information Architectures	1
Structural Relationship of DIA to Other Office Information Architectures	1
The Needs of Today's Offices	2
How Document Interchange Architecture Answers the Needs of Today's Offices	3
Chapter 2. Concepts	5
Introduction to DIA Data Streams	5
DIA Processes	5
Process Roles	6
Local and Remote DIA Processes	6
DIA Logical Components	7
Sources and Recipients in an Office System Network	7
Nodes as the Logical Components in DIA	7
DIA Logical Components in an Office System Network	8
Node Addresses	8
Relationship of DIA Processes, Logical Components, DIA Services, and Products	9
Chapter 3. Services	11
Function Sets	14
Restrictions on the Availability of DIA Services	15
Document Access	15
Levels of Ownership	15
Access Codes	16
Affinity	17
DIA Session Services	18
Document Library Services	18
Responsibilities of the Logical Components	19
Document Class	20
Search Result List	20
Document Descriptors	21
Affinity Relationships of Document Library Services	22
Document Distribution Services	23
Distribution Addressing	25
Source/Recipient Operand (End User) Addresses	26
Office System Node Address Operands	26
Rules for Use of Addressing Forms	27
Distribution Affinity	27
File Transfer Services	32
Responsibilities of the Logical Components	32
Application Processing Services	32
Affinity Relationships	34
Chapter 4. Request/Reply Protocols	35
General Rules	35
DIA Command Classes	37

DIA Request/Reply Correlation	37
DIA Command Class Protocol Rules	40
Chapter 5. Document Interchange Unit	41
Document Interchange Unit (DIU) Components	41
DIU Structured Field	42
DIU Introducer	42
DIU Introducer Extension (ISS)	44
Structured-Field Documentation Conventions	44
Document Interchange Unit (DIU) Structure	45
DIU Prefix	46
Command Sequence	47
Command	48
Operands	48
Data Units	49
Document Units	50
Document Unit Identifier	51
Document Profile	52
Document Content Introducer	53
DIU Suffix	54
Structured-Field Segmentation	56
Summary of DIU Syntax Rules	58
CGCSGID Encodings	60
Chapter 6. Exception Detection, Classification, and Reporting	61
Exception Condition Detection	61
Exception Condition Classification	61
Exception Condition Reporting	63
Recommended Recovery Actions	66
Chapter 7. Function Sets and Commands: Introduction	69
Function Sets	69
Function Set Negotiation	69
Function Set Definition	69
Command Description	70
Chapter 8. Function Sets and Commands: DIA Session Services	71
SIGN-ON	72
SIGN-OFF	81
ACKNOWLEDGE	82
SET-CONTROL-VALUE	86
Chapter 9. Function Sets and Commands: Document Library Services	91
DELETE	92
DELIVER	95
FILE	99
RETRIEVE	104
SEARCH	113
Chapter 10. Function Sets and Commands: File Transfer Services	119
DELIVER	120
FILE (Format 2)	124
RETRIEVE (Format 4)	127

Chapter 11. Function Sets and Commands: Document Distribution Services	133
CANCEL-DISTRIBUTION	137
DELIVER	144
LIST	151
Unformatted Summary Status	157
Unformatted Recipient Status	158
Unformatted Source Status	160
OBTAIN	165
PROCESS-BIT-STRING	173
REQUEST-DISTRIBUTION	177
STATUS-LIST	185
Chapter 12. Application Services	187
Application Processing Services Commands	187
DELIVER	188
EXECUTE	192
FORMAT	197
MODIFY	203
Appendix A. Operands	209
ACCESS-CODE (Format 41)	209
ATTRIBUTE-LIST (Format 1)	210
BIT STRING-REPRESENTATION (Format 1)	212
CANCEL-ACTION (Format 1)	213
CHARGE-CODE (Format 1)	214
CONTROL-VALUE (Format 1)	214
CORRELATION (Format 1)	216
COUNT (Format 1)	217
DESCRIPTOR-CONTENT-DEFINITION (Format 41)	217
DESTINATION-NODE-ADDRESS (Format 1)	219
DISTRIBUTION-IDENTIFIER (Format 1)	220
DISTRIBUTION-NAME (Format 1)	221
DOCUMENT-PASSWORD (Format 1)	221
DOCUMENT-TYPE (Format 1)	222
EXCEPTION-CODE (Format 1)	223
FILE-OPTION (Format 1)	224
FORMATTED-DOCUMENT-NAME (Format 1)	225
FORMATTER-NAME (Format 1)	225
FORMAT-PARAMETERS (Format 1)	225
FUNCTION-SET (Format 1)	226
GRAPHIC-CHARACTER-SET-ID (Format 1)	227
IDENTIFIED-DATA (Format 1)	228
IDENTIFIED-DATA (Format 2)	228
IDENTIFIED-DATA (Format 3)	229
IDENTIFIED-DATA (Format 41)	230
IDENTIFIED-DATA (Format 42)	231
LIBRARY-NAME (Format 1)	232
LIBRARY-NAME (Format 41)	232
LIST-ACTION (Format 1)	233
MESSAGE (Format 1)	235
MESSAGE (Format 2)	236
MODIFY-DATA (Format 41)	236
OBTAIN-OPTION (Format 1)	239

ORIGINATING-NODE-ADDRESS (Format 1)	241
PROCESS-PARAMETERS (Format 1)	242
PROCESS-NAME (Format 1)	242
PROCESS-PASSWORD (Format 1)	243
RECIPIENT-ADDRESS (Format 1)	243
RECIPIENT-ADDRESS (Format 42)	244
RECIPIENT-PASSWORD (Format 1)	245
RECOVERY-ACTION (Format 1)	245
REPLY-DATA (Format 1)	246
RETRIEVE-COUNT (Format 1)	246
SCAN-DATA (Format 1)	247
SEARCH-DATA (Format 41)	248
SEARCH-DATA (Format 42)	254
SEARCH-OPTION (Format 1)	258
SEARCH-REQUEST-NAME (Format 1)	259
SELECT-LIMIT (Format 1)	259
SIGN-ON-ID (Format 1)	259
SIGN-ON-ID (Format 42)	260
SIGN-ON-PASSWORD (Format 1)	261
SOURCE-ADDRESS (Format 1)	261
SOURCE-ADDRESS (Format 42)	262
SOURCE-PASSWORD (Format 1)	263
STATUS-INFORMATION (Format 1)	263
TIME-LIMIT (Format 1)	265
Appendix B. Code Points	267
Appendix C. DIA Document Types	277
Appendix D. Graphic Character Sets	279
Appendix E. Character Set Translation	291
Appendix F. DIU General Exception Conditions	297
Prefix Exception Conditions	298
Command Exception Conditions	299
Document Unit Exception Conditions	302
Data Unit Exception Conditions	302
Suffix Exception Conditions	303
Process Exception Conditions	303
Glossary	305
Index	311

FIGURES

1.	DIA and Other Office Information Architectures	2
2.	Logical Connection between Local DIA Processes	6
3.	Logical Connection between Remote DIA Processes through an SNA Network	7
4.	Relationship among Document Interchange Architecture Logical Components	8
5.	Structure of DIA Services within a DIA Server	11
6.	Local DIA Processes Using Document Library Services	12
7.	Remote DIA Processes	13
8.	Affinity Default Table for Document Library Services	22
9.	Distribution Address Format	25
10.	Affinity Default for LIST and OBTAIN Processing	28
11.	Affinity Default for Canceling a Distribution	29
12.	Affinity Default for Canceling Status	30
13.	Affinity Default for REQUEST-DISTRIBUTION	31
14.	Affinity Default for Application Processing Services	34
15.	DIU Structured Field	42
16.	Document Interchange Unit Structure Overview	45
17.	DIU Prefix	46
18.	DIU Command Sequence	47
19.	DIU Data Unit	49
20.	DIA Document Unit Type 3	50
21.	DIU Document Unit Identifier	51
22.	DIU Document Profile	52
23.	DIU Document Content Introducer	53
24.	DIU Suffix	54
25.	Type 2 Suffix Processing Summary	55
26.	Document Unit Segmentation Example	57
27.	Summary of DIU Syntax Rules	59
28.	Exception Condition Classes	62
29.	Exception Condition Code Format	63
30.	Exception Condition Coding Values	64
31.	Function Set 10	71
32.	Function Set 8	91
33.	Function Set 11	119
34.	Function Set 2	134
35.	Function Set 3	134
36.	Function Set 4	135
37.	Function Set 5	135
38.	Function Set 6	136
39.	Function Set 7	136
40.	Unformatted Summary Status Document Unit	157
41.	Unformatted Recipient Status Document Unit	158
42.	Unformatted Source Status Document Unit	160
43.	Function Set 9	187
44.	Document Type Code Assignments	278
45.	Character Set 103 of Code Page 256	283
46.	Character Set 946 of Code Page 256	284
47.	Character Set 930 of Code Page 256	285
48.	Character Set 337 of Code Page 256	286

49.	Description of Code Page 256 Graphics.	287
50.	Conversion Table (00946 to 00930)	291

CHAPTER 1. INTRODUCTION

Document Interchange Architecture (DIA) is one of the IBM office information architectures. This chapter introduces the IBM office information architectures and describes their roles in an IBM office system network. This chapter also describes the needs of today's offices and how Document Interchange Architecture meets those needs.

OFFICE INFORMATION ARCHITECTURES

Office information architectures specify the rules by which office systems networks operate. An office systems network is a collection of interconnected office systems and can include a number of products.

An architecture is a set of design principles that defines the structure of the connections and the model for the interactions among various parts of a system or network. Each of the following office information architectures addresses one aspect of information handling in an office:

- Document Content Architecture defines how a document's contents are formatted. For information on Document Content Architecture, see the Document Content Architecture references listed in "What Else To Read" in the preface of this book.
- Document Interchange Architecture (DIA) defines sets of **functions** that aid the interchange, distribution, application services, or storage of documents.
- Systems Network Architecture (SNA) defines synchronous and asynchronous protocols for routing documents across networks. This book's discussion of SNA is limited to how DIA uses SNA's transportation services. For an overview of SNA, see SNA Concepts and Products.

Structural Relationship of DIA to Other Office Information Architectures

The structure chart in Figure 1 illustrates the relationship between DIA and other IBM office information architectures.

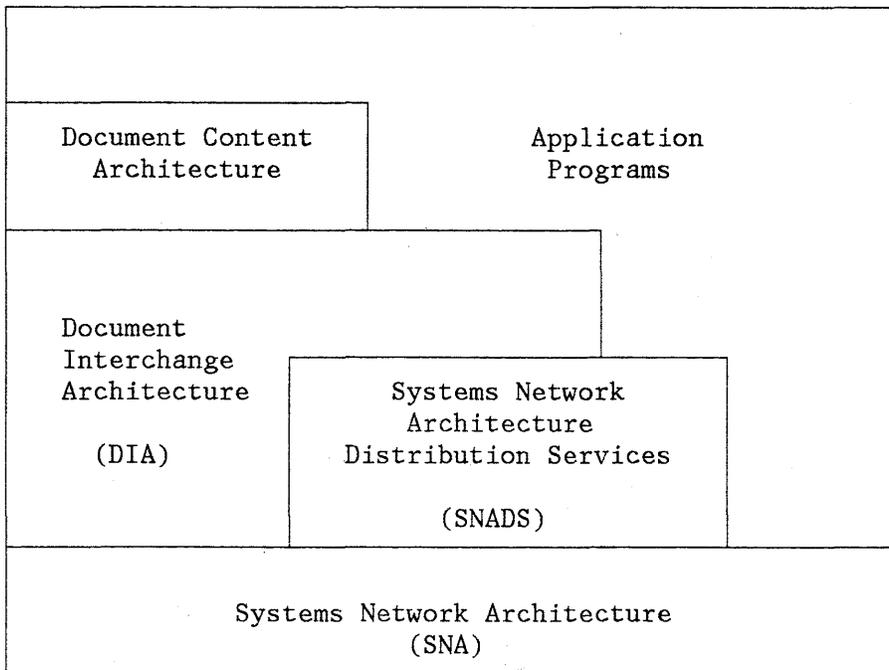


Figure 1. DIA and Other IBM Office Information Architectures

THE NEEDS OF TODAY'S OFFICES

Voice, text, data, images — the information handled in today's offices takes many forms. Throughout this book, the term document refers to a block of information of any size, regardless of its form. This definition includes facsimile system images, binary data files, digitized audio files, and other information not usually thought of as documents.

Office workers handle text documents, data documents, voice documents, and image documents, regardless of the extent to which electronic office systems are used. In handling these documents, office workers perform some combination of the following activities:

- Creating Documents. A salesman dictates a letter. A legal secretary prepares a contract. A catalog designer cuts up last year's catalog, deleting old items and including new ones. An executive combines three reports into a summary for a board of directors. People create correspondence, reports, proposals, contracts, and manuscripts, and they assemble documents from other documents that already exist.
- Revising Documents. A typist corrects one character in a letter. A sales manager adds a paragraph to a proposal. A researcher rewrites an entire report. In a typical office, people revise documents, making minor editorial corrections or rewriting the entire document.

- Presenting Documents. An artist designs a chart and wants to see its final output. A writer revises a draft and wants to see how the text will fall on a page.
- Distributing Documents. The sales office sends an electronic file of this month's performance to the regional office. A lawyer delivers an important document by hand. Interoffice mail delivers a file copy to a filing clerk's desk. People distribute documents to individuals or to files, through internal or external mail, hand delivery, or electronic means.
- Filing and Retrieving Documents. The librarian puts the XYZ Report on the shelf. A medical office assistant gets a patient's file from a file cabinet. The accounting clerk searches for and retrieves a customer's account file from electronic storage. A company vice-president searches for last July's PDQ account file. People file documents in file cabinets, libraries, or electronic storage; they retrieve files, and log and track those files.

HOW DOCUMENT INTERCHANGE ARCHITECTURE ANSWERS THE NEEDS OF TODAY'S OFFICES

Document Interchange Architecture (DIA) defines the functions for interchanging documents between separate office systems that are connected in a network. DIA provides the following services to interconnect office systems:

- Document Library Services
- Document Distribution Services
- Application Processing Services
- File Transfer Services.

Using one of the Document Library Services functions, a person or an application program can file documents in an available document library, retrieve them from the library, delete them from the library, or search for particular documents or categories of documents.

Using the Document Distribution Services functions, a person or an application program can deliver documents to recipients in the network. A user can schedule a distribution by document priority, confirm its delivery, and obtain reported errors. These services are commonly called electronic document distribution.

Using File Transfer Services, a person or an application program can file, retrieve and deliver documents to and from user document libraries.

Using Application Processing Services functions, a person or applications program can modify the file that describes a document so that a library search can find it. Users can call a program that transforms documents from revisable-form-text to final-form-text. A person can create a specialized program and execute it.

Document Interchange Architecture specifies uniform, consistent rules for exchanging documents. Users can add many parameters to DIA commands that communicate, without ambiguity, the user's instructions for filing, access, retrieval, and revision of the document. The coding scheme of DIA is flexible enough to convey any document, regardless of its content (text, data, image, or audio) from one system to another.

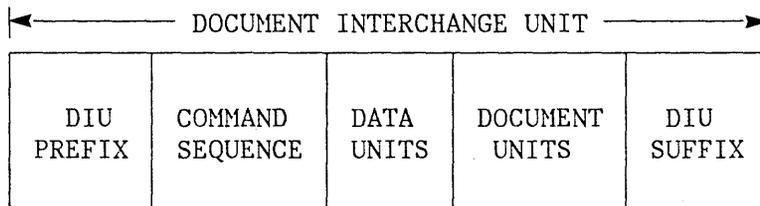
CHAPTER 2. CONCEPTS

This chapter describes the basic concepts of Document Interchange Architecture (DIA). It defines the logical components of an office system network and describes how documents are interchanged among those logical components.

INTRODUCTION TO DIA DATA STREAMS

Document Interchange Architecture defines the protocols and the formats necessary for data streams to interchange documents, consistently and predictably, in an office systems network. Protocols are the sequencing rules for, and the meanings of, the requests and replies that synchronize DIA processes. Data streams are the continuous succession of data elements being interchanged, in either character or binary-digit form, using a defined format.

Self-defined, structured segments called document interchange units (DIUs) make up DIA data streams. Document interchange units envelop the user's document with DIA commands and other fields. The figure below illustrates the components of the DIU.



The document interchange unit and each of its components begin with an introducer. The DIA introducer is a five-byte field. The first two bytes indicate the length (LL) of the component, the second two bytes indicate the general data stream identification code (ID), and the last byte indicates the format (F) of the component. Throughout this book, LLIDF refers to this introducer. For additional information about the document interchange unit, see Chapter 5, "Document Interchange Unit."

DIA PROCESSES

In an office system network, the physical control, data link control, path control, and transmission control layers determine the electrical switching of connections and the physical movement of data through the network. These layers control the network to make the most efficient use of the network's physical components. The user of the network never sees the effects of this control; to the user, this control is transparent.

Because the physical network is transparent, consider the network a connection between two users, regardless of the physical path used. This connection is called a logical connection. DIA processes are the logical processes that make this connection possible.

At either end of the logical connection is a user. In this sense, a user is a human being, an application program, or a device (a logical process, stored in microcoded hardware, as in programmable printers).

Generally, a process is a continuously executing sequence of actions. In this book, process refers to the sequence of logic in an office system network. A DIA process, then, is the continuously executing sequence of logic involved in the request for DIA services or in the fulfillment of that request.

A DIA session is the logical connection between two DIA processes, either within an office system or across a transport network.

Process Roles

When a DIA session is negotiated, the DIA processes establish the process roles of requester (DIA process B) and server (DIA process A). Further discussion of requester and server roles is provided under "DIA Logical Components."

Local and Remote DIA Processes

Document Interchange Architecture processes can be either local processes or remote processes. Local processes occur within an office system and do not require the facilities of a transport network. Remote processes occur across an external transport network, such as an SNA-supported network.

Figure 2 and Figure 3 illustrate the differences between local and remote DIA process connections.

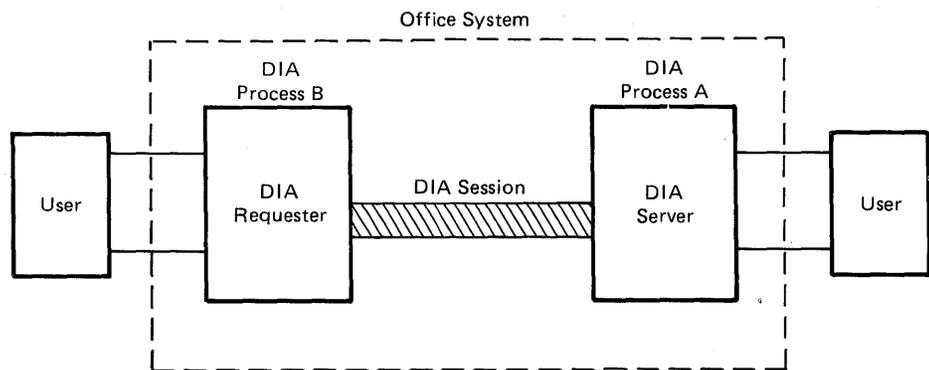


Figure 2. Logical Connection between Local DIA Processes

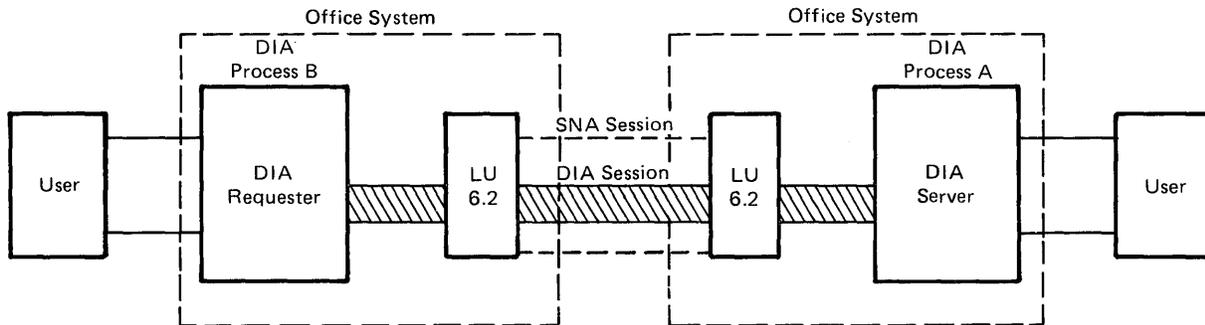


Figure 3. Logical Connection between Remote DIA Processes through an SNA Network

DIA LOGICAL COMPONENTS

The logical components of Document Interchange Architecture are source nodes, recipient nodes, and office system nodes. These logical components act on behalf of sources and recipients to request DIA services and to fulfill those requests.

Sources and Recipients in an Office System Network

In an office system network, users can be both sources and recipients of documents. Typically, source refers to users that send documents, and recipient refers to users that receive documents.

DIA defines the logical components that act on behalf of sources that are requesting DIA services. In fulfilling those requests, the logical components act on behalf of both sources and recipients.

Nodes as the Logical Components in DIA

In general, nodes are the locations within a network at which processes occur. Document Interchange Architecture defines processes as occurring at nodes. DIA defines three nodes according to their role in the interchange of documents.

- A source node is a location within an office system network at which a DIA process initiates and controls the interchange of documents.
- A recipient node is the location within an office system network at which a DIA process controls and receives documents. An office system network can require that recipient nodes request the delivery of documents. DIA defines these recipient nodes as being in a solicited environment. Alternatively, recipient nodes can receive documents as a matter of course. DIA defines these recipient nodes as being in an unsolicited environment.
- An office system node is the location within an office system network where a DIA process receives, stores, routes or delivers documents for source nodes and recipient nodes. Office system nodes that implement Document Library Services include storage called a document library.

DIA Logical Components in an Office System Network

An office system network connects source nodes, recipient nodes, and office system nodes. Source and recipient nodes can communicate with each other directly or have logical connections to the remainder of the office systems network through office system nodes.

Figure 4 illustrates relationships among users, source nodes, recipient nodes, and office system nodes.

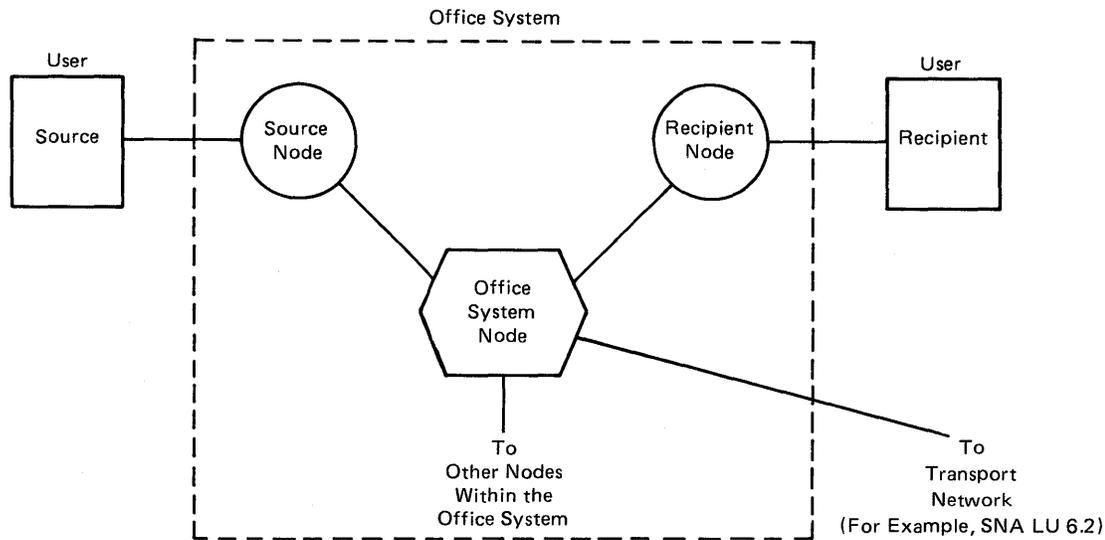


Figure 4. Relationship among Document Interchange Architecture Logical Components

Node Addresses

Each node has a logical address. The logical address of the source node is the source address. The logical address of the recipient node is the recipient address.

In Figure 4, the office system node can act either as an originating node for a document delivered to a recipient node and as a destination node for documents from the source node.

UNIQUENESS: Source addresses and recipient addresses are unique within an office system node. Originating node addresses and destination node addresses are unique within an office systems network. Source addresses are unique within originating nodes, and recipient addresses are unique within destination nodes.

ADDRESSES WHEN NODES OPERATE CONCURRENTLY AS DIFFERENT LOGICAL COMPONENTS: An office system node can act concurrently as an originating node and a destination node. When operating in this manner, the value that specifies the originating node address and the value that specifies the destination node address are identical. Likewise, when a DIA process acts as if it is located at both a source node and a recipient node, the value that specifies the source address and the value that specifies the recipient address are identical.

Relationship of DIA Processes, Logical Components, DIA Services, and Products

In the logical connection of DIA processes, a particular product that implements DIA can act as a source node, a recipient node, or an office system node. The DIA services that the product implements is the only limiting factor on what type of node it is. At a minimum, a product must implement the base services of DIA, including DIA Session Services.

The server (DIA process A) is usually at the office system node at which the requested DIA service is located. The requester (DIA process B) is the DIA process requesting the service.

The request/reply protocol for the command used to request the DIA service defines the sequence of requests and replies between the requester (process B) and the server (process A). Chapter 4, "Request/Reply Protocols" describes the general form of these protocols. The specific sequence of requests and replies for a particular command is found in the definition of that command.

CHAPTER 3. SERVICES

This chapter gives a comprehensive description of DIA services and explains the concepts of document ownership and user affinity. Finally, for each of the DIA services, this chapter defines the relationships among the logical components and describes the default assignments of affinity.

Document Interchange Architecture services perform the following on behalf of end users:

- Document Library Services file, retrieve, search, delete, and deliver documents to and from a DIA document library.
- Document Distribution Services distribute documents to a distribution list defined by the user. These services are limited by (1) document ownership, (2) the source node's capacity to act as an agent in distributing documents, and (3) the recipient node's capacity to act as an agent in receiving documents.
- File Transfer Services file, retrieve, and deliver documents to and from user document libraries.
- Application Processing Services execute user-created programs, format documents, modify document description and access controls, and deliver documents.
- DIA Session Services establish, control, and end the DIA session through which the user requests other DIA services and the DIA services fulfill these requests.

Figure 5 illustrates the structure of Document Interchange Architecture services within a DIA server.

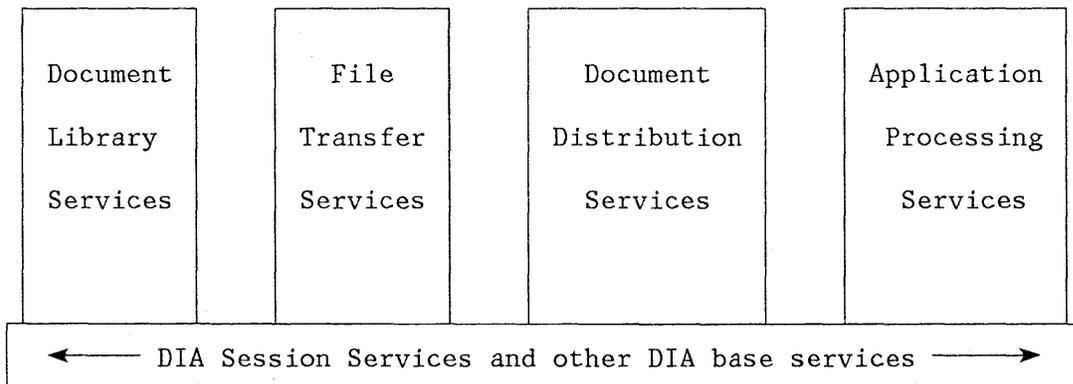


Figure 5. Structure of DIA Services within a DIA Server

A requester (DIA process B) can request DIA services. A server (DIA process A) can fulfill that request through either a local process or a remote process.

The following figure illustrates the logical connection of a local DIA process using Document Library Services, Document Distribution Services, File Transfer Services, and Application Processing Services, respectively:

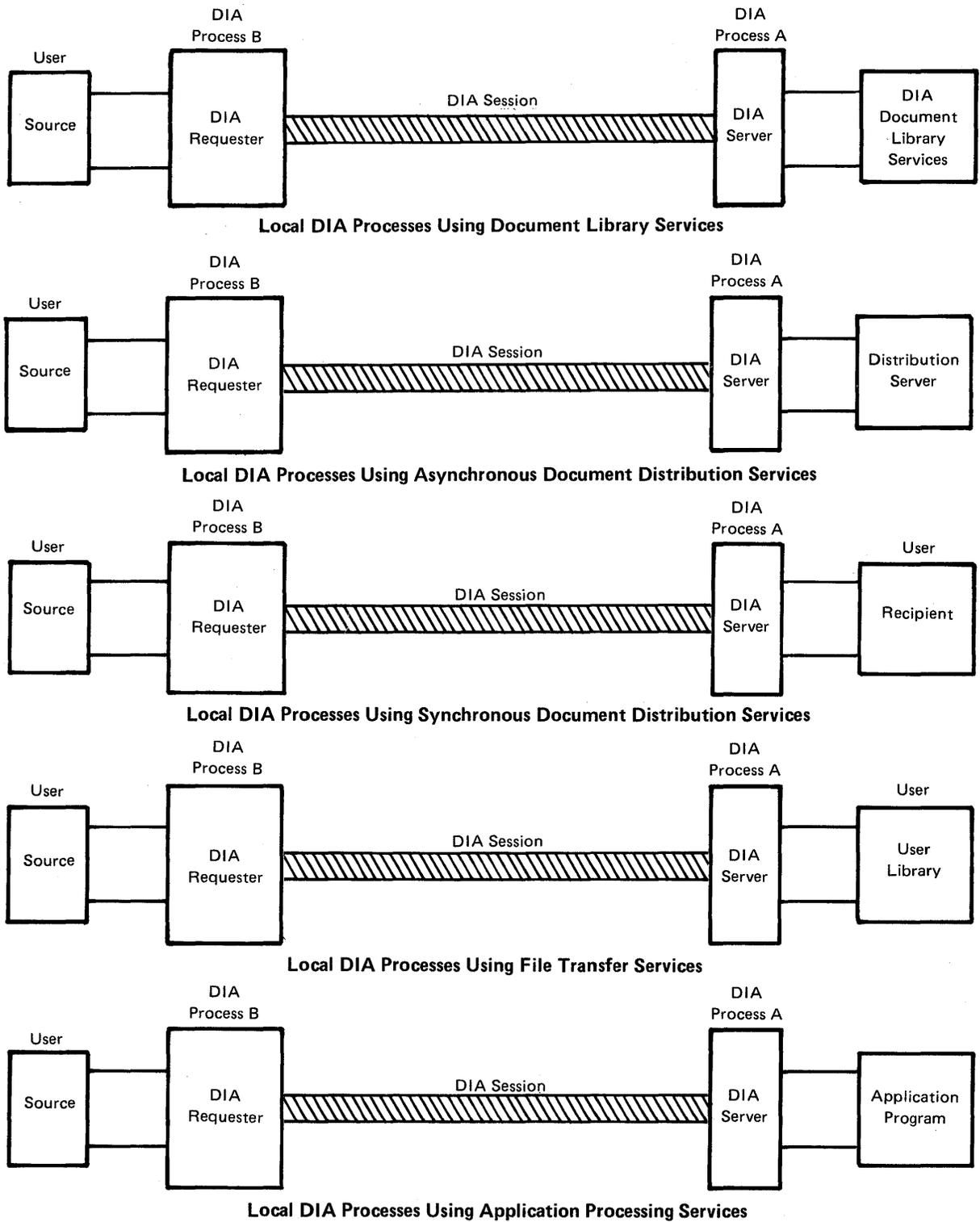


Figure 6. Local DIA Processes

The following figures illustrate the logical connection through an SNA network that uses Document Library Services, Document Distribution Services, File Transfer Services, and Application Processing Services, respectively:

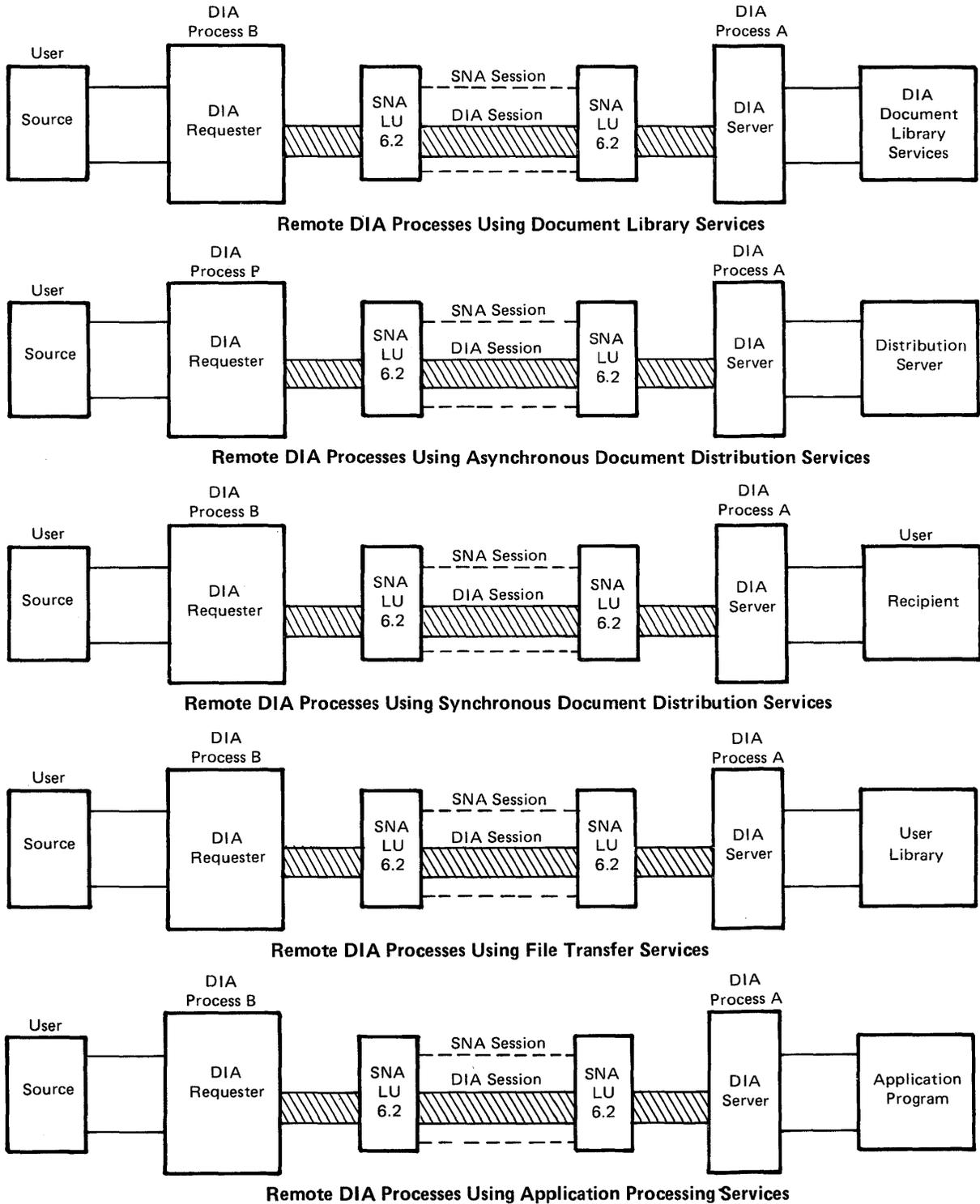


Figure 7. Remote DIA Processes

FUNCTION SETS

Each of the DIA servers provides one or more function sets. A function set is a set of commands that identify the scope of the services provided. Each function set contains all the commands required for a given category of services.

DIA defines the following function sets:

Document Distribution Services

- Function set 2 contains the commands necessary to deliver information from an office system node to a recipient node when the recipient node must request delivery (solicited environment).
- Function set 3 contains the commands necessary to deliver information from an office system node to a recipient node when the office system node delivers the information to the recipient node without being specifically requested to do so (unsolicited environment).
- Function set 4 contains the commands necessary to send documents and requests for DIA services from a source node that originates image data (facsimile) to an office system node.
- Function set 5 contains the commands necessary to initiate and control document distribution requests between a source node and an office system node.
- Function set 6 contains the commands necessary to send documents to a recipient node from a source node that originates image data (facsimile) without going through intermediate office system nodes.
- Function set 7 contains the commands necessary to distribute information between source nodes and recipient nodes without going through intermediate office system nodes.

Document Library Services

- Function set 8 contains the commands necessary to maintain user documents in a document library.

Application Processing Services

- Function set 9 contains commands that call another process to perform the tasks that a user requests.

DIA Session Services

- Function set 10 contains the commands necessary to create, change, or delete DIA user-related control variables.

File Transfer Services

- Function set 11 contains the commands necessary to transfer documents between DIA processes.

RESTRICTIONS ON THE AVAILABILITY OF DIA SERVICES

Most applications require restrictions on the ability of users to use Document Interchange Architecture services. Not all users should be able to modify attributes of all documents in a library, to receive every document distributed, or to act on behalf of any other user.

Document Interchange Architecture allows users to specify restrictions on the ability to modify document attributes, to receive documents in a distribution, and to act for another user. This capability to limit users is based on the concepts of document access, user affinity, and passwords.

Document Access

Document Interchange Architecture allows document owners to restrict access to stored documents. The user filing a document in a library can specify the owners of the document and can assign access codes to allow nonowners access to the document. Document Library Services allow or reject a request for access to a document in the document library based on the level of ownership and the access code.

Levels of Ownership

DIA defines two levels of document ownership.

- A primary owner is the user that issues a request to file a document. This owner is identified by an office system node address and a source address or by a sign-on-ID operand.

A primary owner has the following privileges:

- Assigning, modifying, or deleting document access codes
- Removing primary ownership of the document
- Having the authority to obtain access to the documents based on ownership alone
- Modifying the parameters used to search for a document.

- An owner-delegate is a user to whom the primary owner has delegated ownership privileges.

Owner-delegates have the following privileges:

- Rejecting and removing delegated ownership of documents
- Having the authority to obtain access to the document based on document ownership alone
- Modifying the parameters used to search for a document.

Access Codes

Access codes, specified by the primary owner when filing a document, restrict access to the document. Only users whose access code is identical to the one specified by the primary owner can access the document. If a user requests access to a document on behalf of another user, both users must have the access code of the document.

Documents are assigned three levels of availability:

- Public documents

Any requester can obtain access to public documents without document ownership or access codes.

- Private documents

Access to the document is restricted to the primary owner or an owner-delegate.

- Shared-private documents

Access to the document is restricted to primary owners, owner-delegates, and requesters with an access code that matches one of the access codes of the associated document.

Affinity

DIA defines rules for affinity for each DIA category of services. Affinity is a relationship that permits a source or recipient to obtain access to the DIA resources available to another user for the purpose of performing tasks on behalf of that other user.

DIA defines three roles in affinity relationships:

- The principal is the user on whose behalf an action is taken.
- The agent (or surrogate) is the user who is acting on behalf of the principal.
- The temporary agent is a user who performs, on behalf of a principal, a limited number of tasks but who does not have affinity with that principal.

Two lists control the access to DIA services in affinity relationships: an authorization list and an affinity list. An authorization list specifies those users that have authority to act on behalf of a specific source or recipient. An affinity list identifies the sources and recipients for whom a specific user can act.

An affinity relationship exists when both of the following requirements are satisfied:

- The source or recipient (a principal) has a signed-on user listed as a member of the authorization list.
- The signed-on user has the principal as a member of the signed-on user's affinity list.

Each signed-on user can be on more than one authorization list, and each user can be on more than one affinity list.

DIA allows users to supply a source address and a password during sign-on. When requesting a service, a user must supply a source address and a password if these are not provided at sign-on. Temporary agents must use both the source address and the password of the principal on whose behalf they are acting.

Each of the DIA services has default rules for affinity. Refer to the description of each specific category of services to find the default rules for affinity for that category of service.

When there is a contention for the resources of a DIA service, the contention is resolved on a "first-come, first-served" basis.

DIA SESSION SERVICES

DIA processes use DIA Session Services commands to establish a DIA session through which they can exchange documents. DIA processes use the SIGN-ON command to establish a DIA session and negotiate and agree upon the set of DIA services to be performed during the time that the processes are in session. Not all products that implement DIA support the same range of functions. Although DIA offers a wide range of office system functions, few office systems require the full range of implementation choices that are available. Negotiation and agreement between processes on the functions to be used during a session is necessary.

During activation of a DIA session, DIA Session Services identify the users, validate the users' identification, and authorize the activation of a DIA session. DIA Session Services specify accounting information and define the types of documents that the user wishes to receive during this session. DIA Session Services also specify the number of commands within a command sequence that the DIA processes can receive during the session.

DIA processes use the SIGN-OFF command to end a DIA session. When a DIA process sends a SIGN-OFF command, DIA Session Services terminate the session.

When a DIA session is totally within an SNA session, an abnormal termination of the SNA session also terminates the DIA session. An abnormal termination of an SNA session acts as an implied SIGN-OFF command and an implied DIA session reset. The status of the DIA commands at the time the DIA session is terminated can be found by referring to the protocol rules for DIA command classes in Chapter 4, "Request/Reply Protocols."

Both system errors and SNA communications subsystem failures can cause loss of DIUs and reinitialization of DIA processes. DIA handles this condition by specifying that, if an explicit SIGN-OFF command was not received, a subsequent SIGN-ON command received from the same session partner implies an intervening SIGN-OFF command for the previous DIA session.

DOCUMENT LIBRARY SERVICES

Document Library Services store and retrieve documents electronically. These activities are analogous to the manual filing and retrieving of paper documents that take place in most offices.

Document Library Services can also perform activities that are cumbersome in a manual system. For example, searching a large library for a document that meets certain criteria is a difficult task to be performed manually. But using Document Library Services, when a user electronically files a document in a DIA document library, a set of descriptors is filed with it. See "Document Descriptors" on page 21 for more information on descriptors.

Document Library Services use interchange document profiles to search for documents in a document library. For example, a user requests that the office system node (a Document Library Services server) search for all documents about a particular subject, by a certain author, that the library received between any two dates. Upon completing the search, the office system node gives the user a list of the documents that meet the search criteria. Then the user requests that the office system node retrieve a copy of a specific document on the list and deliver it to the user for viewing or printing.

Responsibilities of the Logical Components

A source node attached to Document Library Services provides the following functions for different end users:

- Allows an end user to file a document in the document library or to retrieve or delete documents from the document library.
- Allows an authorized end user, other than the user that filed the document, to retrieve it from the document library.
- Allows an authorized end user to search for and retrieve a document in the library for other end users. For example, a secretary can retrieve documents from the document library for those persons who filed them in the document library.

An office system node providing Document Library Services performs the services that end users request through source nodes. The DIA processes at the office system node:

- File documents in the document library. Document Library Services assigns each document a unique Library-Assigned Document Name (LADN) and returns this name to the requester. The requester uses this Library-Assigned Document Name to uniquely identify the document at a later time.
- Search the descriptors in the document profile of documents in the document library for documents whose descriptors match the user's search criteria and that the end user has authority to obtain.
- Return to the source node a list of all documents meeting the search criteria.
- Deliver copies of documents to the source node that requested the documents.
- Delete ownership of documents upon the request of an authorized end user.

Document Class

DIA allows the definition of a document library to include a provision for filing the classifications of documents along with the documents in the document library. The value specified in the document class parameter can be used as a search argument. When the definition of the library includes user-defined classification categories, the user specifies the class of the document by including a document class profile parameter in the document unit of the FILE command.

The server has the following options for validating the document class:

- If the FILE request includes the document class profile parameter, then:
 - If the specification of the document class is valid, file the document.
 - If the specification of the document class is invalid, then the server must take one of the following actions:
 - File the document under a default classification
 - Reject the FILE request and return a catastrophic exception condition code.
- If the FILE request does not include the document class profile parameter, then the server takes one of the following actions:
 - File the document under a default classification
 - Reject the FILE request and return a catastrophic exception condition code.

If the capability to classify documents is absent from the definition of a document library, and a FILE request specifies a document class profile parameter, then the service considers the value specified to be only a search value.

Search Result List

The Search Result List is a list of references (or pointers) to documents selected by a search of the document library. The selection process of the search uses information that the requester supplied in the search request. The server compares this information with the information in the document search parameters in each document profile. The server enters pointers in the Search Result List. These point to documents for which search request information and document profile information satisfy the comparison.

The requester of the search also provides a search request name to identify the Search Result List. The server qualifies this name with a prefix consisting of the originating office system node address, source address, and the date and time of the search. The requester can then refer to the pointer on the Search Result List to retrieve the document or to request other applications.

Document Library Services preserve the Search Result List until the requester replaces it with a search result list with the same search request name. When the document library is reorganized, the pointers to the documents selected by the

search process are not dynamically changed to the new organization of the document library. Document Library Services invalidates and deletes the Search Result List when the library is reorganized.

Document Descriptors

Document descriptors are user-defined collections of identifier and format values of the parameters of the interchange document profile (IDP) associated with documents in the document library. The IDP contains parameters that identify the contents of the document, such as the name under which it is filed, the author, the subject it covers, and a date associated with the document filed in the DIA library. Document Library Services return document descriptors in a document called a document descriptor document in response to a SEARCH or RETRIEVE command.

The following format example illustrates a document descriptor document returned as the result of a request that only the base subprofile parameters be returned in the document descriptor document.

<u>Field</u>	<u>Length</u>	<u>Value</u>	<u>Name</u>
Length	2	X'002F'	Document descriptor length
ID	2	X'C904'	Document descriptor ID
Format	1	X'01'	Document descriptor Format 1
Length	2	X'002A'	Length of specified subprofile parameters
ID	2	X'CA04'	Base subprofile ID
Format	1	X'01'	Format 1 subprofile
Length	2	X'0015'	Length of parameter
ID	2	X'C700'	Document name ID
Format	1	X'01'	Document name Format 1
Document name	16	C'DIAMEMOA10141980'	User document name
Length	2	X'0007'	Length of parameter
ID	2	X'C706'	Document type ID
Format	1	X'01'	Document type Format 1
Document type	2	X'0002'	Final-form-text
Length	2	X'0009'	Length of parameter
ID	2	X'C701'	Character set ID
Format	1	X'01'	CGCSGID Format 1
Document GCID	4	X'00670100'	Character set 103, Code Page 256

Affinity Relationships of Document Library Services

Figure 8 summarizes affinity default conditions.

Document Library Services Command Operand Usage	Session Principal (Affinity Default)
Conditions: No source address or password	Signed-on user
Source address = signed-on user No password	Signed-on user
Source address \neq signed-on user No source password Source address has affinity with signed-on user	Specified user
Source address \neq signed-on user No source password Source address does not have affinity with signed-on user	This is an exception condition
Source address = signed-on user Valid source password	Signed-on user
Source address \neq signed-on user Valid source password Source address has affinity with signed-on user	Specified user
Source address \neq signed-on user Valid source password No affinity with signed-on user	Specified user
No source address Valid source password	This is an exception condition

Figure 8. Affinity Default Table for Document Library Services

DOCUMENT DISTRIBUTION SERVICES

Document Distribution Services distributes documents from source nodes to recipient nodes within an office system network. Document Distribution Services can distribute documents during a single DIA session (synchronous delivery) or by routing them through office system nodes for subsequent delivery (asynchronous delivery).

When sending a document asynchronously, the sender can request a notification that the document has been delivered to its recipients. When the recipient accepts delivery, this notification, called confirmation-of-delivery (COD), is returned to the sender.

The sender of a document can also specify a priority for the distribution that causes the information to reach the recipient ahead of other information or to reach some recipients faster than it reaches others.

When a recipient node establishes a DIA session with its office system node, it can obtain a summary list of documents that await delivery. The recipient node can either accept or cancel delivery of any of the documents.

Users can send documents to a user-defined distribution list stored at an office system node. Document Distribution Services at the office system node places a copy of the document in the queue of each recipient on the distribution list. Each recipient can request delivery of his individual copy.

DIA assigns a distribution document name that uniquely identifies, within the office system network, each request for a distribution. The OSN uses this name to correlate confirmation-of-delivery (COD) notification and exception condition messages with the corresponding requests for a distribution.

Using Document Distribution Services, source nodes provide the following functions for users:

- Requests the distribution of documents to one or more recipients located in an office system network.
- Categorizes the distributions as priority or non-priority.
- Requests that a confirmation-of-delivery (COD) be returned to the sender of the document when one or more recipients accept delivery.
- Cancels an outstanding confirmation-of-delivery request. (This cancellation affects only the confirmation request; the request to distribute the information remains in effect.)
- Receives document-distribution-related feedback status, such as a notification that the intended recipient is invalid, possibly due to a misspelled recipient address. Feedback status need not be returned or sent during the same DIA session over which the distribution request flowed.

- Specifies that the distribution information is classified as personal. Information so classified requires that the intended recipient supply additional authorization before receiving the distributed information. In this way, a manager can distribute a personal document or message to a group of recipients authorized to receive such material and be assured that only those recipients can receive it.
- Requests distributions on behalf of other end users.

Using Document Distribution Services, recipient nodes provide the following functions for end users:

- Exchange information directly while in a DIA session with the source node.
- Determine what distributed information at the office system is available for delivery.
- Obtain information about the distributions that are ready for delivery at the office system node. (This information can be about all of the distributions or only those distributions that are members of a particular class of service, such as priority, non-priority, or personal.)
- Cancel delivery of the recipient's documents available at the office system node.
- Request delivery of documents on behalf of other end users.

When Document Distribution Services at an office system node distributes documents for asynchronous delivery, the documents remain in storage at the destination office system node until the recipient nodes request delivery.

An office system node operating as an originating node provides the following functions:

- Assigns and returns to the source node a unique distribution document name for each distribution request received.
- Stores the distribution request and the information to be distributed.
- Routes the distribution request and the associated information to the office system nodes, including itself, that serve the specified recipients.
- Maintains a correlation table for a confirmation-of-delivery that is currently outstanding. As each destination node returns a confirmation-of-delivery, the originating node updates the correlation table. When queried by an attached source node, the originating node returns the current confirmation-of-delivery status and information about exception conditions such as recipients that could not be found, due perhaps to a misspelled recipient address.

An office system node operating as a destination node provides the following functions:

- Places distribution requests and documents in a queue until they can be delivered to recipients. The destination node receives a distribution list only

for those recipients that the destination node serves. The destination node places the documents in each recipient's queue.

- Delivers distribution requests and documents to recipient nodes.
- Sends confirmation-of-delivery (COD) to the originating node when the recipient takes delivery of the document.
- Lists the names of the distributions awaiting delivery to recipient nodes. These distributions are contained in the queues at the office system node.
- Cancels delivery of a specified distribution upon request.

Distribution Addressing

For the distribution of documents, DIA logical components must be identifiable by an address. The distribution address contains a field that denotes the recipient and a field that denotes the office system node serving the recipient. The recipient can be an individual, a department, a mailbox, a mailroom, or the location of a predefined distribution list.

The address of the sender of the distribution has the same two-level format described above (a field that denotes the office system node and a field that denotes the source node).

The customer assigns a 1- to 8-byte code, unique to the customer distribution system, to each office system node. These codes, address tokens, are the first part of the distribution address.

Likewise, the customer assigns a 1- to 8-byte token for each user. These address tokens are the second part of the distribution address. They are unique within the office system node that serves the logical entity that the address represents.

Figure 9 illustrates the form of the document distribution address (source address or recipient address).

Office System Node	Source/Recipient
XXXXXXXX	XXXXXXXX

Figure 9. Distribution Address Format

The first level token (office system node) and the end-user (source node or recipient node) define the components of the address that Document Interchange Architecture-based products implement. These two levels are the only limitations that DIA imposes on the structure of addresses. The binding of these two levels is left to product designers. See "SNADS-DIA Network Addressing" in the Transaction Programmer's Guide.

DIA defines the following operands for addressing to be used with all function sets of DIA:

Source/Recipient Operand (End User) Addresses

- Source-Address (SA) Format 1 (1-8 bytes)
- Recipient-Address (RA) Format 1 (1-8 bytes)
- Sign-On-ID (SOID) Format 1 (1-8 bytes)
- Source-Address (SA) Format 42 = (T1.T2 | T1.T2.T4 | T1.T3 | T1.T3.T4)
- Recipient-Address (RA) Format 42 = (T1.T2)
- Sign-On-ID (SOID) Format 42 = (T1.T2.T4 | T1.T3.T4).

Combinations of the T tokens uniquely identify an end user within an OSN. Only the combinations shown above are valid. The contents of the tokens are shown below:

T1 = Domain (1-8 bytes)	A subgrouping of source or recipients within an OSN.
T2 = Name (1-32 bytes)	A source or recipient identifier unique within the domain subgrouping.
T3 = Global Name (1-32 bytes)	Source or recipient identifier unique within the OSN without qualification by the domain identifier.
T4 = Authorization Value (1-32 bytes)	An added level of authorization checking used in conjunction with the other fields.

Office System Node Address Operands

- Originating Node Address (ONA) Format 1 (1-8 bytes)
- Destination Node Address (DNA) Format 1 (1-8 bytes).

Rules for Use of Addressing Forms

The DIA addressing forms are used for interchange according to the following rules:

- The originating node address and destination node address operands are used for office system node identification. The source address and recipient address operands are used for local control and delivery (within an OSN).
- The Format 1 address forms (SOID, SA, RA, ONA, DNA) are the DIA base architecture.
- The Format 42 address forms (SOID, SA, and RA) are an extension to the base.
- All DIA office system node products must support all Format 1 address operands (SOID, SA, RA, ONA, DNA) and may support the Format 42 forms (SOID, SA, and RA).
- The format of the source and recipient addresses must be either all Format 1 or all Format 42 on any one DIA session.
- The T1.T2 form of the source address (Format 42) is used only on returned status information and on the DELIVER command replying to a function set 2 command.
- Connectivity to any office system node product is assured with ONA/DNA and SA/RA Format 1. Source or recipient node products must negotiate support of the source address or recipient address Format 42 operands by specifying Sign-On-ID Format 42 operands on the sign-on request. The sign-on reply contains a Sign-On-ID of Format 1, but all source address and recipient address forms sent during the DIA session must be the same format as the Sign-On-ID specified on the sign-on request. Format 42 source address or recipient address forms are used for input to tables for translation to Format 1.

Distribution Affinity

The following figures describe the use of the recipient address and recipient password and of the source address and source password when issuing Document Distribution Services commands.

LIST and OBTAIN Command Command Operand Usage	Session Principal (Affinity Default)	Session Principal (Affinity Specified)
Conditions: No recipient address or password	Signed-on recipient and recipients with affinity to the signed-on recipient	Signed-on recipient and recipients with affinity to the signed-on recipient
Recipient address = signed-on user No recipient password	Signed-on recipient	Signed-on recipient and recipients with affinity to the signed-on recipient
Recipient address != signed-on user No recipient password Recipient address has affinity with signed-on user	Specified recipient	This is an exception condition
Recipient address != signed-on user No recipient password Recipient address does not have affinity with signed-on user	This is an exception condition	This is an exception condition
Recipient address = signed-on user Valid recipient password	Signed-on recipient	Signed-on recipient and recipients with affinity to the signed-on recipient
Recipient address != signed-on user Valid recipient password Recipient address has affinity with signed-on user	Specified recipient	Specified recipient and recipients with affinity to the specified recipient
Recipient address != signed-on user Valid recipient password No affinity with Signed-on user	Specified recipient	Specified recipient and recipients with affinity to the specified recipient
No recipient address Valid recipient password	This is an exception condition	This is an exception condition

Figure 10. Affinity Default for LIST and OBTAIN Processing

CANCEL-DISTRIBUTION Command Command Operand Usage	Session Principal (Affinity Default)
Conditions:	
No recipient address or password	Signed-on recipient
Recipient address = signed-on user No recipient password	Signed-on recipient
Recipient address \neq signed-on user No recipient password Recipient address has affinity with signed-on user	Specified recipient
Recipient address \neq signed-on user No recipient password Recipient address does not have affinity with signed-on user	This is an exception condition
Recipient address = signed-on user Valid recipient password	Signed-on recipient
Recipient address \neq signed-on user Valid recipient password Recipient address has affinity with signed-on user	Specified recipient
Recipient address \neq signed-on user Valid recipient password No affinity with signed-on user	Specified recipient
No recipient address Valid recipient password	This is an exception condition

Figure 11. Affinity Default for Canceling a Distribution

CANCEL-DISTRIBUTION Command Command Operand Usage	Session Principal (Affinity Default)
Conditions:	
No source address or password	Signed-on source
Source address = signed-on user No source password	Signed-on source
Source address \neq signed-on user No source password Source address has affinity with signed-on user	Specified source
Source address \neq signed-on user No source password Source address does not have affinity with signed-on user	This is an exception condition
Source address = signed-on user Valid source password	Signed-on source
Source address \neq signed-on user Valid source password Source address has affinity with signed-on user	Specified source
Source address \neq signed-on user Valid source password No affinity with signed-on user	Specified source
No source address Valid source password	This is an exception condition

Figure 12. Affinity Default for Canceling Status

REQUEST-DISTRIBUTION Command Command Operand Usage	Session Principal (Affinity Default)
Conditions:	
No source address or password	Signed-on source
Source address = signed-on user No source password	Signed-on source
Source address != signed-on user No source password Source address has affinity with signed-on user	Specified source
Source address != signed-on user No source password Source address does not have affinity with signed-on user	This is an exception condition
Source address = signed-on user Valid source password	Signed-on source
Source address != signed-on user Valid source password Source address has affinity with signed-on user	Specified source
Source Address != signed-on user Valid source password No affinity with signed-on user	Specified source
No source address Valid source password	This is an exception condition

Figure 13. Affinity Default for REQUEST-DISTRIBUTION

FILE TRANSFER SERVICES

File Transfer Services define commands that allow DIA processes to file and retrieve documents in a user-specified library.

Responsibilities of the Logical Components

A requesting node attached to File Transfer Services provides the following functions for different end users:

- Allows an end user to file a document in a specified end-user library.
- Allows an end user to retrieve documents from an end-user library.

A server node providing File Transfer Services performs the following services for end users:

- File documents into the specified library as requested.
- Delivers copies of the documents to the requesters.

Note: The File Transfer Services commands are a subset of Document Library Services commands. Whereas Document Library Services was architected for host (archive) libraries, File Transfer Services was designed for private libraries. Only private documents in private libraries are supported. (Affinity relationships are not supported.)

APPLICATION PROCESSING SERVICES

Application Processing Services defines commands that cause an office system node to perform additional functions. These additional functions allow end users to:

- Modify the document profile associated with a document (for example, by adding or deleting the descriptors).
- Invoke a program that transforms documents from one type of document to another type. See Appendix C for document types.
- Invoke specific application programs, procedures, or processes.

A source node using Application Processing Services provides the following functions for end users:

- Requests the execution of programs that are stored at the office system node
- Requests the modification of the descriptors in a document profile
- Requests that the programs that transform and translate documents be invoked.

An office system node at which Application Processing Services are located provides the following functions:

- Interprets and validates requests from the source nodes
- Modifies descriptors in the document profile as specified by end users
- Schedules execution of programs and procedures requested by the end user.

Affinity Relationships

Figure 14 summarizes the default conditions for affinity in Application Processing Services.

Application Processing Services Command Operand Usage	Session Principal (Affinity Default)
Conditions:	
No source address or password	Signed-on source
Source address = signed-on user No source password	Signed-on source
Source address \neq signed-on user No source password Source address has affinity with signed-on user	Specified source
Source address \neq signed-on user No source password Source address does not have affinity with signed-on user	This is an exception condition
Source address = signed-on user Valid source password	Signed-on source
Source address \neq signed-on user Valid source password Source address has affinity with signed-on user	Specified source
Source address \neq signed-on user Valid source password No affinity with signed-on user	Specified source
No source address Valid source password	This is an exception condition

Figure 14. Affinity Default for Application Processing Services

CHAPTER 4. REQUEST/REPLY PROTOCOLS

This chapter describes how DIA processes use DIA commands to exchange information.

GENERAL RULES

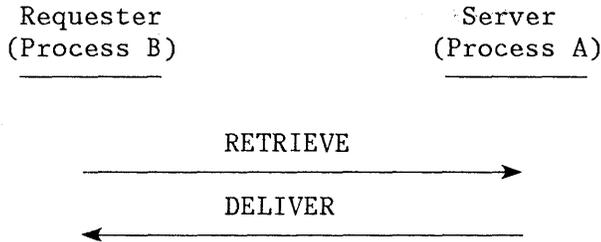
Commands drive the flow of DIUs between two DIA processes. Typical commands distribute a document from one office system to other office systems or retrieve a document from the document library and return it to the requester.

The rules for sequencing the flow of DIUs between two DIA processes comply with the request/reply protocols that are specified for each command that DIA defines. The following general rules apply to issuing commands within a DIA session:

- Only one DIA process can be sending command requests at any one time.
- The DIA process that sends the SIGN-ON request is the process that sends the first command in the DIA session.
- When an exception condition is encountered in the processing of a command, the service returns an ACKNOWLEDGE command with a code for an exception condition as a reply.
- When no exception condition is encountered in the processing of a command and that command requires a reply, a positive reply is returned.

The mapping of DIA request/reply protocols to the SNA LU 6.2 protocol boundary and to the SNADS protocol boundary is described in the Transaction Programmer's Guide. The mapping of DIA request/reply protocols to other communication transport mechanisms is described in the publications for the IBM products that implement DIA.

To illustrate the DIA request/reply command protocol, consider the following scenario: A requester (process B) sends a request to a server (process A) to retrieve a document from the server's (process A's) document library. The server (process A) interprets the request, retrieves the document from the library, and delivers the document to the requester (process B).



The above figure illustrates this basic DIA request/reply protocol. The RETRIEVE command is the request for the function (retrieving the document), and DELIVER is the reply to the request for the RETRIEVE function. Notice that the server (process A) responds to the requester (process B) on demand. A protocol that requires a service reply is one class of request/reply protocol defined by DIA: a Synchronous Reply Required (SRR) protocol.

A requester uses a command to request that the server perform a unit of work. To invoke a service function, the requester must specify the command to be performed, any associated input data, and the request/reply protocol to be used. The command class specifies the request/reply protocol. In the above example, a synchronous reply is required. The function server responds to the function request according to the class of the command specifying the request. The results of the request for the function are returned with a DIA command. The replying command also specifies the request/reply protocol of the service that processes the function. A request for a function can result in the return of multiple replies.

DIA COMMAND CLASSES

DIA defines the following command classes:

- No Reply Required Command Class (NRR)

The requester uses the NRR command class when the function requested does not require a reply.

- Synchronous Reply Required Command Class (SRR)

The requester uses the SRR command class when the function requested is to be performed synchronously (immediately) by the server and the results returned with one or more replying commands. The example on the previous page is an SRR request/reply scenario.

- Asynchronous Reply Required Command Class (ARR)

The requester uses the ARR command class when the function requested need not be performed synchronously but can be performed any time at the server's convenience. The results are returned to the requester with one or more replying commands some time later, possibly in any order.

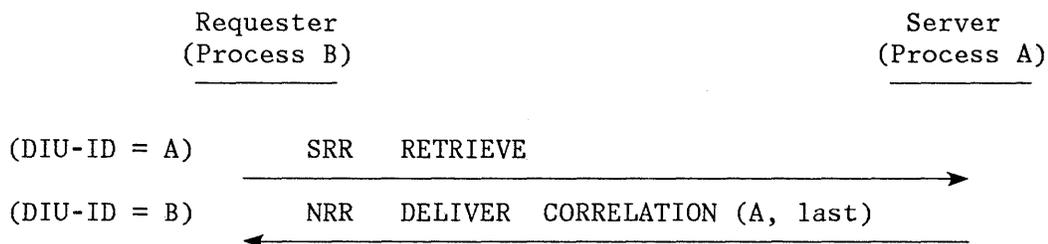
In general, these command classes could be used with any DIA command. However, DIA defines its current services using a subset of these command classes for each service command. These services are called function sets and are defined at the beginning of chapters 7 through 11.

DIA REQUEST/REPLY CORRELATION

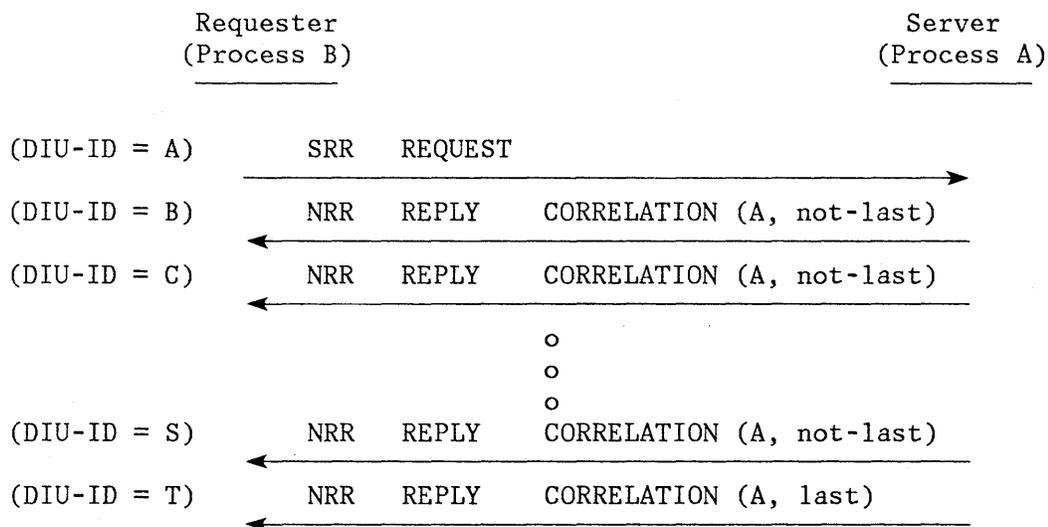
A document interchange unit contains either a request or a reply when it is exchanged between DIA processes. You can uniquely identify each DIU by a DIU identifier (DIU-ID) contained in the DIU prefix.

Replies are correlated with a previously received request (command) by referring to the DIU identifier of the function request and indicating whether the reply is the last or not-last. Each reply must have a CORRELATION operand to correlate the reply to the function request to which it is replying. A description of the CORRELATION operand is in Appendix A.

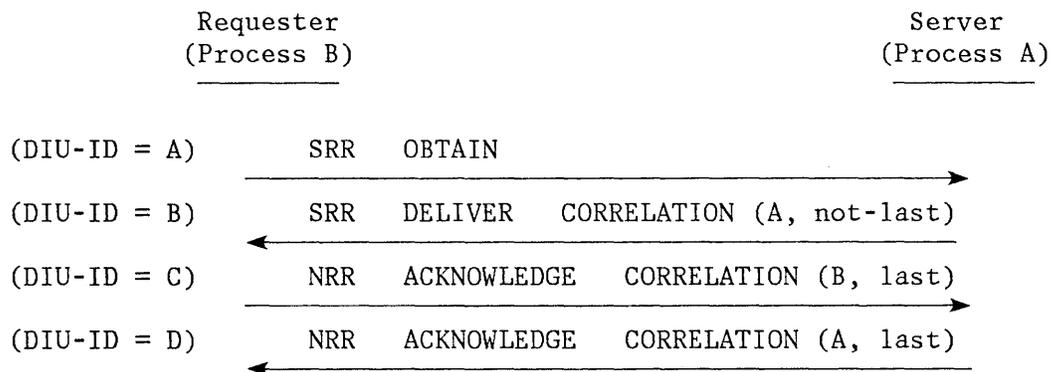
The following scenario illustrates the DIA request/reply correlation. Assume the RETRIEVE DIU prefix has a DIU-ID = A and the DELIVER DIU prefix has a DIU-ID = B. Then the replying DELIVER command can be unambiguously correlated to the RETRIEVE request by using the CORRELATION operand in the reply specifying the DIU-ID of the RETRIEVE command and indicating this is the last replying command.



Multiple replying commands to a request can be accomplished by indicating in the CORRELATION operand that each reply is not last until the last reply is sent. The last replying command has a CORRELATION operand that indicates it is the last replying command. For example:



The request/reply protocol outlined can also be used for replies to replying commands. For example:



In this example, the DELIVER command is not the last replying command to the OBTAIN command. The replying DELIVER command requests that an acknowledgement of data delivery be returned. When the data has been received, an ACKNOWLEDGE replying command is returned to the DELIVER command. When the service processing the OBTAIN command has delivered all the information requested, or that can be delivered at this time, it sends an ACKNOWLEDGE command as the last reply to the OBTAIN command.

DIA COMMAND CLASS PROTOCOL RULES

The following protocol rules apply:

- NRR - No Reply Required Command Class
 - Replying commands sent to NRR requests need not be synchronized or correlated.
 - An ACKNOWLEDGE command with an exception condition code is allowed to reply to and refer to this class of command. The exception condition information is for a statistical record or for a problem determination logging.
- SRR - Synchronous Reply Required Command Class
 - The SRR command requester cannot send another function request until the last reply has been returned. The reply can be any command that has a CORRELATION operand correlating it with the SRR request. The SRR command class can be used as a reply.
- ARR - Asynchronous Reply Required Command Class

The ARR command class is used for any command that requires a reply with the following provisions:

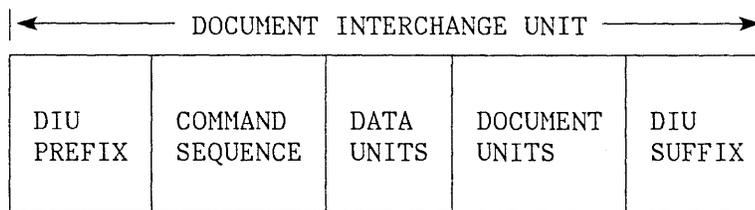
- The reply need not be the next command sent by the receiver.
- The reply need not be sent during the current DIA session.
- Replies can be sent in any order.
- Replies need not be received in the order sent by the ARR request processor.
- Termination of the DIA session while a reply to an ARR command is outstanding does not affect either the ARR command or the reply to that command.

CHAPTER 5. DOCUMENT INTERCHANGE UNIT

This chapter defines the format and content of a document interchange unit (DIU). Also defined in this chapter is the DIA self-defining, variable-length, structured-field data stream. The chapter concludes with a summary of the DIU syntax, parsing, and character set rules.

DOCUMENT INTERCHANGE UNIT (DIU) COMPONENTS

The basic unit of information exchanged between DIA processes is the document interchange unit (DIU). A DIU is made up of the following data stream components:



- The prefix contains the information to introduce and identify the DIU.
- The command sequence contains the command that specifies the function to be performed and related processing information. The command sequence contains from 1 to 255 commands.
- The data unit contains information that may be referred to by one or more DIA commands in the command sequence. This field is optional and is present when a command so designates. A DIU contains from zero to 255 data units.
- A document unit contains the document profile and optionally the document content. This field is optional and is present only when a document profile and content are sent from one DIA process to another. A DIU contains from zero to 255 document units.
- A suffix specifies the end of the DIU and reports whether any abnormal conditions occurred while the DIU was being transmitted.

These data stream components can include subcomponents, such as command operands and document profiles. Each DIU component begins with a structured field called an introducer. The introducer uniquely identifies each field and specifies its length. Consequently, all fields and components (and hence, the entire data stream) are self describing and variable in length.

DIU STRUCTURED FIELD

The DIU structured field consists of five parts: the total length (LL) of the structured field, the structured-field identifier (ID), the format (F), an optional structured-field identifier extension (ISS), and an optional data variable as illustrated in Figure 15.

The structured-field data variable can consist of scalar data values, fixed-formatted data vectors, or self-defining fields of any form. For example, the data variable can contain other structured fields (LLIDFs) or self-defining, variable-length data fields where the variable length data is preceded by a self-defining LT introducer, the 1-byte L field defines the length of the LT introducer and the variable length data, and the 1-byte T field defines the type of self-defining field.

DIU Introducer

The LLIDF parts of the DIU structured field are called the DIU introducer. The optional ISS part of the DIU structured field is called the DIU introducer extension. Each DIU data stream component contains an introducer. However, introducer extensions are permitted only on the major DIU data stream components: prefix, commands, data units, document units, and suffix.

The DIU structured field is defined as follows:

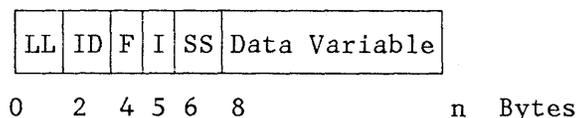


Figure 15. DIU Structured Field

LL = Structured-field length

The length LL can vary from 5 to 32767 bytes, including the LLIDF(ISS) and structured-field data variable.

ID = Structured-field identifier

The ID consists of two parts: a class identifier and a type identifier, where:

I = class (for example, prefix, command class, operand class)

D = type (for example, type of command, type of data unit).

F = Format byte

The format byte defines the format of the data variable and indicates whether the optional introducer extension (ISS) is present or not.

DIA defines the F byte as follows:

- Bit 0 - Introducer Extension Indicator
1 = ISS is present and follows the introducer (LLIDF)
0 = No ISS is present
- Bit 1 - Imbedded Structure Indicator
1 = Data variable format is in LT format
0 = Data variable is defined by bits 4-7
- Bits 2 - 3 Reserved
- Bits 4 - 7 Data Variable Format Indicator
A 4-bit binary number specifying the format and syntax of the data variable.

The F bytes for formats 1, 2, 3, 41, and 42 are as follows:

	Bit							
	0	1	2	3	4	5	6	7
Format 1	B	0	R	R	0	0	0	1
Format 2	B	0	R	R	0	0	1	0
Format 3	B	0	R	R	0	0	1	1
Format 41	B	1	R	R	0	0	0	1
Format 42	B	1	R	R	0	0	1	0

where B is a binary number (0 or 1)
and R is a reserved bit (value must be zero)

The IDF values of the DIU structured fields are defined in Appendix B.

DIU Introducer Extension (ISS)

The ISS portion of the structured field is defined as follows:

I = Indicator byte

The indicator byte identifies the structure of the construct.

Bits 0 - 1 Reserved
Bit 2 - Segmentation Indicator
 1 = Not last, a structured field segment follows
 0 = Last or only structured field segment
Bits 3 - 7 Reserved

SS = Sequence number - X'0000'

Setting the high-order bit of the F byte to one indicates the presence of an ISS introducer extension in a structured field. If the ISS introducer extension is omitted, the high-order bit of the F byte is set to zero.

DIU commands, data units, and document units can be segmented to accommodate DIA processes with limited buffer sizes or data lengths greater than 32767 bytes. These data stream components can also be segmented when the total length of the data is not known when the introducer is generated. Structured-field segmentation is defined and controlled through use of indicators in the I byte, and is discussed in detail in "Structured-Field Segmentation."

Structured-Field Documentation Conventions

The documentation conventions used in this book for DIU structured-field LLIDF, LLIDF(ISS), LLIDFISS notations are:

- The LLIDF notation describes a structured field where the ISS extension is not permitted.
- The LLIDF(ISS) notation describes a structured field where the ISS extension is optional.
- The LLIDFISS notation describes a structured field where the ISS extension must be present.

DOCUMENT INTERCHANGE UNIT (DIU) STRUCTURE

Figure 16 illustrates the components of a DIU data stream. The DIU data stream components are the prefix, command sequence (command), data units, document units, and suffix. All DIU data stream components have LLIDF structured-field introducers. DIU introducer extensions (ISS) are permitted on DIU data stream components only; specifically, prefix, commands, data units, document units, and suffix.

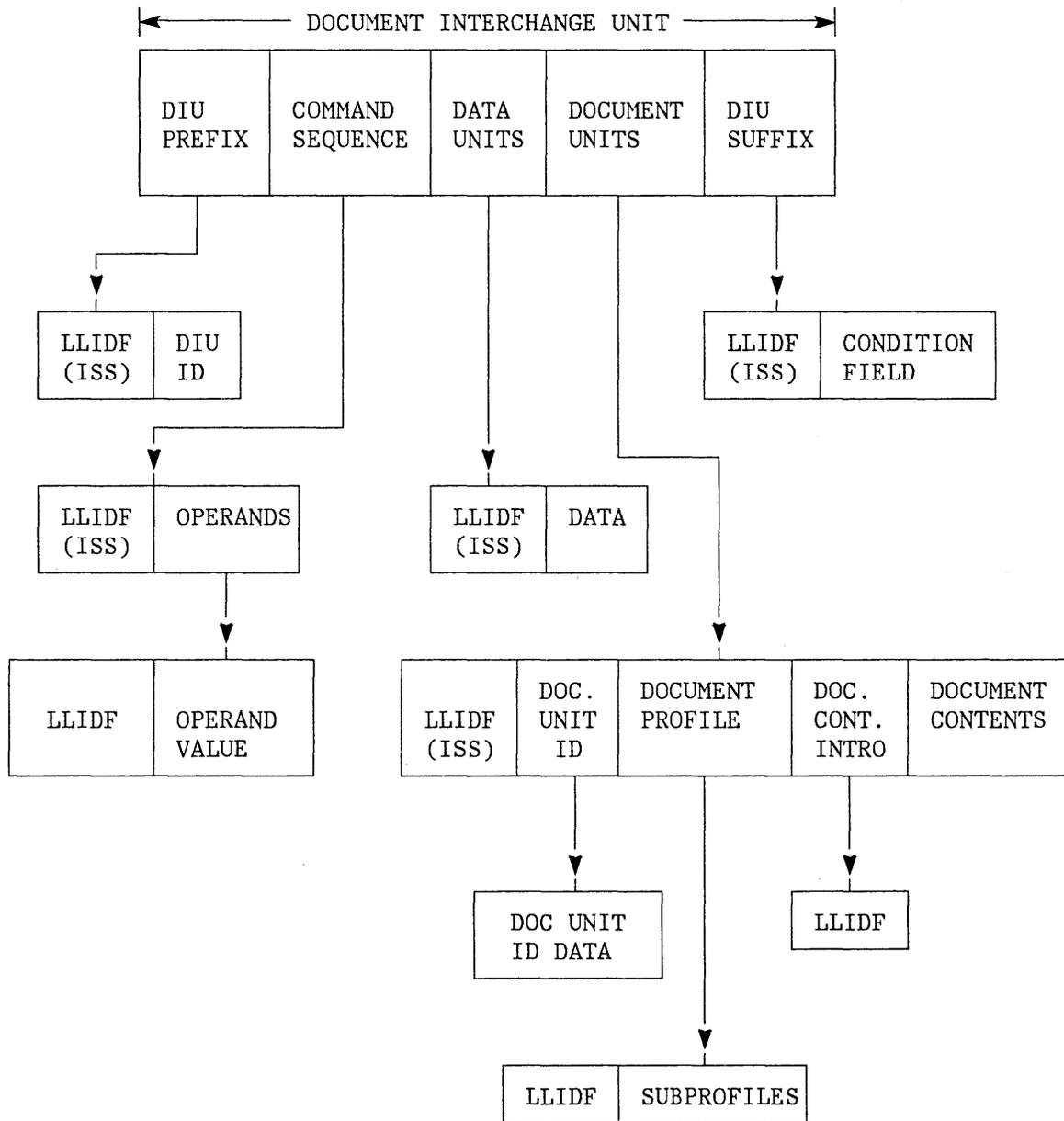


Figure 16. Document Interchange Unit Structure Overview

DIU Prefix

The DIU prefix begins and identifies the DIU. The DIU prefix consists of its structured-field introducer LLIDF(ISS) and an optional data variable called the DIU identifier (DIU-ID).

The DIU-ID field is used to uniquely identify the DIU. The DIU-ID field is also used to correlate replying commands to a previously received function request (a DIA command). The DIA command request/reply protocol, including command correlation, is defined in Chapter 4, "Request/Reply Protocols."

The DIU-ID field is a 0- to 16-byte user-supplied value and can be omitted when the function request does not require a replying command.

The prefix structured-field ID class byte (I) defines the data stream component as a DIU prefix; the type byte (D) defines the architected version of this DIU.

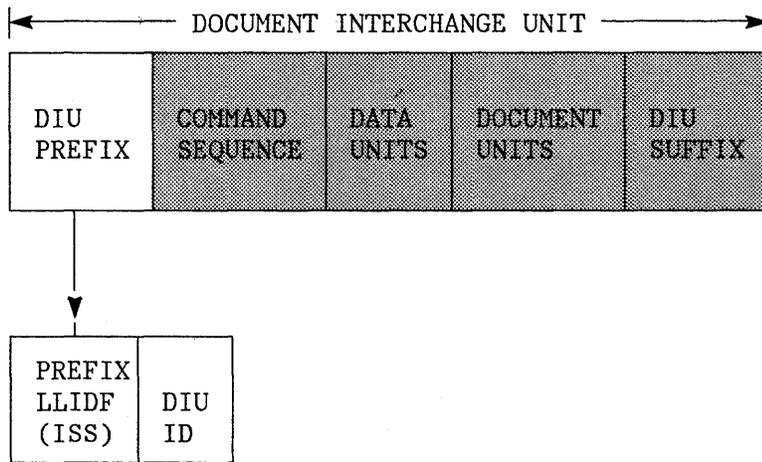


Figure 17. DIU Prefix

Command Sequence

The DIU command sequence consists of 1 to 255 commands. Each command defines a unit of work to be performed by the DIU receiver. Execution of commands within the command sequence is the responsibility of the DIU receiver. The functions requested by the commands must be performed in the order specified in the command sequence.

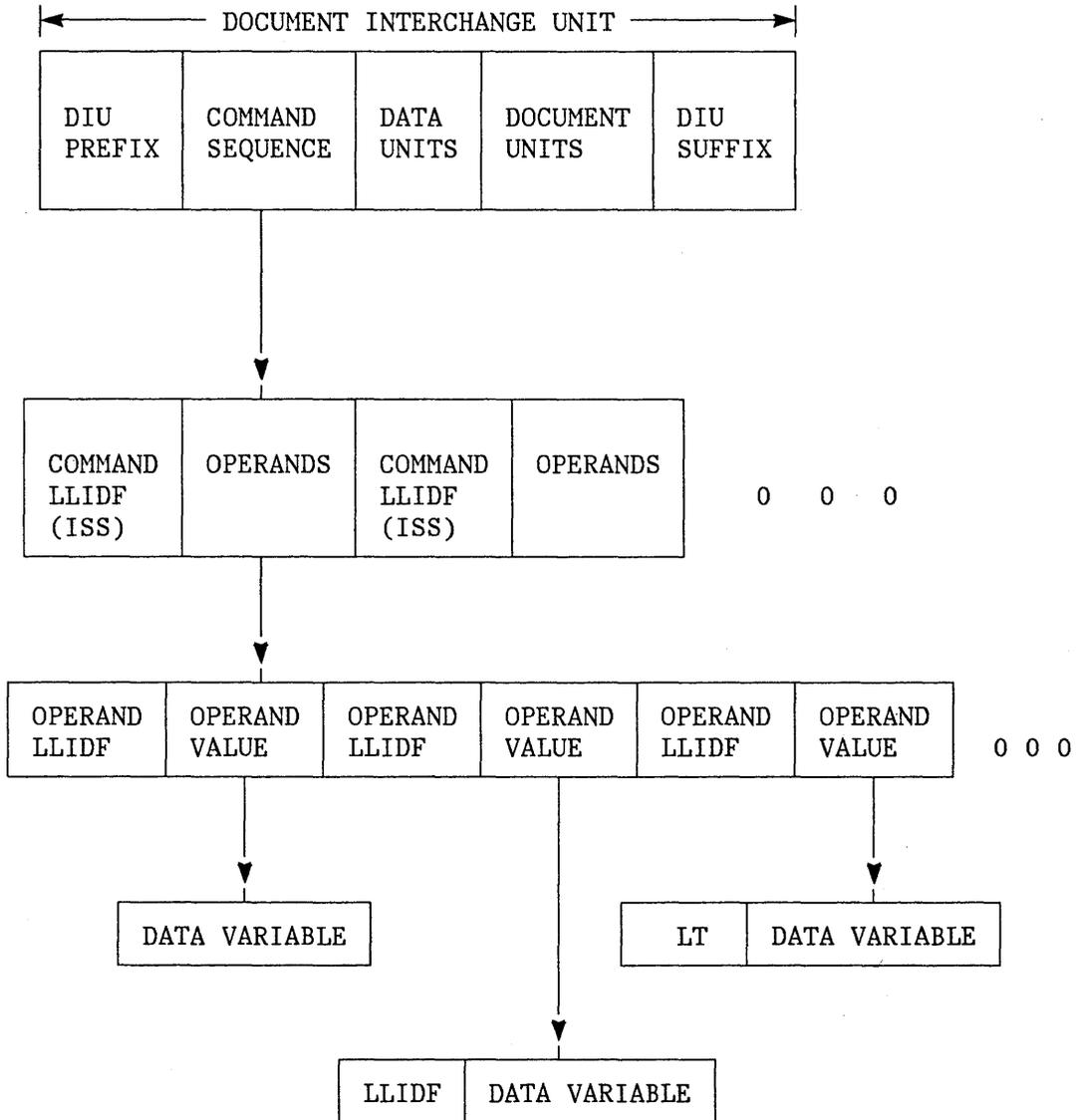


Figure 18. DIU Command Sequence

Command

Each command consists of its structured-field introducer LLIDF(ISS) and zero or more operands. The command operands are totally contained in the command's structured-field data variable; hence, the length (LL) of the command's structured field includes the command introducer LLIDF(ISS) and all command operands. Each DIA command is uniquely identified by its structured-field IDF introducer.

The 2-byte command structured-field identifier (ID) defines the command class and the specific command type (for example, a FILE command or a DELIVER command). The introducer format byte (F) defines the format and structure of the data variable. In this case, the format byte defines the required and optional operands and the order of the operands within the data variable, if any, that this command requires. The DIA command classes are defined in Chapter 4, "Request/Reply Protocols." The specific DIA commands are defined in Chapters 7 through 11. Commands can be segmented. A description of structured-field segmentation is found in "Structured-Field Segmentation."

Operands

Operands either contain or refer to data that is used in the execution of a command. Operands are uniquely defined by their structured-field introducer LLIDF. Operand data is contained in the data variable part of the operand structured field; hence, the length (LL) of the operand structured field includes the operand introducer LLIDF and the operand data.

The 2-byte operand structured-field identifier defines the operand class and operand type. The operand class defines the location of the operand data. The operand classes defined are:

- Immediate Data - the operand data is located entirely within the operand's structured-field data variable.
- Data Unit Reference - the operand data is located in a data unit component of the DIU. The operand data variable contains a relative pointer to the data unit. The format byte defines the format of the relative pointer.
- Document Unit Reference - the operand data is located in a document unit component of the DIU. The operand data variable contains a relative pointer to the document unit. The format byte defines the format of the relative pointer.

The operand type defines the specific type of operand, such as a RECIPIENT-ADDRESS operand, or a CORRELATION operand.

The operand introducer format byte (F) defines the format and structure of the data variable. For immediate data operands, the operand data variable may consist of scalar data values, fixed-format data vectors, or self-defining fields. For example, the data variable may contain other structured fields (LLIDFs) or self-defining, variable length data fields where the variable length data is preceded by a self-defining LT introducer; the 1-byte L field defines the length the LT introducer and the variable length data; and the 1-byte T field defines the type of self-defining field within that operand. If bit 1 of the F byte is set to one, the data variable contains only self-defining, variable-length LT data fields.

For data unit or document unit reference operand classes, the format byte is followed by a 1-byte binary number in the range from 1 to 255 whose value corresponds to the relative occurrence within the DIU of the data unit or document unit being referred to.

Data Units

A data unit contains information that is referred to by one or more commands in the command sequence.

Data units contain operand data that is used in the execution of a command. Data units are uniquely defined by their structured-field introducer LLIDF(ISS). The data unit structured-field data variable contains the operand data; hence, the length (LL) of the data unit structured field includes the data unit introducer LLIDF(ISS) and the data variable.

Operands within the command sequence refer to data units. The format of the data variable within the data unit corresponds to the format of the immediate data type form of the operand that refers to the data unit. Data units can be segmented. A description of structured-field segmentation is found in "Structured-Field Segmentation."

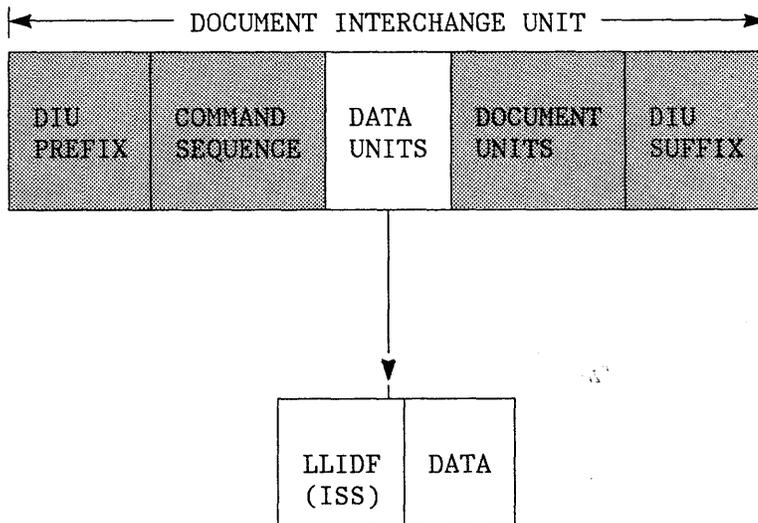


Figure 19. DIU Data Unit

Document Units

The document unit contains a document unit identifier field, an optional document profile, a document content introducer, and optionally, the document content itself as shown in Figure 20.

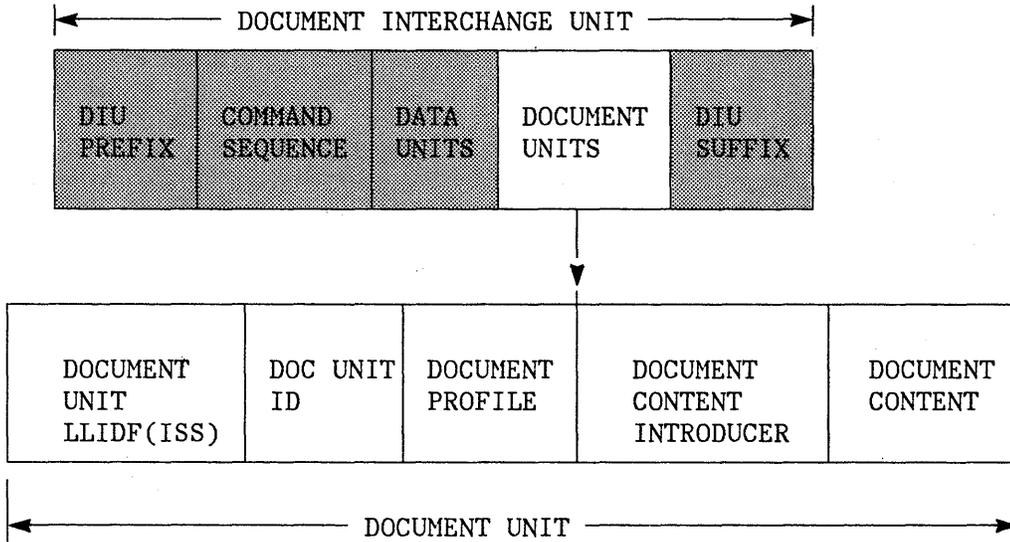


Figure 20. DIA Document Unit Type 3

The document unit is uniquely defined by its structured-field introducer LLIDF(ISS). For the DIA interchange Document Unit Type 3 (IDF - interchange document unit class and type; X'C90301'), the document unit structured-field data variable consists of a document-unit-identifier field, an optional document profile, a document content introducer, and optionally, the document content; hence, the length (LL) of the document unit structured field includes the document unit introducer and all entities within the data variable. Document unit structured fields may be segmented. A description of structured-field segmentation can be found in "Structured-Field Segmentation."

Document Unit Identifier

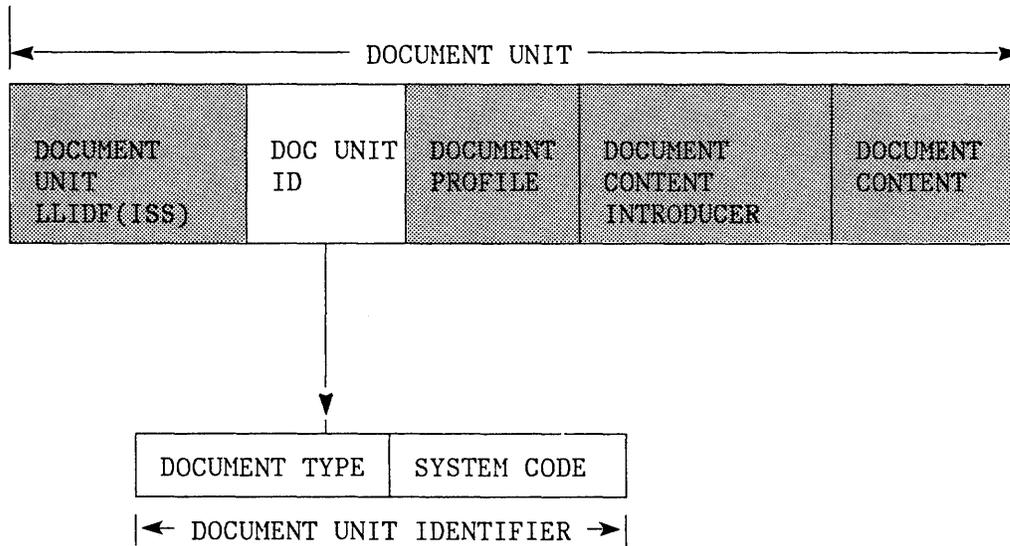


Figure 21. DIU Document Unit Identifier

The document unit identifier for the interchange Document Unit Type 3 is a 15-byte, fixed-length positional data field that immediately follows the document unit structured-field introducer LLIDF(15S). The document unit identifier consists of two parts and is defined as follows:

- The Document Type field is a 2-byte binary number that identifies the type of data contained in the Document Content field of the document unit.
 - Document Type field values in the range of 0 to 32767 specify the interchange data stream; for example, a Document Content Architecture Revisable-Form-Text document or a Document Content Architecture Final-Form-Text document. Document Type fields are registered by IBM. The interchange data streams are listed in Appendix C.
 - Document Type field values from 32768 to 65535 specify noninterchange data stream identifiers (for example, core image module, program temporary fix, and so on) and depend on the value of the system code parameter for their meaning. The product identified in the system code parameter controls the noninterchange data stream identifiers. The identifiers can be assigned in any manner that satisfies that product's requirements.
- The System Code field is a 13-byte alphanumeric name that identifies the product that created this document unit. The System Code field can contain a registered IBM system identifier or a customer-assigned identifier. IBM-registered identifiers begin with the characters IBM.

Document Profile

A document profile contains information relating to or describing a document. All information in a document profile applies to the entire document. The document profile is shown in Figure 22.

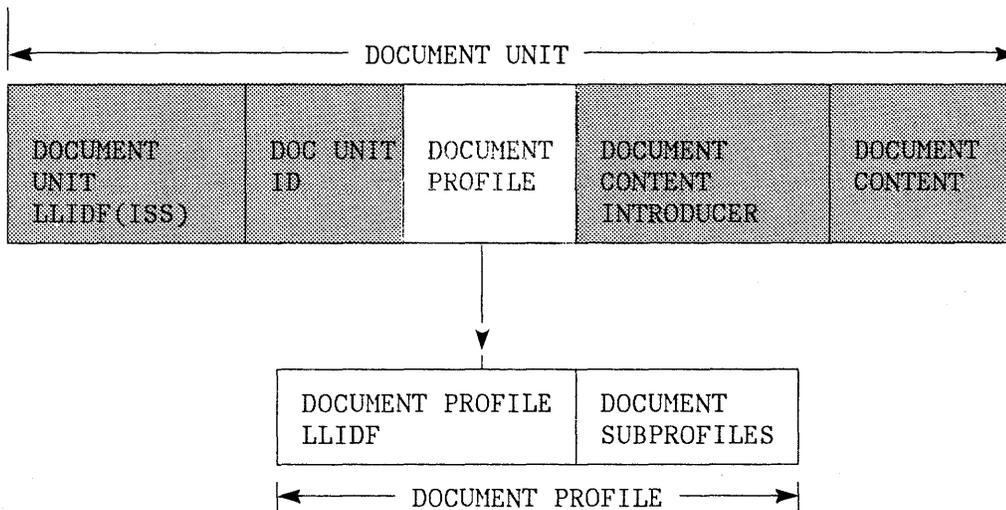


Figure 22. DIU Document Profile

The document profile is defined by a structured-field introducer, LLIDF. The structured-field data variable consists of one or more document subprofiles; hence, the length of the document profile structured field includes the introducer LLIDF and all document subprofiles.

The interchange document profile, which has an assigned IDF value of X'CA0301', is used within the Interchange Document Unit Type 3 (IDF X'C90301') to exchange information between DIA processes. The internal structure and syntax of the interchange document profile and subprofiles are defined in Interchange Document Profile.

Document Content Introducer

The document content introducer defines the end of the document profile and the start of the document content within the document unit.

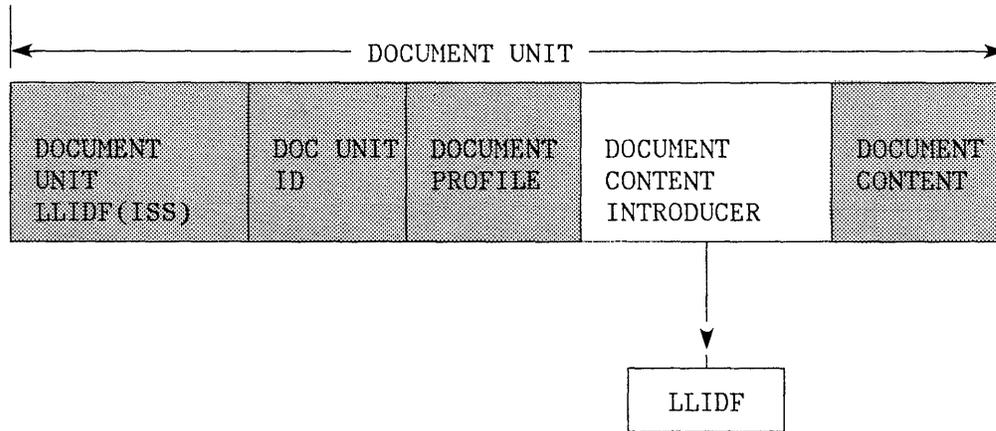


Figure 23. DIU Document Content Introducer

The document content introducer consists of a structured-field introducer LLIDF with no data variable. The length field (LL) contains X'0005' to indicate the length of the LLIDF introducer. The class and type bytes of the IDF field identify this structured field as a document content introducer of a specific type. Two types of the document content introducers are defined:

- Document Content Introducer Type 1 (IDF X'CB0101') specifies that the document content immediately follows the document content introducer LLIDF.
- Document Content Introducer Type 2 (IDF X'CB0201') specifies that no document content follows the document content introducer LLIDF.

DIU Suffix

The DIU suffix specifies the end of a DIU and indicates whether any abnormal conditions occurred while the DIU was being transmitted.

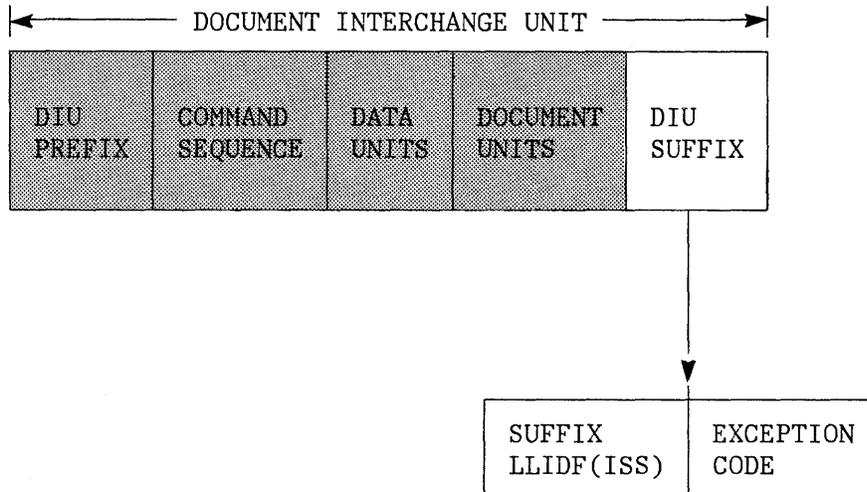


Figure 24. DIU Suffix

The suffix consists of a structured-field introducer LLIDF and an optional data variable, which is the exception code field. The class and type bytes of the IDF field identify this structured field as a suffix of a specific type. The following suffix types are defined:

- Type 1 Suffix - Normal termination of a DIU
- Type 2 Suffix - Abnormal termination of a DIU

If the DIU is terminated normally (Type 1 Suffix), the exception condition code data variable is omitted.

If the DIU has terminated abnormally (Type 2 Suffix), the exception condition code data variable contains an exception condition code that describes the DIU sender-detected error that caused the termination. The format of the exception condition code is defined in Chapter 6, "Exception Detection, Classification, and Reporting." The Type 2 Suffix must be sent at a DIU component boundary or a DIU component segment boundary (last or not last), such as after the command sequence.

The receiver of a Type 2 Suffix:

- Responds to the Type 2 Suffix by sending an ACKNOWLEDGE command with a normal completion condition code.
- Is not required to send an ACKNOWLEDGE command when a Type 2 Suffix is received on an NRR replying command, but may choose to do so. The NRR replying command with a Type 2 Suffix is equivalent to an ACKNOWLEDGE command with an exception condition code. See Figure 25 for additional information.

The sender of a Type 2 Suffix:

- Requires that an ACKNOWLEDGE command be received after sending a Type 2 Suffix on other than NRR replying commands. An ACKNOWLEDGE command with an exception condition code indicates that the receiver detected an error before processing the Type 2 Suffix.
- Might receive an ACKNOWLEDGE command after sending an NRR replying command with a Type 2 Suffix. See Figure 25 for additional information.

The Type 2 Suffix is used to specify that the detected exception condition was so severe that the sender will not attempt to recover that DIU. Figure 25 summarizes the Type 2 Suffix processing.

DIU with a Type 2 Suffix	Receiver of a Type 2 Suffix	Sender of a Type 2 Suffix
SRR Request	Sends an ACKNOWLEDGE. Ignores the SRR request.	Receives an ACKNOWLEDGE. The next DIA request may have no relevance to the abnormally-terminated DIU.
SRR Reply	Sends an ACKNOWLEDGE. Ignores the SRR reply.	Receives an ACKNOWLEDGE. The next DIU to be sent must contain the DIA reply.
NRR Reply to SRR reply	Sends an ACKNOWLEDGE. The next DIU to be received must be the DIA reply.	Receives an ACKNOWLEDGE. The next DIU to be sent must contain the DIA reply.
NRR Reply to SRR request	An ACKNOWLEDGE is permitted, but need not be sent. The SRR request associated with the NRR reply is terminated.	The NRR reply will terminate the SRR request.

Figure 25. Type 2 Suffix Processing Summary

STRUCTURED-FIELD SEGMENTATION

A structured field can be segmented to accommodate (1) structured fields whose length exceeds 32767 bytes, (2) DIA processes with limited buffer space, or (3) situations where the length of the structured field cannot be determined before the structured field must be transmitted. The entire structured field must be segmented if segmentation is performed at all. The last segment can contain a null data variable.

The technique defined to segment a structured field allows the structured-field data variable to be subdivided into smaller pieces called segments. Each data variable segment is preceded by the structured-field introducer LLIDF of the data variable that is being segmented. Segmentation of a structured field requires the inclusion of the introducer extension ISS; hence, the structured-field introducer for each structured-field segment is of the form LLIDFISS.

The structured-field introducer extension ISS is used to indicate the last, only, or a not-last segment through the use of the segmentation indicator. The I byte of the introducer extension defines the segmentation indicator. This technique of segmenting a structured field requires that the last segment be coded with the last or only segmentation indicator, and that all previous segments are coded not last as illustrated in Figure 26. The introducer extension ISS can be included with a structured field that is not a segment. However, the segmentation indicator must be set to indicate last or only segment.

The DIA process that generates the structured field performs its segmentation. Segmentation of the data variable is independent of the content of the structured-field data variable. The DIA process receiving the segmented structured field is responsible for reconstructing the data variable before the data is used for processing.

The encoding of the segmentation indicator bit in the introducer extension is described in "DIU Introducer Extension (ISS)."

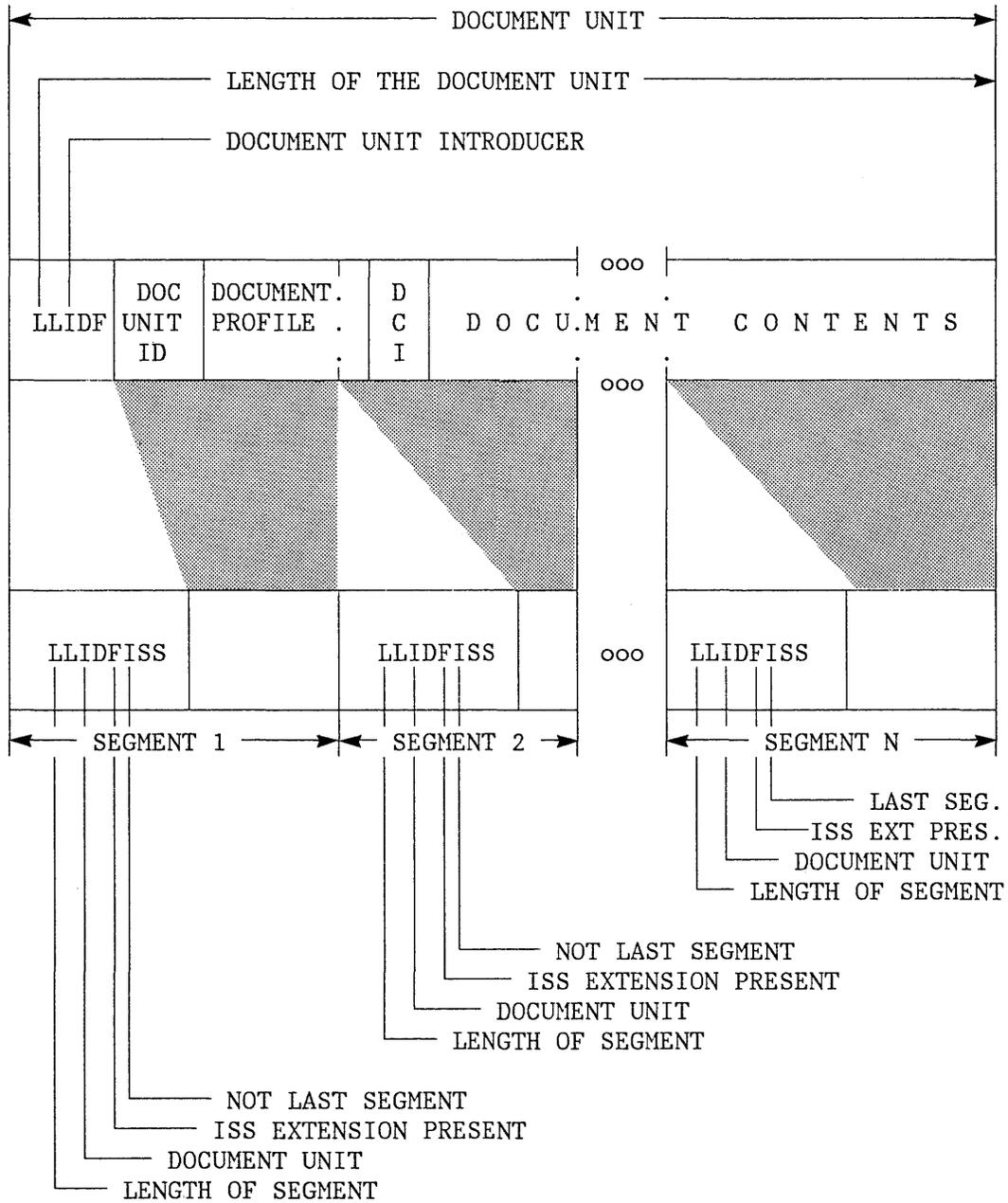


Figure 26. Document Unit Segmentation Example

The example above illustrates the mapping of an unsegmented document unit into a segmented document unit.

SUMMARY OF DIU SYNTAX RULES

The following table presents a summary of the general syntax rules for DIU data stream components:

- Row 1 specifies whether this component is a required or optional component of a DIU.
- Row 2 specifies the minimum and maximum number of times that each individual DIU data stream component can occur within a single DIU.
- Rows 3 and 4 specify the rules for the order of occurrence of the DIU data stream components both for the normal (nonexception) DIU and for exception (abnormal) situations.
- Row 5 specifies whether the DIU component can be segmented. DIU subcomponents cannot be segmented.

R O W	SYNTAX RULES	DOCUMENT INTERCHANGE UNIT (DIU) COMPONENTS				
		PREFIX	COMMAND	DATA UNIT	DOC UNIT	SUFFIX
1	OPTIONAL-	no	no	yes	yes	no
2	RANGE- MIN: MAX:	1 1	1 255	0 255	0 255	1 1
3	ORDER- NORMALLY PRECEDED BY: NORMALLY FOLLOWED BY:	nothing or suffix of last DIU command	prefix or another command another command or data unit or doc unit or type 1 suffix	last command or another data unit another data unit or doc unit or type 1 suffix	last command or last data unit or another doc unit another doc unit or type 1 suffix	last command or last data unit or last doc unit nothing or prefix of next DIU
4	EXCEPTION- PRECEDED BY: FOLLOWED BY:	nothing type 2 suffix	prefix or another command type 2 suffix	last command or another data unit type 2 suffix	last command or last data unit or another doc unit type 2 suffix	not applic. not applic.
5	CAN COMPONENT BE SEGMENTED?	no	yes	yes	yes	no

Figure 27. Summary of DIU Syntax Rules

CGCSGID ENCODINGS

The DIU consists of five logical components: a prefix, a command sequence, optional data units, optional document units, and a suffix. Within these components, DIA specifies certain operands and parameters as having character values. DIA further characterizes character-valued operands and parameters as either ID type, name type, password type, or message type.

The characters and character-to-value mappings used in the values of these operands and parameters are defined and registered in the IBM Corporate Specification C-H 3-3220-050, the Registry of Graphic Character Sets and Code Pages. That document assigns registered values to character sets and to mappings of these character sets to code pages of hexadecimal values. DIA processes use these registered identifiers to support connectivity and character information interchange.

DIA specifies that the CGCSGID (Coded Graphic Character Set Global Identifier) in effect at the beginning of each DIU component is 00337/00256 (character set 00337; code page 00256). This default, CGCSGID, can be temporarily overridden during the parsing or processing of some operands or parameters by using the LT, self-defining data variable encoding method or the MESSAGE (Format 2) operand.

Note: The shorter and more familiar acronym GCID was formerly used instead of CGCSGID.

In order to enhance connectivity and character data exchange, DIA associates each character type operand or parameter with rules that the sender of the operand or parameter must follow. However, receivers must allow for the possibility that senders have not followed the DIA rules specified for character-valued operands or parameters. Therefore, receivers must be prepared to handle character data that do not conform to the DIA rules.

Detailed information about DIA character set handling, including charts showing several character set-to-code page mappings are shown in Appendix D, "Graphic Character Sets." A character set conversion table is shown in Appendix E, "Character Set Translation."

CHAPTER 6. EXCEPTION DETECTION, CLASSIFICATION, AND REPORTING

This chapter describes the types of exception conditions detected, the error reporting structure within DIA, and the recovery responsibilities of the DIA processes.

EXCEPTION CONDITION DETECTION

The objective of Document Interchange Architecture is to provide the reliable exchange of information between DIA processes. To satisfy this objective, the DIA process sending a DIU is responsible for insuring that the DIU is both precise and correct. The DIA process receiving the DIU is responsible for reporting any exception conditions that it detects. The DIA process receiving the DIU is also responsible for recommending appropriate recovery actions, if any. The DIA process that sent the DIU containing an exception condition is responsible for carrying out the recovery action.

EXCEPTION CONDITION CLASSIFICATION

DIA exception conditions are described in terms of an ordered taxonomy, such as exception condition class, severity, condition code, exception condition object, and exception condition data.

The exception condition classes are defined in the following figure:

Exception Condition Classes	Class Definition
No-Exception	This class represents the case where no exception conditions were detected.
Session	This exception condition class is used to report violations of defined or negotiated session protocols, such as requesting an application service outside the negotiated function sets.
Syntax	This exception condition class is used to report violations of DIU syntax rules, such as omitting a required operand for a DIA command.
Sematic	This exception condition class is used to report conflicting parameters, such as specifying an incorrect password.
Process	This exception condition class is used to report exception conditions detected during function request processing, such as insufficient resources to complete a requested function.
Sender	This exception condition class is used to report a situation that prevents a DIU sender from completing transmission of the DIU, such as a permanent disk input/output error that occurs when reading a document being delivered in a DIU.

Figure 28. Exception Condition Classes

Associated with these exception condition classes is a severity indicator. The severity indicators are:

- Information - the DIU request is completed normally.
- Warning - the requested results might be incorrect.
- Severe - the request concluded with an exception.
- Catastrophic - the request was not processed.

The condition code defines the specific condition detected, for example, function not supported, unauthorized access, or data not found. A complete list of condition codes is found in the tables below.

The exception condition object defines the DIA object that is incorrect; for example, an operand value (such as password) was incorrect, or a command is not supported. A complete list of exception condition objects is in the tables later in this chapter.

The exception condition data field is used to report the object and object value that cause the exception condition. For example, the exception condition data field contains the operand introducer LLIDF and its data variable (such as the PASSWORD operand) that is incorrect.

EXCEPTION CONDITION REPORTING

These exception conditions are reported using either the generalized ACKNOWLEDGE command or, for a sender-detected error, the Suffix Type 2 Exception Condition Code field. Exception conditions reported using the ACKNOWLEDGE command are located in the EXCEPTION-CODE operand. In either case, whether contained in a Suffix Type 2 Exception Condition Code or in an EXCEPTION-CODE operand, the structured-field data-variable format is the same. The data variable format is defined in Figure 29.

When the processes detect no exception condition, the Exception Condition Class field specifies X'00' and the other operand fields need not be sent and can be ignored if received.

The Condition Code field value of X'00' indicates a user-specified condition. DIA defines the exception condition to be outside the scope of the DIU operation. Exception conditions with this code are passed to the application process for interpretation and disposition.

The EXCEPTION CODE operand and Suffix Type 2 structured-field data variable has the following format:

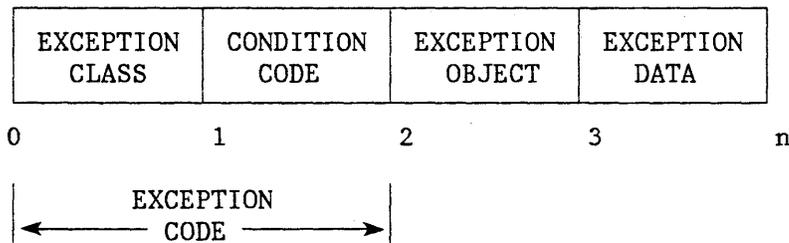


Figure 29. Exception Condition Code Format

The byte containing the class of the exception condition has the following structure. The severity of the exception condition is contained in the two high-order bits of the exception condition class and is defined following the class values.

Byte Value	Exception Class
X'00'	No exception
X'x1'	Session
X'x2'	Syntax
X'x3'	Semantic
X'x4'	Process
X'x5'	Sender
X'C6' - 'FF'	Reserved

Figure 30. Exception Condition Coding Values

The severity of the exception condition is encoded in the two high-order bits of the exception condition class byte. The definition of these bits are:

Bit Values	Condition Severity	Action Taken
B'00'	Information	Request processed to normal conclusion
B'01'	Warning	Request result may be incorrect
B'10'	Severe	Request concluded with exception
B'11'	Catastrophic	Request not processed

The condition code byte has the following definition:

Byte Value	Condition Definition	Byte Value	Condition Definition
X'00'	User-specified condition	X'0E'	Format invalid
X'01'	Function not supported	X'0F'	Length invalid
X'02'	Data not supported	X'10'	Indicator invalid
X'03'	Unauthorized access	X'11'	Range exceeded
X'04'	Resource not available	X'12'	Intervention required
X'05'	Password invalid	X'13'	Time-Out
X'06'	Execution terminated	X'14'	Cancelled
X'07'	Data not found	X'15'	Subfield length invalid
X'08'	Segmentation	X'16'	Subfield type invalid
X'09'	Data found	X'17'	Invalid parameters
X'0A'	Sequence	X'18'	Content error
X'0B'	Input/output error	X'19'	
X'0C'	ID invalid	- 'FF'	Reserved
X'0D'	Reserved		

The exception condition object byte has the following definition:

Byte Value	Object	Byte Value	Object
X'00'	Reserved	X'10'	Document content introducer
X'01'	DIU Prefix	X'11'	Document content control
X'02'	DIU ID	X'12'	Document content data
X'03'	Reserved	X'13'	DIU suffix
X'04'	Reserved	X'14'	Segment
X'05'	Reserved	X'15'	Recoverable unit
X'06'	Reserved	X'16'	Unsupported
X'07'	Command	X'17'	Unknown
X'08'	Command Operand	X'18'	User-Specified Data retired
X'09'	Operand Value	X'19'	Data Object Profile
X'0A'	Data Unit	X'1A'	Data Object Data
X'0B'	Data Unit Content	X'1B'	
X'0C'	Document Unit	X'1C'	
X'0D'	Document Unit ID	-X'FF'	Reserved
X'0E'	Document profile		
X'0F'	Document profile Parameter		

The Exception Condition Data field is a variable-length byte-string of up to 247 bytes. It contains the DIU data stream component or subcomponent that was detected as having the condition described by the exception condition code. When the entity has a DIU introducer, the exception condition data field will contain the introducer and the data bytes that are bounded by the length field of the introducer. If the exception condition data cannot be explicitly identified with a DIU introducer, then the exception condition data field will contain the byte string in which the exception condition is detected, beginning with the first byte that generated the exception condition.

RECOMMENDED RECOVERY ACTIONS

The receiver of a DIU that causes an exception condition can recommend the form of recovery action to the sender of that DIU. This recommendation is carried in the RECOVERY-ACTION operand of the ACKNOWLEDGE command. The receiver of the RECOVERY-ACTION operand is not bound to the recommended action. If the recommendation is not followed and the subsequent recovery is unacceptable to the sender, the session can be terminated. The recovery actions that can be recommended in the coding for the RECOVERY-ACTION operand and their encoding are as follows:

Byte Value and Meaning	Recommended Recovery Action
X'00' None	Recovery action is to be determined by the sender of the offending DIU.
X'01' Resend	The sender of the offending DIU should send that DIU again immediately after receiving the ACKNOWLEDGE command containing this value.
X'02' Skip and resend	The sender should send the offending DIU again after sending all other DIUs scheduled for reply to the current request.
X'03' Postpone	The offending DIU should not be sent again until the sender of the exception condition code requests the DIU on this or on a subsequent DIA session.
X'04' Cancel	The offending DIU should not be sent again.
X'05' Terminate the exchange	Stop sending the offending DIU and terminate the command that requested it. Only send the DIU or any other scheduled DIUs after a subsequent command is issued to request them.

If the RECOVERY-ACTION operand does not appear in an ACKNOWLEDGE command that has a non-zero exception condition class, the recovery action default is X'00', none. Recovery action values are ignored on a normal ACKNOWLEDGE command.

CHAPTER 7. FUNCTION SETS AND COMMANDS: INTRODUCTION

This chapter:

- Introduces the function set as describing the capabilities of an office system
- Describes the negotiation for a function set that occurs during a DIA session
- Describes the presentation in the next five chapters of the definition of each function set
- Describes the presentation in the next five chapters of DIA commands.

FUNCTION SETS

Because office systems vary in their capabilities, DIA commands are grouped into function sets that identify the scope of work for a DIA session. These function sets have been defined so that each set contains all the commands required for a well-defined, usable, and complete set of functions for a given category of services.

Function Set Negotiation

DIA processes establish a logical connection, called a DIA session, through which they exchange information. The DIA session is negotiated when the two DIA processes exchange SIGN-ON commands and agree on the scope of work that is to be performed. This agreement is necessary because not all products that implement DIA support the same range of functions. DIA defines a wide range of office system functions; most office systems require only a subset of these functions for their operation.

During the negotiation, the DIA processes determine which role each will play. The process that is the command requester is identified as process B, and the process that is the command server is identified as process A. When DIA processes act simultaneously as a requester and a server of a DIA function, the DIA process must assume the role of both processes A and B.

Function Set Definition

Figure 31 through Figure 43 define how the DIA commands are grouped to form the various function sets. The figures list each command in the function set and identify the valid command class and the request/reply protocol for each command. The request/reply protocol is represented by send or receive in the columns for process A and process B. Send indicates that the process is the command requester, and receive indicates that the process is the command server. Support of any function set requires that a DIA process assuming the role of either process A or process B must recognize and process all the commands designated as receive.

COMMAND DESCRIPTION

Chapters 8 through 12 contain the DIA command descriptions for DIA Session Services, File Transfer Services, Document Distribution Services, Document Library Services, and Application Services.

Each description of a command begins with the name of the command and a list of the command operands. Operands that are optional are enclosed in brackets. Required operands are shown without brackets. Operands cannot be repeated unless the command descriptions explicitly states that they can be.

The function of each command is explained, followed by a description of each operand. The detailed definitions of each operand are contained in Appendix A. The structured-field identifier and format fields (IDF) for each command are defined in Appendix B.

The description of each command contains the request/reply protocol used between the requester and server. Scenarios for normal DIU processing and for DIUs containing exception conditions are also shown.

The description of each command concludes with a list of exception conditions that are specific to that command. The general exception conditions that are common to all DIA commands are described in Appendix F.

CHAPTER 8. FUNCTION SETS AND COMMANDS: DIA SESSION SERVICES

DIA Session Services commands establish, maintain, and terminate DIA sessions between DIA processes. The following commands are summarized here:

- The SIGN-ON command is used to negotiate a DIA session between two DIA processes, and to determine the functions to be used during that session. It also enables the DIA processes to validate each other's authority to exchange information.
- The SIGN-OFF command terminates a DIA session.
- The ACKNOWLEDGE command is a general reply that notifies a DIA process that 1) a requested DIU command completed normally or 2) completed with an exception condition.
- The SET-CONTROL-VALUE command provides the ability for one DIA process to establish, change, or delete the value associated with a control variable (such as a password) defined at the receiving DIA process.

Function set 10 contains the Session Services commands to create, change or delete DIA control variables.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
SET-CONTROL-VALUE	SRR	receive	send
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive
SIGN-OFF	NRR	send/rec	send/rec

Figure 31. Function Set 10

SIGN-ON

<u>Command</u>	<u>Operands</u>
SIGN-ON	FUNCTION-SET [,COUNT] [,SIGN-ON-ID] [,SIGN-ON-PASSWORD] [,CHARGE-CODE] [,DOCUMENT-TYPE] [,GRAPHIC-CHARACTER-SET-ID] [,CORRELATION]

One DIA process uses the SIGN-ON command to request a DIA session with another DIA process.

The SIGN-ON request proposes the set of functions to be used during the DIA session. The proposed function sets represent either the full receiving capability of the requester or the specific function sets that the requester wants to use during the session.

The DIA process that receives the SIGN-ON request examines the proposed FUNCTION-SET options and does one of the following:

1. Agrees to a DIA session to support the requested function sets by returning a SIGN-ON (NRR) reply whose FUNCTION-SET operand value includes the same FUNCTION-SET-IDs as were requested, but with the complementary process roles specified.
2. Agrees to a DIA session to support a subset of the requested function sets by returning a SIGN-ON (NRR) reply whose FUNCTION-SET operand value includes a subset of the same FUNCTION-SET-IDs requested, but with the complementary process roles specified.
3. The receiver of a negotiated SIGN-ON NRR reply can reject the DIA session negotiation by returning a negative ACKNOWLEDGE command with the appropriate exception codes.

The receipt of a SIGN-ON request within an active DIA session from the same requester is treated as an implicit request to SIGN-OFF and begin a new DIA session with the specified parameters. For example, a DIA process can effect a mid-session change of function set roles or session options.

Operand Descriptions

FUNCTION-SET

The FUNCTION-SET (Format 1) operand identifies the specific set of proposed DIA functions to be used during the requested DIA session and identifies the role that the sender of this operand wishes to assume.

COUNT

The COUNT (Format 1) operand is optional. It specifies a numeric value from 1 to 255 that indicates the maximum number of commands the sender of the SIGN-ON command can receive in a DIU command sequence. The default value is 1.

SIGN-ON-ID

The SIGN-ON-ID operand, if present, identifies the DIA end user to participate in a DIA session; either SIGN-ON-ID operand (Format 1 or Format 42) can be used. The SIGN-ON-ID operand value identifies the end user and is used for implementing the authorization to services or access to data. If the command does not contain the operand, the products can default the value to a predefined 1- to 8-byte character name, such as the LU name or the default SIGN-ON-ID operand that identifies one communications session and that serves the same function as the SIGN-ON-ID operand. If the receiver cannot find a default value and identification is required, then a SOURCE-ADDRESS operand (which serves the same function as the SIGN-ON-ID address) must be specified on every command request that is sent during the session.

SIGN-ON-PASSWORD

The SIGN-ON-PASSWORD (Format 1) operand, if present, specifies a 1- to 8-character access-authorization key associated with the DIA end user that wishes to participate in a DIA session.

The SIGN-ON-PASSWORD operand is required for products that implement the procedure that provides password-protected authorization to obtain access. When the SIGN-ON-PASSWORD operand is used, the SIGN-ON-ID (Format 1) operand must also be specified.

CHARGE-CODE

The CHARGE-CODE (Format 1) operand, if present, contains a character string that identifies the accounting information used to accrue any charges the user incurs during the requested DIA session.

The CHARGE-CODE operand is required when the specific product that implements DIA provides support for user accounting.

DOCUMENT-TYPE

The DOCUMENT-TYPE (Format 1) operand, if present, identifies the specific document types that can be received during the DIA session.

The DOCUMENT-TYPE operand value is a vector. It consists of one or more 2-byte document type identifiers, each of which specifies a document type identifier value for a document type that can be received. If the operand is omitted, documents of any type can be delivered to the requester during the DIA session. The document type values specified by this operand are applicable only to the requester's receiving capabilities, not to the requester's sending capability.

When the DOCUMENT-TYPE operand is present, the document type for each document to be delivered is checked against the specified values. If the document type matches the specified value, then the document is delivered. If there is no match, but the DIA session partner is capable of transforming the document to a specified type, the document is transformed and delivered. If there is no match and document type transformation is not possible or available, the document is not sent. Documents queued for delivery to the signed-on recipient are filtered in this session; they are held in the distribution queue until some action is taken to have them delivered or cancelled.

DIA neither requires nor specifies any mandatory document type conversions that office system nodes must support. When conversions are supported however, they must be able to ensure that there is no loss of information or meaning.

This operand does not apply to the delivery of document types X'0008' and X'000A'. See Appendix C for a description of the document types.

GRAPHIC-CHARACTER-SET-ID

The GRAPHIC-CHARACTER-SET-ID (Format 1) operand, if present, identifies the specific character sets and code pages that can be used for data received during the DIA session. If the SIGN-ON command does not contain the GCID operand, validation of CGCSGID usage in text data is not required for the DIA session. Any document type that is included in the DOCUMENT-TYPE operand on the SIGN-ON command is delivered.

The use of this operand is restricted to SIGN-ON commands that are sent from a requesting (process B) node to a serving (process A) node. The values specified by this operand are applicable to the receiving capabilities of the node and not its sending capability. This operand does not apply to the delivery of document types X'0008' and X'000A'.

CORRELATION

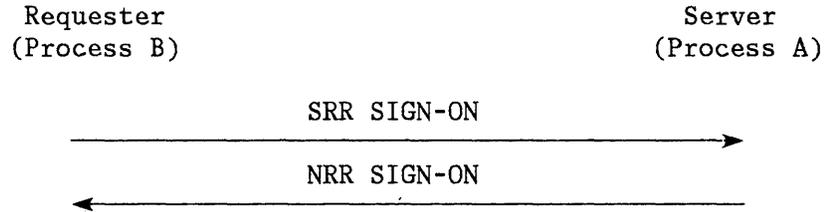
The CORRELATION (Format 1) operand is required on the SIGN-ON (NRR) reply to correlate the reply with the SIGN-ON request. Specifically, the CORRELATION (Format 1) operand identifies the SIGN-ON (SRR) request and indicates that no further replies will be sent.

Request/Reply Protocol

The following scenarios illustrate the possible replies to the SIGN-ON command:

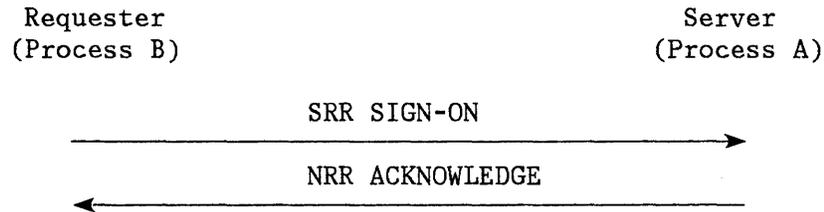
- Scenario 1 - Normal Condition

The following is a normal sign-on scenario.



- Scenario 2 - Exception Condition.

Exception conditions detected during the SIGN-ON command processing will be replied to with an ACKNOWLEDGE command that contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

See Appendix F, "DIU General Exception Conditions" for descriptions of the common general exception conditions for all DIA commands.

Additional exception conditions specific to the SIGN-ON command are:

- The SIGN-ON command ID is invalid.

Exception = Catastrophic, Session, ID-Invalid, Command
Exception Code = X'C10C07'
Exception data = LLIDF of the SIGN-ON command.
- The FUNCTION-SET operand is not specified in the SIGN-ON command.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception data = LLIDF of the FUNCTION-SET operand.
- The SIGN-ON-ID operand length is invalid.

Exception = Catastrophic, Syntax, Length-Invalid, Operand-Value
Exception Code = X'C20F09'
Exception data = LLIDF and data of the SIGN-ON-ID operand.
- The SIGN-ON-PASSWORD operand length is invalid.

Exception = Catastrophic, Syntax, Length-Invalid, Operand-Value
Exception Code = X'C20F09'
Exception data = LLIDF and data of the SIGN-ON-PASSWORD operand.
- The CHARGE-CODE operand length is invalid.

Exception = Catastrophic, Syntax, Length-Invalid, Operand-Value
Exception Code = X'C20F09'
Exception data = LLIDF and data of the CHARGE-CODE operand.
- The COUNT operand length is invalid.

Exception = Catastrophic, Syntax, Length-Invalid, Operand-Value
Exception Code = X'C20F09'
Exception data = LLIDF and data of the COUNT operand.
- The COUNT operand is out-of-range.

Exception = Catastrophic, Syntax, Range-Exceeded, Operand-Value
Exception Code = X'C21109'
Exception data = LLIDF and data of the COUNT operand.

- The CORRELATION operand Reply-Indicator value is not valid.
 Exception = Catastrophic, Syntax, Data-Not-Supported, Operand-Value
 Exception Code = X'C20209'
 Exception data = LLIDF and data of the CORRELATION operand.
- The proposed FUNCTION SET operand options are not supported.
 Exception = Catastrophic, Semantic, Function-Not-Supported, Operand-Value
 Exception Code = X'C30109'
 Exception data = LLIDF and data of the FUNCTION SET operand.
- The SIGN-ON FUNCTION SET operand options are not accepted.
 Exception = Catastrophic, Semantic, Cancelled, Command
 Exception Code = X'C31407'
 Exception data = LLIDF and data of the FUNCTION SET operand.
- Incompatible function sets or roles have been negotiated.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 Exception Code = X'C30209'
 Exception data = LLIDF and data of the FUNCTION SET operand.
- The SIGN-ON-ID operand verification failed.
 Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
 Exception Code = X'C30309'
 Exception data = LLIDF and data of the SIGN-ON-ID operand.
- The SIGN-ON-ID operand is required, but it is missing.
 Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
 Exception Code = X'C30708'
 Exception data = LLIDF of the SIGN-ON-ID operand.
- The SIGN-ON-PASSWORD operand is invalid.
 Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
 Exception Code = X'C30509'
 Exception data = LLIDF and data of the SIGN-ON-PASSWORD operand.
- The SIGN-ON-PASSWORD operand is required, but it is missing.
 Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
 Exception Code = X'C30708'
 Exception data = LLIDF of the SIGN-ON-PASSWORD operand.

- The SIGN-ON CHARGE-CODE operand is invalid.
 Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
 Exception Code = X'C30309'
 Exception data = LLIDF and data of the CHARGE-CODE operand.
- The SIGN-ON CHARGE-CODE operand is required, but it is missing.
 Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
 Exception Code = X'C30708'
 Exception data = LLIDF of the CHARGE-CODE operand.
- The GCID operand was specified with incompatible process roles.
 Exception = Warning, Semantic, Function-Not-Supported, Command-Operand
 Exception Code = X'430108'
 Exception data = LLIDF and data of the GCID operand
- The DOCUMENT TYPE operand was specified with incompatible process role(s).
 Exception = Warning, Semantic, Function-Not-Supported, Command-Operand
 Exception Code = X'430108'
 Exception data = LLIDF and data of the DOCUMENT TYPE operand.
- The user is already signed-on to another DIA session.
 Exception = Catastrophic, Process, Unauthorized-Access, Command
 Exception Code = X'C40307'
 Exception data = LLIDF and data of the SIGN-ON-ID operand.

Support Considerations

The following general support considerations apply when sending a SIGN-ON command:

- A SIGN-ON command request or reply, when sent, must be the only command in the DIU command sequence.
- A SIGN-ON SRR command received from the same signed-on user within an active DIA session is an implicit request to terminate the current session and an explicit request to begin a new session with the specified parameters. The DIA process is reinitialized, and the new sign-on exchange proceeds in the normal manner.

A second SIGN-ON request for the same SIGN-ON-ID operand can occur as the result of an undetected communications system outage or as the result of the remote DIA session partner wishing to obtain a mid-session change of DIA session parameters.

The following support considerations apply to the return of operands on replying SIGN-ON NRR commands:

- FUNCTION SET

The following rules apply to the value of the FUNCTION-SET operand that is returned in the reply:

1. The function set identifiers returned on the replying SIGN-ON command must be the same or a subset of the function set identifiers proposed in the SIGN-ON request command.
2. Complementary roles must be returned for each function set selected by the sign-on receiver for use within this DIA session.
3. It is invalid for the sign-on receiver to select terminal-to-terminal function sets (function sets 6 & 7) for use with any other function set, except function set 10 (DIA Session Services), within the same DIA session. If a choice of processing roles is presented in the initial request to sign-on, however, the receiver of the request can choose a compatible subset of the function sets it desires to support for the DIA session being established.

- SIGN-ON-ID

The return of the SIGN-ON-ID operand in replying SIGN-ON commands is optional. If the operand is omitted, then the rules for handling SIGN-ON-ID defaults apply. See the operand description for the semantics of how to handle SIGN-ON-ID operand defaults.

- SIGN-ON-PASSWORD

The return of the SIGN-ON-PASSWORD operand on replying SIGN-ON commands is optional. If the process receiving the SIGN-ON reply requires the SIGN-ON-PASSWORD operand to authorize access and the operand is not present, then the receiving process can elect not to enter into a DIA session. If the process receiving the SIGN-ON reply does not support password protection and the operand is present, then the password can be ignored. The support of password protection is a product decision.

- CHARGE-CODE

The return of the CHARGE-CODE operand on SIGN-ON NRR replying commands is optional.

If the process receiving the SIGN-ON reply requires the CHARGE-CODE operand for user accounting and the operand is not present, then the receiving process can elect not to enter into a DIA session. If the process receiving the SIGN-ON reply does not support user accounting and the CHARGE-CODE operand is returned, then the operand can be ignored. The support of user accounting procedures is a product decision.

- COUNT

The COUNT operand should be returned in replying SIGN-ON commands when the replying DIA process can receive more than one command in the DIU command sequence. The value of this operand specifies the maximum number of commands that the process can receive in any given DIU instance.

The process that requests the SIGN-ON command must validate the COUNT operand value if it proposes to send DIUs that contain multiple commands. If the requester cannot comply with the receive constraints of the replying process, then the reply must be rejected with a negative ACKNOWLEDGE command. The negative ACKNOWLEDGE command terminates the request for a DIA session.

- GCID and DOCUMENT TYPE

These operands apply only to the requester (process B) role. If they appear in connection with a server (process A) role, these operands are ignored.

SIGN-OFF

<u>Command</u>	<u>Operands</u>
SIGN-OFF	-none-

The SIGN-OFF command, whenever sent or received, terminates a DIA session.

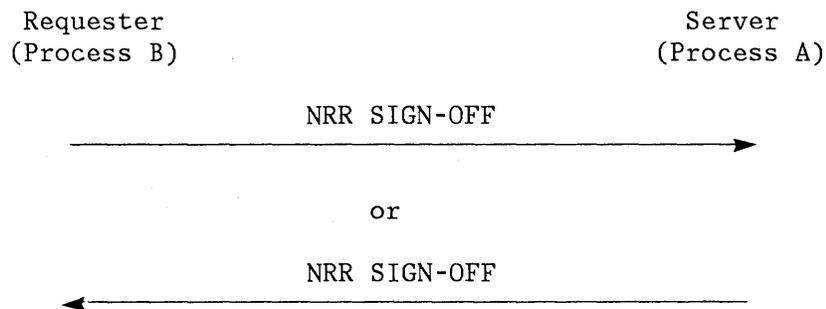
Operand Descriptions

This command contains no operands.

Request/Reply Protocol

Scenario 1 - Normal Condition.

The SIGN-OFF command is an NRR command. A reply is not expected. This command can be sent by either partner of the DIA session. A SIGN-ON command is the only valid command that can follow a SIGN-OFF command.



Exception Conditions

See Appendix F for descriptions of the common general exception conditions for all DIA commands. There are no specific exception conditions unique to the SIGN-OFF command.

ACKNOWLEDGE

<u>Command</u>	<u>Operands</u>
ACKNOWLEDGE	CORRELATION, EXCEPTION-CODE [,REPLY-DATA] [,RECOVERY-ACTION]

The ACKNOWLEDGE command is a replying command that notifies the requester whether a previously requested DIA command has been successfully or unsuccessfully completed.

The ACKNOWLEDGE command reports exception conditions detected during the processing of any DIA command. In addition to reporting exception conditions, the sender of the ACKNOWLEDGE command can also include a recommended recovery action. The receiver of the ACKNOWLEDGE command can use the recommended recovery action to recover from the specific condition. The sender of the function request command is responsible for recovery when exception conditions are reported. The function requester can use, but is not required to perform, the recommended recovery action returned on the ACKNOWLEDGE command.

The ACKNOWLEDGE command is also a reply to indicate that the correlated function request was completed successfully. In this case, the capability exists to pass back to the function requester limited reply information (for example, the REPLY-DATA operand returning the Library-Assigned Document Name (LADN) to a FILE command). When the ACKNOWLEDGE command replies to indicate successful completion of a correlated request, the semantic definition of the ACKNOWLEDGE command is defined by the request to which it is replying.

Operand Descriptions

CORRELATION

The CORRELATION (Format 1) operand is used to correlate a replying ACKNOWLEDGE command to a previously sent request. Specifically, the CORRELATION (Format 1) operand identifies the request to which the ACKNOWLEDGE command is replying and indicates that no further replies will be sent to the request.

EXCEPTION-CODE

The EXCEPTION CODE (Format 1) operand is used to specify the successful or unsuccessful completion of the request. Successful completions are indicated by specifying a X'00' in the Exception-Class field of the EXCEPTION-CODE operand. Unsuccessful request completions are indicated by specifying a non-zero Exception-Class field of the EXCEPTION-CODE operand. The types of errors that can be reported include session errors, syntax errors, semantic errors, process errors, and sender errors. The EXCEPTION-CODE operand data variable is defined in Chapter 6, "Exception Detection, Classification, and Reporting."

The EXCEPTION-CODE operand can be repeated on the ACKNOWLEDGE command to report multiple exception conditions within a request. When multiple exception conditions are reported, the exception condition with the highest severity appears as the first EXCEPTION-CODE operand specified. No ordering of the exception conditions is required after the first occurrence of the EXCEPTION-CODE operand.

REPLY-DATA

The REPLY-DATA (Format 1) operand, if present, is a command specific operand that returns data to the requester of a command. For example, the REPLY-DATA operand is used in ACKNOWLEDGE commands that are replying to a REQUEST-DISTRIBUTION command. In this case, the REPLY-DATA operand contains the unique distribution document name assigned to the document to be distributed.

RECOVERY-ACTION

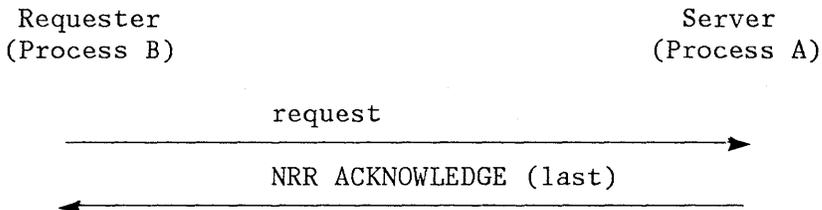
The RECOVERY-ACTION (Format 1) operand, if present, contains the recommended recovery action for the exception condition detected by the sender of the ACKNOWLEDGE command. If the operand is omitted, the receiver of the ACKNOWLEDGE command determines the recovery action. The RECOVERY-ACTION operand data variable is defined in Chapter 6, "Exception Detection, Classification, and Reporting."

Request/Reply Protocol

The following scenarios illustrate possible replies to the ACKNOWLEDGE command:

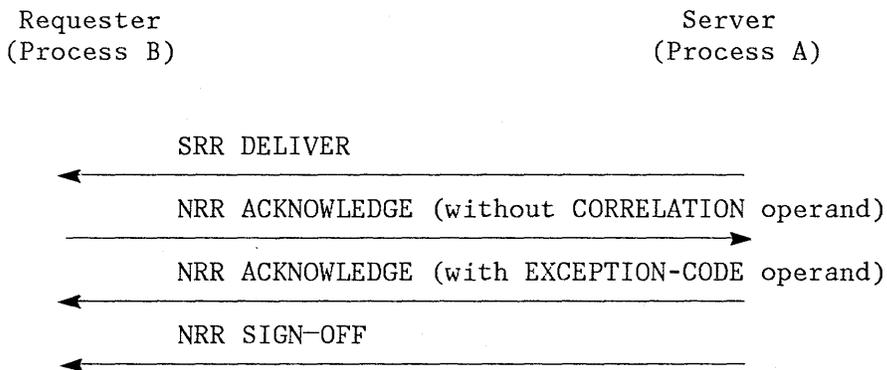
- Scenario 1 - Normal Conditions

The ACKNOWLEDGE command is a reply to a function request. The ACKNOWLEDGE command can indicate in the EXCEPTION-CODE operand whether the request was successful or unsuccessful. The ACKNOWLEDGE command always indicates that it is the last reply.



- Scenario 2 - ACKNOWLEDGE commands replying to ACKNOWLEDGE commands.

An exception condition detected in an ACKNOWLEDGE command causes the command to be rejected and the processing to be concluded. The receiver of an ACKNOWLEDGE command that is in error replies with an NRR ACKNOWLEDGE command containing the appropriate exception condition code (refer to Chapter 6, "Exception Detection, Classification, and Reporting"). This ACKNOWLEDGE reply is followed by a SIGN-OFF command to terminate the DIA session.



An ACKNOWLEDGE command without a CORRELATION operand is a syntax error. This exception condition is reported in the subsequent NRR ACKNOWLEDGE reply.

Exception Conditions

See Appendix F, "DIU General Exception Conditions" for descriptions of the common general exception conditions for all DIA commands.

The following exception conditions apply to the ACKNOWLEDGE command itself and not to the exceptions reported by the command.

- The ACKNOWLEDGE replying command does not correlate to an outstanding request.

Exception = Catastrophic, Session, Sequence, Command
Exception Code = X'C10A07'
Exception data = LLIDF and data of the CORRELATION operand
- The CORRELATION operand reply-indicator value is not valid.

Exception = Catastrophic, Syntax, Data-Not-Supported, Operand-Value
Exception Code = X'C20209'
Exception data = LLIDF and data of the CORRELATION operand
- The EXCEPTION-CODE operands are out of sequence.

Exception = Catastrophic, Syntax, Sequence, Command-Operand
Exception Code = X'C20A08'
Exception data = LLIDF and data of the Exception Code operands
- The REPLY-DATA operand contains invalid data.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF and data of the REPLY-DATA operand
- The REPLY-DATA operand is present on a reply to a command that does not require reply data.

Exception = Warning, Process, Data-Not-Supported, Command-Operand
Exception Code = X'440208'
Exception data = LLIDF and data of the REPLY-DATA operand

SET-CONTROL-VALUE

<u>Command</u>	<u>Operands</u>
SET-CONTROL-VALUE	CONTROL-VALUE

The SET-CONTROL-VALUE command is used to maintain SIGN-ON-PASSWORD values and DOCUMENT-PASSWORD values. This command can be used to establish a new password, to delete a password and its value, or to change the value of an existing password.

Operand Descriptions

CONTROL-VALUE

The CONTROL-VALUE (Format 1) operand specifies the type of password, the Old-Value field, and the New-Value field to be associated with the password.

When used to establish a password, the Old-Value field of the CONTROL-VALUE operand must be null.

When used to change the value of a password, the operand identifier of both the Old-Value field and the New-Value field of the CONTROL-VALUE operand must be identical.

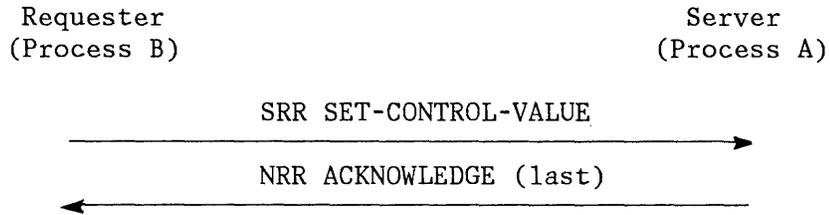
When used to delete a password, the New-Value field of the CONTROL-VALUE operand must be null.

Request/Reply Protocol

The following scenarios illustrate possible replies to the SET-CONTROL-VALUE command:

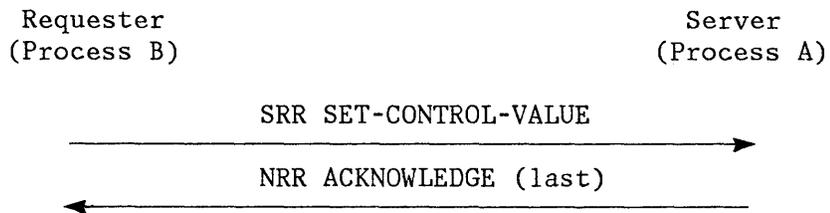
- Scenario 1 - Normal Conditions

The normal reply to a SET-CONTROL-VALUE command is an ACKNOWLEDGE command with a value of 0 in the EXCEPTION-CODE operand.



- Scenario 2 - Exception Conditions.

Exception conditions detected while the SET-CONTROL-VALUE command is being processed receive a reply consisting of an ACKNOWLEDGE command that contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

See Appendix F, "DIU General Exception Conditions" for descriptions of the common general exception conditions for all DIA commands.

Additional exception conditions specific to the SET-CONTROL-VALUE command are:

- The authorization value is required, but not specified in the CONTROL-VALUE operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Operand-Value
Exception Code = X'C20709'
Exception data = LLIDF and data of the CONTROL-VALUE operand

- An invalid authorization value identifier is specified in the CONTROL-VALUE operand.

Exception = Catastrophic, Syntax, Data-Not-Supported, Command-Operand
Exception Code = X'C20208'
Exception data = LLIDF and data of the CONTROL-VALUE operand

- An invalid SIGN-ON-ID value is specified in the CONTROL-VALUE operand.

Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
Exception Code = X'C30309'
Exception data = LLIDF and data of the CONTROL-VALUE operand

- An invalid password value is specified in the CONTROL-VALUE operand.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception data = LLIDF and data of the CONTROL-VALUE operand

- The Old Value field or the New Value field of the CONTROL-VALUE operand specifies an operand identifier that is not valid for this command.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF and data of the CONTROL-VALUE operand

- The operand identifier for the New-Value field does not equal the operand identifier for the Old-Value field.

Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception data = LLIDF and data of the CONTROL-VALUE operand

Support Considerations

No authorization password is required when changing the SIGN-ON-PASSWORD or DOCUMENT-PASSWORD operands for the signed-on user. However, if the SET-CONTROL-VALUE command is used to change either of these passwords for a user other than the signed-on user, an authorization password is required. In this case, to change another user's SIGN-ON-PASSWORD, the other user's SIGN-ON-ID operand is required for authorization. To change another user's DOCUMENT-PASSWORD operand requires the other user's SIGN-ON-ID and SIGN-ON-PASSWORD operands for authorization. For the latter, the authorization value portion of the CONTROL-VALUE operand will contain two DIA operand introducer and value combinations. These introducer and value pairs can appear in either order.

CHAPTER 9. FUNCTION SETS AND COMMANDS: DOCUMENT LIBRARY SERVICES

Document Library Services commands provide the functions for maintaining user documents in a document library. Details of these commands are described below.

- The DELETE command permanently removes access to the identified document for the requester of the DELETE command. A document that has two or more owners is removed from library storage only after all the owners request deletion.
- The DELIVER command transports a document from a server node to a requester node.
- The FILE command preserves the identified document in the library for an authorized document owner.
- The RETRIEVE command returns a library copy of the identified document to a requester authorized to retrieve the identified document.
- The SEARCH command locates the documents in the library that have characteristics that match the search criteria specified by the requester of the search. This command creates and preserves a named list of references (pointers) to the documents selected by the search. The list of references are used to retrieve the document descriptors or the document contents.

Figure 32 shows the grouping of these commands into the Document Library Services function set.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
DELETE	SRR	receive	send
FILE	SRR	receive	send
RETRIEVE	SRR	receive	send
SEARCH	SRR	receive	send
DELIVER	NRR	send	receive
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive
SIGN-OFF	NRR	send/rec	send/rec

Figure 32. Function Set 8

DELETE

<u>Command</u>	<u>Operands</u>
DELETE	IDENTIFIED-DATA [,ORIGINATING-NODE-ADDRESS] [,SOURCE-ADDRESS] [,SOURCE-PASSWORD]

The document owner uses the DELETE command to remove his ownership of the identified document. If the primary owner deletes his ownership, he also gives up all access authority. When all owners of the identified document have deleted their ownership, the Document Library Services server permanently deletes the document from the document library.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be deleted. Format 3 and Format 42 of the IDENTIFIED-DATA are valid in support of the DELETE command.

The IDENTIFIED-DATA (Format 3) operand identifies the document to be deleted by the position of the document reference in the specified Search Result List. In this version of DIA, only the IDENTIFIED-DATA (Format 3) operand value that specifies document content and IDP is valid in the DELETE command.

The IDENTIFIED-DATA (Format 42) operand identifies the document to be deleted by means of its Library-Assigned Document Name (LADN).

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand, if present, specifies the 1- to 8-character group address token of the requester that initiated this request. If this operand is not present, the group address token of the command sender is assumed.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand, if present, specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS operand (Format 1 or Format 42) is valid. If this operand is not present, the element address token of the command sender is assumed.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated this request. The SOURCE-PASSWORD (Format 1) operand is conditionally required if the SOURCE-ADDRESS (Format 1) is specified. The SOURCE-PASSWORD (Format 1) operand is required if all the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different than that of the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

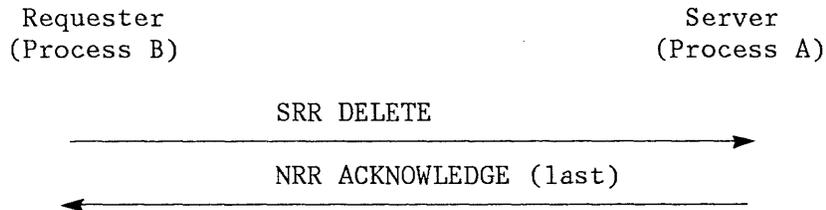
Otherwise, the SOURCE-PASSWORD should be omitted.

Request/Reply Protocol

The following scenarios illustrate possible replies to the DELETE command:

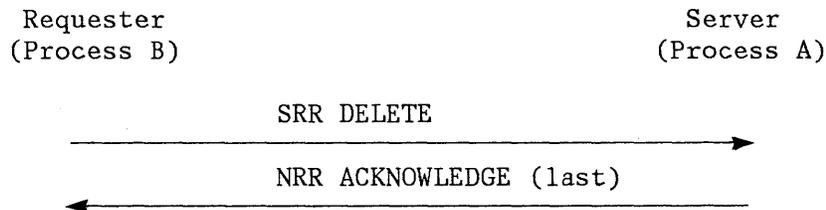
- Scenario 1 - Normal Condition

The reply to a DELETE command is an ACKNOWLEDGE command sent to the requester as a verification that the deletion is complete.



- Scenario 2 - Exception Condition.

Any exception conditions detected during the processing of the DELETE command are identified in an ACKNOWLEDGE command. This command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

See Appendix F, "DIU General Exception Conditions" for the common general exception conditions that apply to all DIA commands. The DELETE command has the following specific exception conditions in addition to the general exception conditions.

- The SOURCE-PASSWORD operand is present and the SOURCE-ADDRESS (Format 1) operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception data = LLIDF of SOURCE-ADDRESS (Format 1) operand

- The SOURCE-ADDRESS operand specifies a requester different from the command sender, the specified requester does not have affinity with the command sender, the specified requester has a password, and the SOURCE-PASSWORD operand is not specified.

Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
Exception Code = X'C30708'
Exception data = LLIDF of SOURCE-PASSWORD operand

- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

- The ORIGINATING-NODE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of ORIGINATING-NODE-ADDRESS operand and data

- The SOURCE-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception data = LLIDF of SOURCE-PASSWORD operand and data

- The document specified in the IDENTIFIED-DATA operand cannot be found.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
Exception Code = X'C40709'
Exception data = LLIDF of IDENTIFIED-DATA operand and data

- The specified requester is not an owner or owner-delegate of the document specified in the IDENTIFIED-DATA operand. The requested library process can be made only for a requester that is a designated document owner.

Exception = Catastrophic, Process, Unauthorized-Access, Operand-Value
Exception Code = X'C40309'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

DELIVER

<u>Command</u>	<u>Operands</u>
DELIVER	IDENTIFIED-DATA, CORRELATION

The DELIVER command is a replying command. It returns a document from a Document Library Service command server to the Document Library Service command requester. For example, the DELIVER command returns a document to a requester in reply to a RETRIEVE command.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be delivered. Only Format 1 of this operand is valid in support of the DELIVER command.

The IDENTIFIED-DATA (Format 1) specifies that the document to be delivered is located in the DIU document unit.

CORRELATION

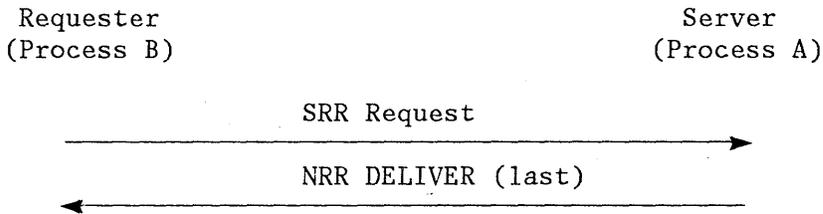
The CORRELATION (Format 1) operand correlates a replying command to a previously sent request. The CORRELATION (Format 1) operand identifies the request to which this DELIVER command is replying and indicates that no further replying commands will be sent to the request.

Request/Reply Protocol

The following scenarios illustrate possible replies to the DELIVER command:

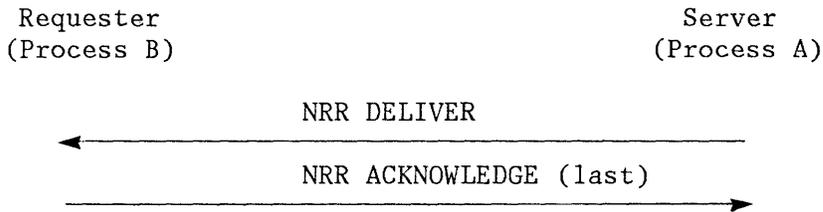
- Scenario 1 - Single DELIVER Reply

The following is an example of a request and of a single DELIVER reply.



- Scenario 2 - Exception Conditions.

Exception conditions detected during the DELIVER command processing are replied to with an ACKNOWLEDGE command that contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

See Appendix F for the common general exception conditions for the DIA commands. The receiver rejects the DELIVER command if any of the following conditions exist:

- The receiver is not ready to receive and send data.

Exception = Catastrophic, Session, Intervention-Required, Unknown
Exception Code = X'C11217'
- The IDENTIFIED-DATA operand is omitted.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
- The CORRELATION operand does not refer to a command previously sent by the receiver.

Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of CORRELATION operand
- The IDENTIFIED-DATA operand refers to nonexistent data.

Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of IDENTIFIED-DATA operand
- The document unit type is not supported.

Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit
Exception Code = X'C3020C'
Exception Data = LLIDF of document unit introducer
- The document content introducer type is not supported.

Exception = Catastrophic, Semantic, Data-Not-Supported,
Document-Content-Introducer
Exception Code = X'C30210'
Exception Data = LLIDF of document content introducer
- The document type in the document unit ID is not supported.

Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit ID
Exception Code = X'C3020D'
Exception Data = LLIDF and data of document unit ID

- The document type parameter in the base subprofile is not supported.

Exception = Catastrophic, Semantic, Data-Not-Supported,
Document-Profile-Parameter
Exception Code = X'C3020F'
Exception Data = LLIDF and data of document type parameter

- The document profile is not supported.

Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Profile
Exception Code = X'C3020E'
Exception Data = LLIDF of the document profile

- Receiving process resources are unavailable.

Exception = Catastrophic, Process, Resource-Not-Available, Document-Unit
Exception Code = X'C4040C'

- The receiving process cancels the delivery of the data.

Exception = Catastrophic, Process, Cancelled, Command
Exception Code = X'C41407'

FILE

<u>Command</u>	<u>Operands</u>
FILE	IDENTIFIED-DATA [,ORIGINATING-NODE-ADDRESS] [,SOURCE-ADDRESS] [,SOURCE-PASSWORD] [,ACCESS-CODE]

The FILE command stores information in the document library. Storable types of information include documents that have associated document profiles, and document profiles that refer to externally-stored documents. The information to be stored is submitted with the request in the DIU document unit.

A stored document must have an interchange document profile (IDP). The IDP must include parameters that specify the user-assigned document name, the document type, and the character set ID (CGCSGID) in which the profile parameters are coded.

The Document Library Services command server creates a unique name for the document by concatenating the document library node address with the date and time that the file process is completed. The Document Library Service returns this Library-Assigned Document Name (LADN) to the requester in the REPLY-DATA operand of the ACKNOWLEDGE command. The library retains the document's LADN as a reference for subsequent document processing.

The requester can specify the access characteristics of the document as private (accessible only by the owners and users with affinity to the owners), public (accessible by any user authorized to use the library), or shared (accessible by owners and members of predefined user groups). Documents can also be filed according to document classification defined when the DIA document library is created.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be filed. Only Format 1 of the IDENTIFIED-DATA operand is valid in support of the FILE command.

The IDENTIFIED-DATA (Format 1) operand specifies that the document to be filed is located in the DIU document unit.

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand is optional. It specifies the 1- to 8-character group address token of the requester that initiated this request. If this operand is not present, the group address token of the command sender is assumed.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand is optional. It specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS (Format 1 or Format 42) is valid. If this operand is not present, the element address token of the command sender is assumed.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated this request. The SOURCE-PASSWORD (Format 1) operand is conditionally required if the SOURCE-ADDRESS (Format 1) is specified. The SOURCE-PASSWORD (Format 1) operand is required if all the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different from the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

Otherwise, the SOURCE-PASSWORD can be omitted.

ACCESS-CODE

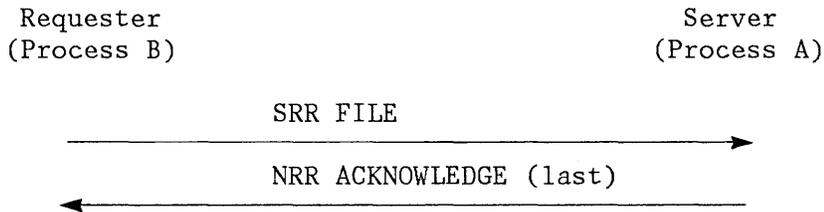
The ACCESS-CODE (Format 41) operand, if present, defines the 4-byte decimal value user-group codes that control accessibility to documents by nonowners. The user group C'0000' is defined to be public (all users can access the document). Members of a user group associated with a particular document are permitted read-only access to the document. If the operand is omitted, the document is filed as a private document that only document owners and users with affinity to the owners can access.

Request/Reply Protocol

The following scenarios illustrate the possible replies to the FILE command:

- Scenario 1 - Normal Condition

The filed document is assigned a LADN and stored as a private, public, or shared document, according to the processing options. The requester of the FILE command is defined as the primary owner of the document. The file server generates the LADN by concatenating the document library node address token (ID) and the date and time that the file process was successfully completed. The reply to a FILE command is an ACKNOWLEDGE command that is sent to the requester when the document has been filed in the document library. The replying ACKNOWLEDGE command also contains the LADN returned in the REPLY-DATA operand. No exception condition is indicated in the EXCEPTION-CODE operand.

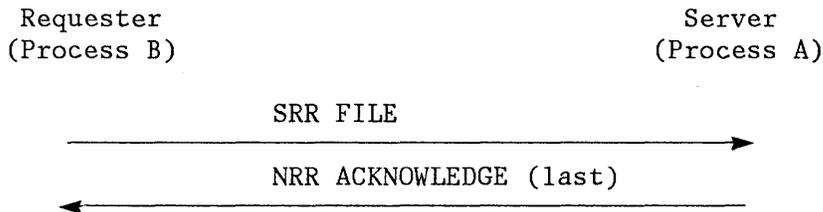


The format of the REPLY-DATA operand is defined as follows:

REPLY-DATA	Library-Assigned Document Name
INTRODUCER	Defined as IDD (Format 42)
LLIDF	LLIDF
X'nnnnC34501'	X'nnnnC32042' (IDD (Format 42) Introducer)
	LT X'0A01' Date and Time
	X'YYMDhms'hs'
	LT X'nn02' Document Library Node Address
	C'ccc . . . c' 1- to 8-byte character string.

- Scenario 2 - Exception Condition.

An ACKNOWLEDGE command that contains the exception condition in the EXCEPTION-CODE operand replies to exception conditions detected during the FILE command processing.



Exception Conditions

The general exception conditions that are common to the DIA commands are described in Appendix F. The following exception conditions are specific to the FILE command and are detected and reported in addition to the general exception conditions.

- The SOURCE-PASSWORD operand is present, and the SOURCE-ADDRESS (Format 1) operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
 Exception Code = X'C20708'
 Exception data = LLIDF of SOURCE-ADDRESS (FORMAT 1) operand

- The SOURCE-ADDRESS operand specifies a requester different from the command sender, the specified requester does not have affinity with the command sender, the specified requester has a password, and the SOURCE-PASSWORD operand is not specified.

Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
 Exception Code = X'C30708'
 Exception data = LLIDF of SOURCE-PASSWORD operand

- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 Exception Code = X'C30209'
 Exception data = LLIDF of SOURCE-ADDRESS operand and data

- The ORIGINATING-NODE-ADDRESS operand contains an invalid address.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 Exception Code = X'C30209'
 Exception data = LLIDF of ORIGINATING-NODE-ADDRESS operand and data
- The SOURCE-PASSWORD operand contains an invalid authorization key.
 Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
 Exception Code = X'C30509'
 Exception data = LLIDF of SOURCE-PASSWORD operand and data
- An invalid document classification is specified.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Document Profile
 Parameter
 Exception Code = X'C3020F'
 Exception data = LLIDF of DOCUMENT-CLASS parameter and data
- The DOCUMENT-CLASS profile parameter is required but is missing.
 Exception = Catastrophic, Semantic, Data-Not-Found, Document Profile
 Parameter
 Exception Code = X'C3070E'
 Exception data = none
- An invalid access code is specified.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 Exception Code = X'C30209'
 Exception data = LLIDF of ACCESS-CODE operand and data
- An invalid access code range is specified.
 Exception = Catastrophic, Semantic, Range-Exceeded, Operand-Value
 Exception Code = X'C31109'
 Exception data = LLIDF of ACCESS-CODE operand and data
- The document specified in the IDENTIFIED-DATA operand cannot be found.
 Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
 Exception Code = X'C40709'
 Exception data = LLIDF of IDENTIFIED-DATA operand and data

RETRIEVE

<u>Command</u>	<u>Operands</u>
RETRIEVE	IDENTIFIED-DATA [,ORIGINATING-NODE-ADDRESS] [,SOURCE-ADDRESS] [,SOURCE-PASSWORD] [,RETRIEVE-COUNT] [,DESCRIPTOR-CONTENT-DEFINITION]

The RETRIEVE command obtains a copy of the identified information from the server's document library. The types of information that can be retrieved include a document (profile and content), a document profile only, a document content only, a set of search-selected document descriptors, and a set of search-selected document descriptors with their associated document content.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be retrieved. Format 3 and Format 42 of the IDENTIFIED-DATA operand are valid in support of the RETRIEVE command.

The IDENTIFIED-DATA (Format 3) operand identifies the document to be retrieved by the position of the document reference in the specified Search Result List.

In this version of DIA, only the IDENTIFIED-DATA (Format 3) operand value that specifies one of the following is valid in support of the RETRIEVE command.

- The document content and associated profile
- The document content only
- The document profile only
- The specified document descriptors
- The specified document descriptors with the associated document content.

The IDENTIFIED-DATA (Format 42) operand identifies the document to be retrieved by means of its Library-Assigned Document Name (LADN).

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand, if present, specifies the 1- to 8-character group address token of the requester that initiated this request. If this operand is not present, the group address token of the command sender is assumed.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand, if present, specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS operand (Format 1 or Format 42) is valid. If this operand is not present, the element address token of the command sender is assumed.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated this request. The SOURCE-PASSWORD (Format 1) operand is conditionally required if the SOURCE-ADDRESS (Format 1) is specified. The SOURCE-PASSWORD (Format 1) operand is required if all of the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different from the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

Otherwise, the SOURCE-PASSWORD should be omitted.

RETRIEVE-COUNT

The RETRIEVE-COUNT (Format 1) operand is applicable only when the IDENTIFIED-DATA (Format 3) operand is present and specifies the maximum number of document descriptors to be returned in response to a RETRIEVE command requesting document descriptors. When this operand is omitted and the request (IDENTIFIED-DATA (Format 3)) is for document descriptors, a default value of 1 is assumed. If the IDENTIFIED-DATA (Format 42) is specified or the IDENTIFIED-DATA (Format 3) operand does not request that document descriptors be retrieved, the RETRIEVE-COUNT (Format 1) operand, if present, is ignored.

DESCRIPTOR-CONTENT-DEFINITION

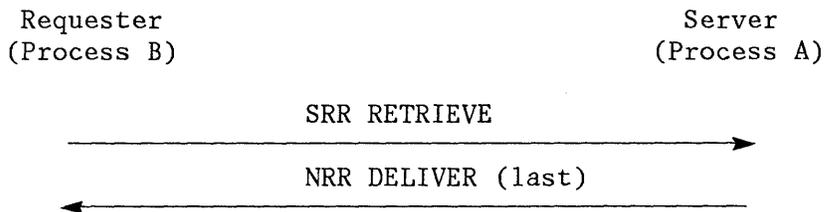
The DESCRIPTOR-CONTENT-DEFINITION (Format 41) operand is applicable only when the RETRIEVE request is for document descriptors (IDENTIFIED-DATA (Format 3) operand is specified) and specifies the introducer IDF values of the document profile parameters to be returned as document descriptors in a reply to the RETRIEVE command. The descriptors are returned in a document-descriptor document. If this operand is omitted and the RETRIEVE request is for document descriptors, only the document name is returned in the document descriptors document. If the IDENTIFIED-DATA (Format 42) is specified or the IDENTIFIED-DATA (Format 3) operand does not request document descriptors to be retrieved, the DESCRIPTOR-CONTENT-DEFINITION (Format 41) operand, if present, is ignored.

Request/Reply Protocol

The following scenarios illustrate the possible replies to the RETRIEVE command:

- Scenario 1 - Normal Condition

The normal reply to a RETRIEVE command is a DELIVER command sent at the conclusion of processing.



The DELIVER DIU document unit contains the requested information. The following examples define the contents of the DELIVER DIU document unit in reply to the RETRIEVE request. Final-Form-Text Document Content Architecture (document type X'0002') is used as the type of document referred to. The examples assume that there is no need for segmentation of the document unit.

IDD (Format 3 or 42) - Document and Document Profile

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>	<u>NAME</u>
LLIDF	5	X'nnnnC90301'	Document Unit Introducer
parameter	2	X'0002'	Document Type Identifier (Final-Form-Text Document Content Architecture)
	13	X'n...n'	System Code
LLIDF	5	X'nnnnCA0301'	Interchange Document Profile Introducer
LLIDF	5	X'nnnnCA0401'	Base Subprofile Introducer
LLIDF	5	X'nnnnC70001'	User Assigned Document Name (UADN) Introducer
parameter	1-44	X'n...n'	User Assigned Document Name
LLIDF	5	X'0007C70601'	Document Type Introducer
parameter	2	X'0002'	Final Form Text Identifier
LLIDF	5	X'0009C70101'	Profile CGCSGID Introducer
parameter	4	X'csidcpid'	Character Set and Code Page

The above base subprofile contains the DIA-required parameters. Other parameters can be present and are entered as specified by the IDP architecture. Refer to IDP.

Additional subprofiles can be specified at this point in the document unit. These can include product private subprofiles and DIA application subprofiles.

LLIDF 5 X'0005CB0101' Document Content Introducer

The final-form-text document content begins here and occupies the space remaining as specified by the LL (length) bytes of the document unit introducer.

IDD (Format 3) - Document Content Only

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>	<u>NAME</u>
LLIDF	5	X'nnnnC90301'	Document Unit Introducer
parameter	2	X'0002'	Document Type Identifier (Final Form Text Document Content Architecture)
	13	X'n...n'	System Code
LLIDF	5	X'0005CB0101'	Document Content Introducer

The final-form-text document content begins here and occupies the space remaining as specified by the LL (length) bytes of the document unit introducer.

IDD (Format 3) - Document Profile Only

Since the interchange document profile can consist of one or more subprofiles, all of these subprofiles are returned in the DIU for the DELIVER command.

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>	<u>NAME</u>
LLIDF	5	X'nnnnC90301'	Document Unit Introducer
parameter	2	X'0002'	Document Type Identifier
	13	X'n...n'	System Code
LLIDF	5	X'nnnnCA0301'	Interchange Document Profile Introducer
LLIDF	5	X'nnnnCA0401'	Base Subprofile Introducer
LLIDF	5	X'nnnnC70001'	User-Assigned Document Name (UADN) Introducer
parameter	1-44	X'n...n'	User-Assigned Document Name
LLIDF	5	X'0007C70601'	Document Type Introducer
parameter	2	X'0002'	Document Type of the Document
LLIDF	5	X'0009C70101'	Profile CGSCGID Introducer
parameter	4	X'csidcpid'	Character Set and Code Page

The above base subprofile contains the DIA required parameters. Other parameters can be present and are entered as specified by the IDP architecture.

Additional subprofiles can be specified at this point in the document unit. These can include product private subprofiles and DIA application subprofiles.

LLIDF	5	X'0005CB0201'	Document Content Introducer
-------	---	---------------	-----------------------------

The type 2 document content introducer specifies that there is no document content to follow.

IDD (Format 3) - document-descriptor document

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>	<u>NAME</u>
LLIDF	5	X'nnnnC90301'	Document Unit Introducer
parameter	2	X'0008'	Document Type Identifier (document-descriptor document)
	13	X'n...n'	System Code
LLIDF	5	X'0005CB0101'	Document Content Introducer

The document-descriptor document content begins here and occupies the space remaining as specified by the LL (length) bytes of the document unit introducer.

IDD (Format 3) - Selected Document Descriptors and Document Content

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>	<u>NAME</u>
LLIDF	5	X'nnnnC90301'	Document Unit Introducer
parameter	2	X'0002'	Document Type Identifier (Final Form Text Document Content Architecture)
	13	X'n...n'	System Code
LLIDF	5	X'nnnnCA0301'	Interchange Document Profile Introducer
LLIDF	5	X'nnnnCA0401'	Subprofile Introducer
LLIDF	5	X'0007C70601'	Document Type Introducer
parameter	2	X'0002'	Final-Form-Text Identifier

The values of the requested descriptors would follow the subprofile introducer. Document type is used as an example.

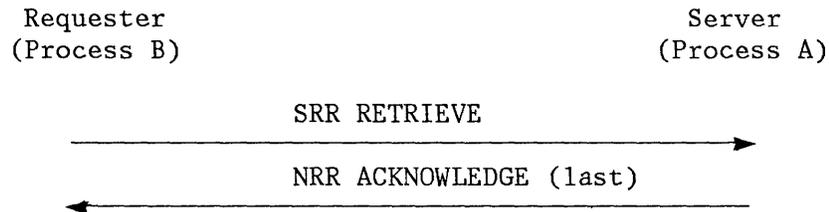
LLIDF	5	X'0005CB0101'	Document Content Introducer
-------	---	---------------	-----------------------------

The final-form-text document content begins here and occupies the space remaining as specified by the LL (length) bytes of the document unit introducer.

A null document-descriptor document is to be returned if a document referred to by the IDENTIFIED-DATA (Format 3) operand cannot be found.

- Scenario 2 - Exception Conditions.

Exception conditions detected during the RETRIEVE command processing are replied to with an ACKNOWLEDGE command that contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to the DIA commands are described in Appendix F. The following exception conditions are specific to the RETRIEVE command and are detected and reported in addition to the general exception conditions.

- The SOURCE-PASSWORD operand is present, and the SOURCE-ADDRESS (Format 1) operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
 Exception Code = X'C20708'
 Exception data = LLIDF of SOURCE-ADDRESS (Format 1) operand

- The SOURCE-ADDRESS operand specifies a requester different from the command sender, the specified requester does not have affinity with the command sender, the specified requester has a password, and the SOURCE-PASSWORD operand is not specified.

Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
 Exception Code = X'C30708'
 Exception data = LLIDF of SOURCE-PASSWORD operand

- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 Exception Code = X'C30209'
 Exception data = LLIDF of SOURCE-ADDRESS operand and data

- The ORIGINATING-NODE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 Exception Code = X'C30209'
 Exception data = LLIDF of ORIGINATING-NODE-ADDRESS operand and data

- The SOURCE-PASSWORD operand contains an invalid authorization key.
 Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
 Exception Code = X'C30509'
 Exception data = LLIDF of SOURCE-PASSWORD operand and data
- The document specified in the IDENTIFIED-DATA operand cannot be found.
 Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
 Exception Code = X'C40709'
 Exception data = LLIDF of IDENTIFIED-DATA operand and data
- The specified requester is not authorized to have access to the document specified in the IDENTIFIED-DATA operand.
 Exception = Catastrophic, Process, Unauthorized-Access, Operand-Value
 Exception Code = X'C40309'
 Exception data = LLIDF of SOURCE-ADDRESS operand and data
- The specified document cannot be returned in an allowable document type specified in the SIGN-ON command for the DIA session.
 Exception = Catastrophic, Process, Data-Not-Supported,
 Document-Content-Control
 Exception Code = X'C40211'
- The specified document cannot be returned in an allowable graphic character set specified in the SIGN-ON command for the DIA session.
 Exception = Catastrophic, Process, Data-Not-Supported, Document-Content-Data
 Exception Code = X'C40212'
- The requested document cannot be retrieved.
 Exception = Catastrophic, Process, Resource-Not-Available, Document-Unit
 Exception Code = X'C4040C'
 Exception Data = None

Support Considerations

The types of data returned for the RETRIEVE (Format 4) command must be limited to the values specified by the DOCUMENT-TYPE or GCID operands of the SIGN-ON command at DIA session establishment.

SEARCH

<u>Command</u>	<u>Operands</u>
SEARCH	SEARCH-REQUEST-NAME [,SEARCH-DATA] [,ORIGINATING-NODE-ADDRESS] [,SOURCE-ADDRESS] [,SOURCE-PASSWORD] [,RETRIEVE-COUNT] [,DESCRIPTOR-CONTENT-DEFINITION] [,TIME-LIMIT] [,SELECT-LIMIT] [,SEARCH-OPTION]

The SEARCH command locates documents in the document library that satisfy requester-specified search criteria. The search criteria consist of combinations of document profile parameters that define search parameters. The search process locates only those documents to which the requester has access authority. If no search criteria are specified, the selection process is a request to create a list of all documents in the library that are owned or delegate-owned by the requester. The search process creates a list of references or pointers to the selected documents and preserves the list with the search request name (qualified with the SOURCE-ADDRESS operand and the ORIGINATING-NODE-ADDRESS operand).

Operand Descriptions

SEARCH-REQUEST-NAME

The SEARCH-REQUEST-NAME (Format 1) operand specifies the 1- to 8-byte name used by the search process to identify and preserve the results of the selected document references or pointers.

SEARCH-DATA

The SEARCH-DATA operand, when present, specifies the arguments that are used by the search process to identify documents in the document library; either SEARCH-DATA (Format 41 or Format 42) are valid. The search process selection procedure compares the search arguments with the values of identified document profile parameters. All documents in the document library that match the search criteria and that the requester has authority to access are selected. References to the selected documents are placed in the Search Result List.

If this operand is omitted, the search process compares the primary owner and owner-delegates to the source ID or source name that is specified in the SOURCE-ADDRESS operand of the command. If the SOURCE-ADDRESS operand is not specified, the comparison is made with the SIGN-ON-ID operand value of the command requester. In this case, all documents in the document library that are owned and delegate-owned are selected and references to them are placed in the Search Result List.

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand, if present, specifies the 1- to 8-character group address token of the requester that initiated this request. If this operand is omitted, the group address token of the command sender is assumed.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand, if present, specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS (Format 1 or Format 42) is valid. If this operand is omitted, the element address token of the command sender is assumed.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated this request. The SOURCE-PASSWORD (Format 1) operand is conditionally required if the SOURCE-ADDRESS (Format 1) operand is specified. The SOURCE-PASSWORD (Format 1) operand is required if all of the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different from the command sender.
- The specified requester does not have affinity with the command sender.

- The specified requester has a password.

Otherwise, the SOURCE-PASSWORD operand should be omitted.

RETRIEVE-COUNT

The RETRIEVE-COUNT (Format 1) operand, when present, specifies the maximum number of document descriptors that are returned in response to a SEARCH command. A RETRIEVE-COUNT of 0 specifies that the requester wants only a count of the number of documents which were selected by the search process. When this operand is omitted, a default value of 1 is assumed.

DESCRIPTOR-CONTENT-DEFINITION

The DESCRIPTOR-CONTENT-DEFINITION (Format 41) operand, when present, specifies the introducer IDF values of the document profile parameters that are returned in a response to a SEARCH command. The descriptors are returned in a document-descriptor document. If this operand is omitted, only the document name is returned in the document descriptor document.

TIME-LIMIT

The TIME-LIMIT (Format 1) operand, if present, specifies the maximum number of minutes (from 1 to 1440) that the command server allows the search process to execute. If this operand is omitted, the command server determines the maximum processing time.

SELECT-LIMIT

The SELECT-LIMIT (Format 1) operand, if present, specifies the maximum number of documents (from 1 to 32767) that the requester wants the command server to select from those that match the search criteria. If this operand is omitted, the command server determines the maximum number.

SEARCH-OPTION

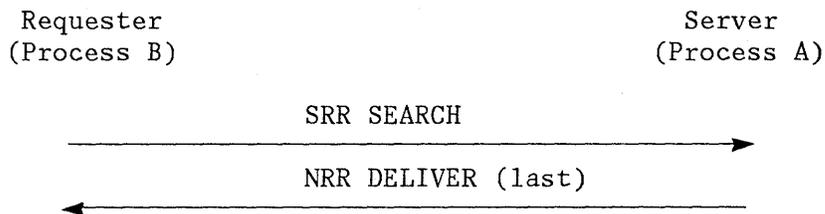
The SEARCH-OPTION (Format 1) operand, if present, defines the scope of the search to be taken. The scoping specifies a search of either all documents that the requester owns or has access to. When a scope other than owned is specified, the SEARCH-DATA operand is required. If the SEARCH-OPTION operand is omitted, the search scope is determined by the SEARCH-DATA operand.

Request/Reply Protocol

The following scenarios illustrate the possible replies to the SEARCH command.

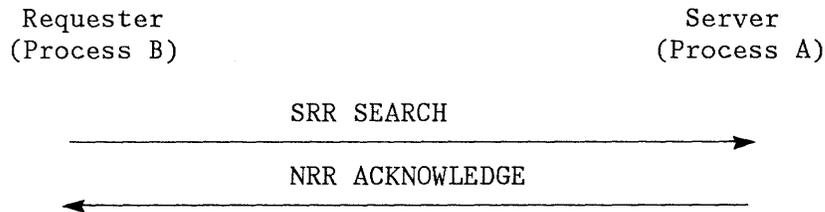
- Scenario 1 - A document-descriptor document is returned to the requester.

When the number of documents selected by the search process is equal to or less than the number specified in the SEARCH command RETRIEVE-COUNT operand, a DIA-defined document-descriptor document is sent to the requester in a DIU document unit with a DELIVER command. The contents of the document-descriptor document are either the user-specified fields, as defined in the SEARCH command DESCRIPTOR-CONTENT-DEFINITION operand, or only the document name specified in the IDP base subprofile. The document-descriptor document contains an entry for each document that the search process selected. The Document Library Services server preserves the references to the search-selected documents (identified by the descriptors) for subsequent processing requests.



- Scenario 2 - Normal conclusion but no documents selected.

An ACKNOWLEDGE command that contains a COUNT operand specified in the REPLY-DATA operand replies to the normal conclusion of the search process that does not select any documents. If no document satisfies the search criteria, the COUNT operand contains a value of 0. When the search process selects a greater number of documents than the value specified in the RETRIEVE-COUNT operand, or when a RETRIEVE-COUNT value of 0 was specified, the value in the COUNT operand is the actual number of documents that satisfy the search criteria.

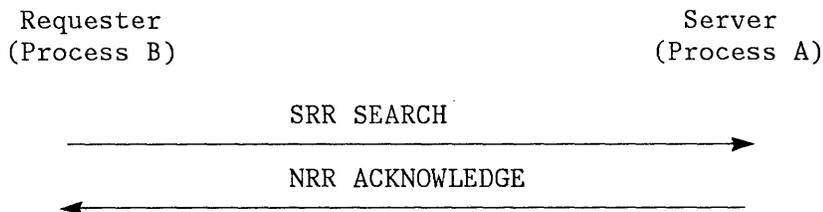


The REPLY-DATA operand containing the COUNT operand has the following format:

REPLY-DATA	COUNT Operand
LLIDF	LLIDF DATA
X'000CC34501'	X'0007C33E01' X'nnnn'

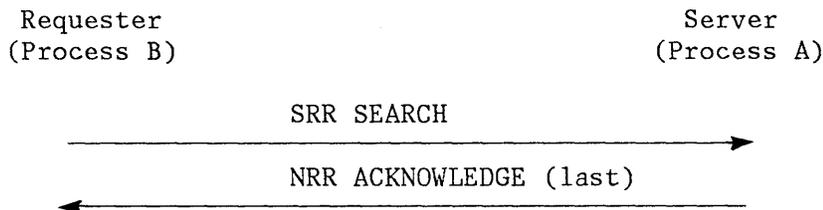
- Scenario 3 - Normal conclusion, but a limit is exceeded.

When the search process is terminated because the time limit or select limit is exceeded, the ACKNOWLEDGE operand contains the appropriate exception code in the EXCEPTION-CODE operand and the count of the number of documents specified in the COUNT operand of the REPLY-DATA operand, if any, selected up to the point of termination.



- Scenario 4 - Exception Conditions.

An ACKNOWLEDGE command reports exception conditions detected during the SEARCH command processing. The exception condition is in the EXCEPTION-CODE operand.



Exception Conditions

See "DIU General Exception Conditions" in Appendix F for a description of common exception conditions that are common to all DIA commands.

The following exception conditions are specific to the SEARCH command.

- The SOURCE-PASSWORD operand is present, and the SOURCE-ADDRESS (Format 1) operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
 Exception Code = X'C20708'
 Exception data = LLIDF of SOURCE-ADDRESS (Format 1) operand

- The SOURCE-ADDRESS operand specifies a requester different from the command sender, the specified requester does not have affinity with the command sender, the specified requester has a password, and the SOURCE-PASSWORD operand is not specified.

Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
Exception Code = X'C30708'
Exception data = LLIDF of SOURCE-PASSWORD operand

- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

- The ORIGINATING-NODE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of ORIGINATING-NODE-ADDRESS operand and data

- The SOURCE-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception data = LLIDF of SOURCE-PASSWORD operand and data

- The search process exceeds the maximum time specified by the TIME-LIMIT operand or the processing server's maximum search execution time.

Exception = Severe, Process, Time-Out, Command
Exception Code = X'841307'

- The search process exceeds the maximum number of documents specified by the SELECT-LIMIT operand or the processing server's maximum number of search-selected document count.

Exception = Severe, Process, Range-Exceeded, Command
Exception Code = X'841107'

CHAPTER 10. FUNCTION SETS AND COMMANDS: FILE TRANSFER SERVICES

The commands of File Transfer Services request the transfer of documents between DIA processes. This chapter summarizes these commands.

- The DELIVER command transports a document from a server node to a requester node.
- The FILE (Format 2) command files a document in a specified library.
- The RETRIEVE (Format 4) command returns a copy of an identified document from a specified library to the command requester.

Figure 33 shows the commands in the File Transfer Services function set.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
FILE (Format 2)	SRR	receive	send
RETRIEVE (Format 4)	SRR	receive	send
DELIVER	NRR	send	receive
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive
SIGN-OFF	NRR	send/rec	send/rec

Figure 33. Function Set 11

DELIVER

<u>Command</u>	<u>Operands</u>
DELIVER	IDENTIFIED-DATA, CORRELATION

The DELIVER command is a replying command that returns a document from a File Transfer Service command server to the File Transfer Service command requester. For example, the DELIVER command returns a document to a requester in reply to a RETRIEVE command.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be delivered. Only Format 1 of this operand is valid in support of the DELIVER command.

The IDENTIFIED-DATA (Format 1) operand specifies that the document to be delivered is located in the DIU document unit.

CORRELATION

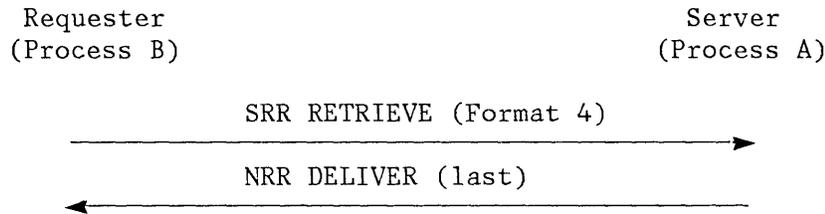
The CORRELATION (Format 1) operand correlates a replying command to a previously-sent request. The CORRELATION (Format 1) operand identifies the request to which this DELIVER command replies and indicates that no further replying commands will be sent.

Request/Reply Protocol

The following scenarios illustrate the possible replies to the DELIVER command:

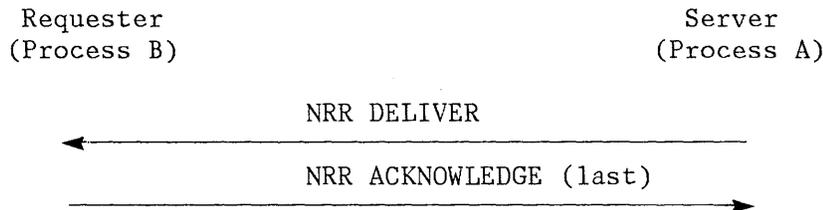
- Scenario 1 - Single Reply

The following is an example of a request and a single reply.



- Scenario 2 - Exception Conditions

When exception conditions occur during the processing of the DELIVER command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to the DIA commands are described in Appendix F. The receiver rejects the DELIVER command if any of the following conditions exist:

- The receiver is not in a state in which it can receive and send data.
Exception = Catastrophic, Session, Intervention-Required, Unknown
Exception Code = X'C11217'
- The IDENTIFIED-DATA operand is not in the DELIVER command.
Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
- The CORRELATION operand does not refer to a command previously sent by the receiver.
Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of CORRELATION operand
- The IDENTIFIED-DATA operand refers to nonexistent data.
Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of IDENTIFIED-DATA operand
- The document unit type is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit
Exception Code = X'C3020C'
Exception Data = LLIDF of document unit introducer
- The document content introducer type is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported,
Document-Content-Introducer
Exception Code = X'C30210'
Exception Data = LLIDF of document content introducer
- The document type in the document unit ID is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit ID
Exception Code = X'C3020D'
Exception Data = LLIDF and data of document unit ID
- The document type parameter in the base subprofile is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported,
Document-Profile-Parameter
Exception Code = X'C3020F'
Exception Data = LLIDF and data of the document type parameter

- The document profile is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Profile
Exception Code = X'C3020E'
Exception Data = LLIDF of document profile
- Receiving process resources are unavailable.
Exception = Catastrophic, Process, Resource-Not-Available, Document-Unit
Exception Code = X'C4040C'
- The receiving process cancels delivery of the data.
Exception = Catastrophic, Process, Cancelled, Command
Exception Code = X'C41407'

FILE (Format 2)

<u>Command</u>	<u>Operands</u>
FILE (Format 2)	IDENTIFIED-DATA ,LIBRARY-NAME [,FILE-OPTION]

The FILE (Format 2) command stores information into a specified library. The types of information include a document with an associated document profile and a document profile alone that refers to externally-stored documents. This information is submitted with the request in the DIU document unit.

If a duplicate document name condition exists in the target library, the action taken depends on the value of the FILE-OPTION operand. The options are to either reject or replace the document.

Because these are user libraries, no unique Library-Assigned Document Name is returned.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be filed. Only Format 1 of the IDENTIFIED-DATA operand is valid in support of the FILE (Format 2) command.

The IDENTIFIED-DATA (Format 1) operand specifies that the document to be delivered is located in the DIU document unit.

LIBRARY-NAME

The LIBRARY-NAME operand specifies the 1- to 44-character address token of the library into which the particular document goes. The FILE (Format 2) command uses either LIBRARY-NAME operand (Format 1 or Format 41). The LIBRARY-NAME operand (Format 41) specifies the CGCSGID of the library name.

FILE-OPTION

The FILE-OPTION (Format 1) operand, if present, specifies the action taken if a duplicate document name condition exists. The possible actions are to reject the request or replace the document. Without this operand, any request to file a document with a duplicate name will default to the reject option.

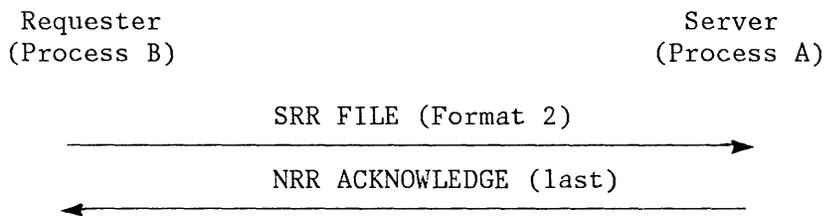
Request/Reply Protocol

The following scenarios illustrate the possible replies to the FILE (Format 2) command:

- Scenario 1 - Normal Condition

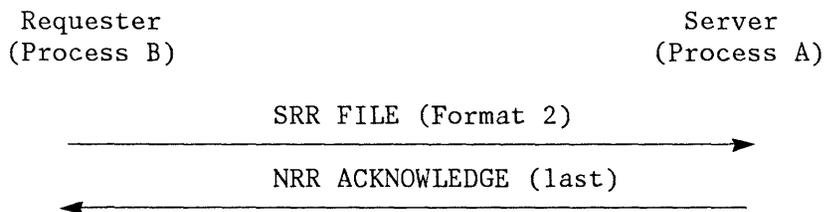
The filed document is a private document, and the requester of the FILE (Format 2) command is the primary owner of the document.

The reply command to a FILE (Format 2) command is an ACKNOWLEDGE command that is sent to the requester when the document is filed in the target library. The EXCEPTION-CODE operand indicates no exception condition.



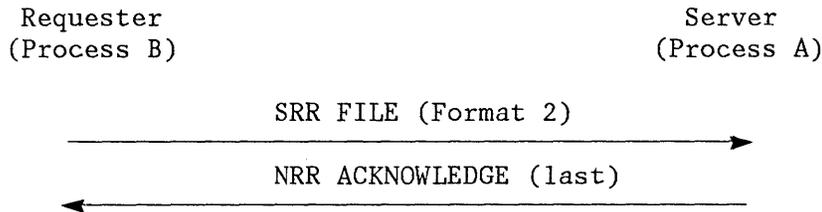
- Scenario 2 - Duplicate Name Condition

If the FILE-OPTION operand is omitted from the FILE (Format 2) command, or if it specifies the reject option and a duplicate document name exists in the library, the server rejects the file request. The reply is an ACKNOWLEDGE command that contains the exception condition in the EXCEPTION-CODE operand.



- Scenario 3 - Exception Condition.

When exception conditions occur during the processing of the FILE (Format 2) command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to the DIA commands are described in Appendix F. In addition to the general exception conditions, the following exception conditions apply to the FILE (Format 2) command.

- The server cannot find the document specified in the IDENTIFIED-DATA operand.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
 Exception Code = X'C40709'
 Exception data = LLIDF of IDENTIFIED-DATA operand and data

- The document name exists, and the File-Option indicates reject.

Exception = Information, Process, Data-Found, Document-Profile-Parameter
 Exception Code = X'04090F'
 Exception data = LLIDF of document name subprofile parameter and data

- The server cannot find the library specified in the LIBRARY-NAME operand.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
 Exception Code = X'C40709'
 Exception data = LLIDF of LIBRARY-NAME operand and data

- The receiving process cancelled the command request.

Exception = Catastrophic, Process, Cancelled, Command
 Exception Code = X'C41407'
 Exception data = LLIDF of the FILE (Format 2) command.

RETRIEVE (Format 4)

<u>Command</u>	<u>Operands</u>
RETRIEVE (Format 4)	IDENTIFIED-DATA ,LIBRARY-NAME

The RETRIEVE (Format 4) command retrieves a copy of the identified information from a specific library.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be retrieved. Format 2 and Format 41 of the IDENTIFIED-DATA operand are valid in support of RETRIEVE (Format 4).

The IDENTIFIED-DATA (Format 2) operand identifies the document to be retrieved by means of a 1- to 44-character document name.

The IDENTIFIED-DATA (Format 41) operand identifies the document to be retrieved by means of a 1- to 44-character document name whose CGCSGID specified explicitly in the operand.

LIBRARY-NAME

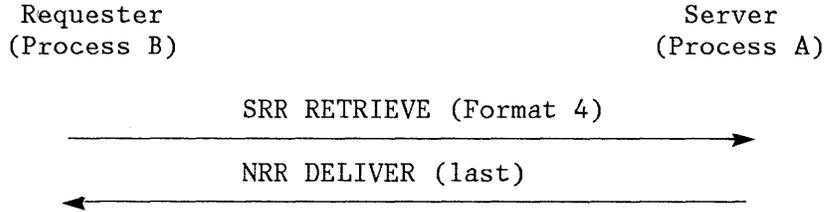
The LIBRARY-NAME operand specifies the 1- to 44-character address token of the library in which the particular document can be found. Either LIBRARY-NAME operand (Format 1 or Format 41) applies. LIBRARY-NAME operand (Format 41) specifies the CGCSGID of the library name.

Request/Reply Protocol

The following scenarios illustrate the possible replies to the RETRIEVE (Format 4) command:

- Scenario 1 - Normal Condition

The normal reply to a RETRIEVE (Format 4) command is a DELIVER command that the server sends at the conclusion of processing.



The DELIVER DIU document unit contains the requested information. The following example defines the contents of the DELIVER DIU document unit in reply to the RETRIEVE (Format 4) request. Final-Form-Text Document Content Architecture (document type 2) is the type of document being referred to. The example assumes that the document unit need not be segmented.

Document and Document Profile

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>	<u>NAME</u>
LLIDF	5	X'nnnnC90301'	Document Unit Introducer
parameter	2	X'0002'	Document Type Identifier (Final-Form-Text Document Content Architecture)
	13	X'n...n'	System Code
LLIDF	5	X'nnnnCA0301'	Interchange Document Profile Introducer
LLIDF	5	X'nnnnCA0401'	Base Subprofile Introducer
LLIDF	5	X'nnnnC70001'	User-Assigned Document Name (UADN) Introducer
parameter	1-44	X'n...n'	User-Assigned Document Name
LLIDF	5	X'0007C70601'	Document Type Introducer
parameter	2	X'0002'	Final-Form-Text Identifier
LLIDF	5	X'0009C70101'	Profile CGCSGID Introducer
parameter	4	X'csidcpid'	Character Set and Code Page

The above base subprofile contains only the DIA required parameters. The base subprofile of an interchange document profile can also include other parameters.

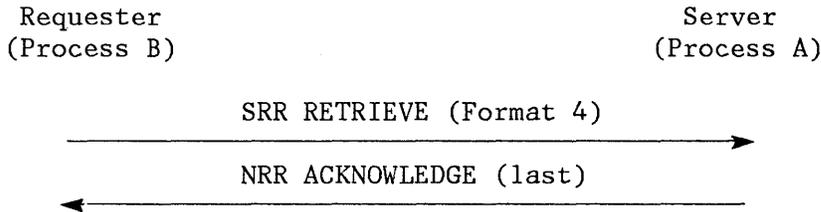
The interchange document profile can also include other subprofiles.

LLIDF 5 X'0005CB0101' Document Content Introducer

The final-form-text document content begins here and occupies the space remaining as specified by the LL bytes (length) of the document unit introducer.

- Scenario 2 - Exception Conditions.

When exception conditions occur during the processing of the RETRIEVE (Format 4) command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to the DIA commands are described in Appendix F. In addition to the general exception conditions, the following exception conditions apply to the RETRIEVE (Format 4) command.

- The server cannot find the document specified in the IDENTIFIED-DATA operand.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
 Exception Code = X'C40709'
 Exception data = LLIDF of IDENTIFIED-DATA operand and data

- The server cannot find the library specified in the LIBRARY-NAME operand.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
 Exception Code = X'C40709'
 Exception data = LLIDF of LIBRARY-NAME operand and data

- The server cannot return the specified document in an allowable document type specified in the SIGN-ON command for the DIA session.

Exception = Catastrophic, Process, Data-Not-Supported,
 Document-Content-Control
 Exception Code = X'C40211'

- The server cannot return the specified document in an allowable graphic character set specified in the SIGN-ON command for the DIA session.

Exception = Catastrophic, Process, Data-Not-Supported, Document-Content-Data
 Exception Code = X'C40212'

- The receiving process cancelled the command request.

Exception = Catastrophic, Process, Cancelled, Command
 Exception Code = X'C41407'
 Exception data = LLIDF of the RETRIEVE (Format 4) command.

Support Considerations

The types of data returned for the RETRIEVE (Format 4) command must be limited to the values specified by the DOCUMENT-TYPE or GCID operands of the SIGN-ON command at DIA session establishment.

CHAPTER 11. FUNCTION SETS AND COMMANDS: DOCUMENT DISTRIBUTION SERVICES

Document Distribution Services deliver DIUs from source nodes to recipient nodes within an office system network. Document Distribution Services can distribute the information between a source node and a recipient node during a single DIA session, or by routing their information through office system nodes for subsequent delivery to recipient nodes. This section includes a brief summary of each command.

- The CANCEL-DISTRIBUTION command cancels a distribution of status information or cancels the delivery of distributed documents or messages.
- The DELIVER command transports documents and messages from an office system node to a source or recipient node. The DELIVER command also transports documents and messages directly between a source node and a recipient node without an intervening office system node.
- The LIST command requests delivery of a list of documents and messages queued for delivery at an office system node for a recipient node or a list of the information concerning the status of distribution requests previously submitted.
- The OBTAIN command requests delivery of one or more documents and/or messages scheduled for delivery to the requester.
- The PROCESS-BIT-STRING command requests an office system node to interpret a bit-stream representation of a DIA function request and perform the requested operation.
- The REQUEST-DISTRIBUTION command transports documents from a source node to an office system node for distribution to the specified recipient nodes. These documents can be submitted with the command, located in the command server's document library, or located in a library accessible to the command server. Messages can be submitted with the command.
- The STATUS-LIST command notifies the recipient node that one or more documents are available from the distribution system or that information about the progress of previous distribution requests is available.

The following figures (Figure 34 through Figure 39) show these commands as Document Distribution Services function sets.

Function set 2 contains the DIA commands necessary to deliver information from an OSN destination node to a recipient node in a solicited environment. In a solicited environment, the recipient node must specifically request that the OSN deliver the information.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
DELIVER	SRR	send	receive
OBTAIN	SRR	receive	send
LIST	SRR	receive	send
STATUS-LIST	NRR	send	receive
CANCEL-DISTRIBUTION	SRR	receive	send
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive
SIGN-OFF	NRR	send/rec	send/rec

Figure 34. Function Set 2

Function set 3 contains the DIA commands used to deliver information from an OSN to a recipient node in an unsolicited environment. In an unsolicited environment, the OSN delivers the information to the recipient node without being specifically requested to do so.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
DELIVER	SRR	send	receive
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive

Figure 35. Function Set 3

Function set 4 contains the DIA commands necessary to send documents and DIA function requests to an OSN from an image source node.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
PROCESS-BIT-STRING	SRR	receive	send
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive

Figure 36. Function Set 4

Function set 5 contains the DIA commands necessary to initiate and control document distribution requests from a source node to an office system node.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
CANCEL-DISTRIBUTION	SRR	receive	send
REQUEST-DISTRIBUTION	SRR	receive	send
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive
SIGN-OFF	NRR	send/rec	send/rec

Figure 37. Function Set 5

Function set 6 contains the DIA commands necessary to send documents between image source and recipient nodes without going through any intermediate office system nodes.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
DELIVER	SRR	send	receive
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	send/rec	send/rec
SIGN-ON Reply	NRR	send/rec	send/rec

Figure 38. Function Set 6

Function set 7 contains the DIA commands necessary to distribute information between source and recipient nodes without going through any intermediate office system nodes.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
DELIVER	SRR	send	receive
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	send/rec	send/rec
SIGN-ON Reply	NRR	send/rec	send/rec
SIGN-OFF	NRR	send/rec	send/rec

Figure 39. Function Set 7

CANCEL-DISTRIBUTION

<u>Command</u>	<u>Operands</u>
CANCEL-DISTRIBUTION	CANCEL-ACTION [,DOCUMENT-PASSWORD] [,RECIPIENT-ADDRESS SOURCE-ADDRESS] [,RECIPIENT-PASSWORD SOURCE-PASSWORD]

The CANCEL-DISTRIBUTION command cancels delivery of distribution information queued for delivery to either the requester, a specific recipient, or recipients on whose behalf the requester has the authority to act. See "Affinity" on page 17 for a discussion of a requester acting on behalf of some other recipient.

The types of distribution information that can be cancelled include:

- Personal and nonpersonal documents
- Personal and nonpersonal documents with appended messages
- Personal and nonpersonal messages only
- Status information about previous distribution requests.

A recipient at a recipient node uses the CANCEL-DISTRIBUTION command to cancel a distribution queued for delivery. The term distribution is either a document (with or without an appended message) or a message only. The signed-on recipient can cancel a distribution in any one of the following ways:

- Sending a CANCEL-DISTRIBUTION command without a RECIPIENT-ADDRESS operand cancels the distribution only for the signed-on recipient.
- Sending a CANCEL-DISTRIBUTION command with the RECIPIENT-ADDRESS operand and no RECIPIENT-PASSWORD operand cancels the distribution for the specified recipient. If the specified recipient does not have affinity with the signed-on recipient, the command is rejected. The signed-on recipient can supply its own recipient address.
- Sending a CANCEL-DISTRIBUTION with both the RECIPIENT-ADDRESS and RECIPIENT-PASSWORD operands, including the case when the specified recipient address does not have a sign-on password defined, cancels a distribution for a recipient that does not have affinity with the signed-on recipient.

In all cases involving personal distributions, the CANCEL-DISTRIBUTION command must include the DOCUMENT-PASSWORD operand if the recipient of a distribution has a document password. The CANCEL-DISTRIBUTION command need not include the DOCUMENT-PASSWORD operand if the recipient does not have a defined sign-on password.

A source at a source node uses the CANCEL-DISTRIBUTION command to cancel status information enqueued for delivery. The signed-on source can cancel status information in any one of the following ways:

- Sending a CANCEL-DISTRIBUTION command without a SOURCE-ADDRESS operand cancels the status information for the signed-on source only.
- Sending a CANCEL-DISTRIBUTION command with a SOURCE-ADDRESS operand and no SOURCE-PASSWORD operand cancels the status information for the specified source only. If the specified source does not have affinity with the signed-on source, the command is rejected. The signed-on source can supply its own source address.
- Sending a CANCEL-DISTRIBUTION with both the SOURCE-ADDRESS and SOURCE-PASSWORD operands cancels the status information for a source that does not have affinity with the signed-on source. If the CANCEL-DISTRIBUTION command includes the SOURCE-PASSWORD operand, the specified password is verified.

Operand Descriptions

CANCEL-ACTION

The CANCEL-ACTION (Format 1) operand identifies the distribution and defines the action to be taken. The Action-Code field and the Distribution-Document-Name field must appear in the CANCEL-ACTION operand.

DOCUMENT-PASSWORD

The DOCUMENT-PASSWORD (Format 1) operand, if present, specifies a 1- to 8-byte personal document authorization key associated with the recipients. The CANCEL-DISTRIBUTION command must include this operand if the requester wants to cancel personal documents and a document password has been defined for that requester. If the command does not include the DOCUMENT-PASSWORD operand, no personal documents are cancelled even if a document password is defined for that recipient. The command must include the RECIPIENT-PASSWORD operand for those users that do not have a defined sign-on password. DIA ignores this operand if the COD option was specified in the CANCEL-ACTION operand.

The DOCUMENT-PASSWORD operand must be the signed-on recipient's document password unless the RECIPIENT-ADDRESS operand is specified. In that case, the DOCUMENT-PASSWORD operand must be the specified recipient's document password.

RECIPIENT-ADDRESS

The RECIPIENT-ADDRESS operand, if present, specifies the address token of the recipient; either RECIPIENT-ADDRESS operand (Format 1 or Format 42) applies.

The CANCEL-DISTRIBUTION command includes the RECIPIENT-ADDRESS operand only when the CANCEL-ACTION operand includes the Deliver option in the Action-Code field.

RECIPIENT-PASSWORD

The RECIPIENT-PASSWORD (Format 1) operand, if present, provides a 1- to 8-character access authorization key for the intended recipient.

If the CANCEL-DISTRIBUTION command includes the RECIPIENT-PASSWORD operand, it must also include the RECIPIENT-ADDRESS operand.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand, if present, specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS operand (Format 1 or Format 42) is valid.

The SOURCE-ADDRESS operand is required if the source wants to cancel status information for a distribution previously sent COD and the source is different from the signed-on source.

In the CANCEL-DISTRIBUTION command, the SOURCE-ADDRESS operand applies only when the CANCEL-ACTION operand includes the COD option in the Action-Code field.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated this request. The SOURCE-PASSWORD (Format 1) operand is conditionally required if the SOURCE-ADDRESS (Format 1) is specified. The CANCEL-DISTRIBUTION command must include the SOURCE-PASSWORD (Format 1) operand if all the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different from the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

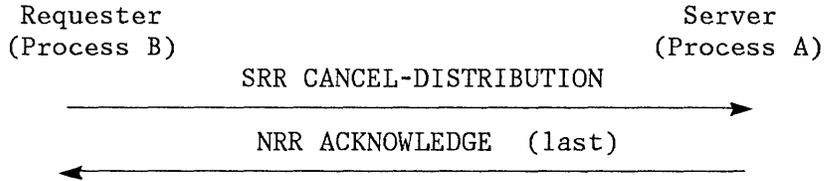
Otherwise, the command should not include the SOURCE-PASSWORD operand.

Request/Reply Protocol

The following scenarios illustrate possible replies to the CANCEL-DISTRIBUTION command:

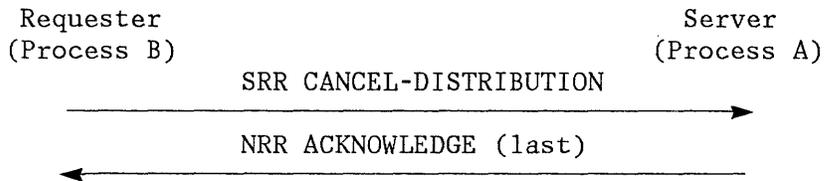
- Scenario 1 - Normal Completion

An ACKNOWLEDGE command is sent to the command requester when the cancellation is the normal reply to a CANCEL-DISTRIBUTION command.



- Scenario 2 - Exception Conditions.

When exception conditions occur during the processing of the CANCEL-DISTRIBUTION command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions."

In addition to the general exception conditions, the following exception conditions are specific to the CANCEL-DISTRIBUTION command.

- The SOURCE-PASSWORD operand is present, but the SOURCE-ADDRESS operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of SOURCE-ADDRESS operand

- The SOURCE-ADDRESS operand specifies a source different from the signed-on source; the specified source does not have affinity with the signed-on source. The specified source has a password, but the CANCEL-DISTRIBUTION command does not include SOURCE-PASSWORD operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of SOURCE-PASSWORD operand

- The RECIPIENT-PASSWORD operand is present, but the RECIPIENT-ADDRESS operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of RECIPIENT-ADDRESS operand

- The RECIPIENT-ADDRESS operand specifies a recipient different from the signed-on recipient; the specified recipient does not have affinity with the signed-on recipient. The specified recipient has a password, but the CANCEL-DISTRIBUTION command does not include the RECIPIENT-PASSWORD operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of RECIPIENT-PASSWORD operand

- The CANCEL-ACTION operand does not include the Distribution-Document Name.

Exception = Catastrophic, Syntax, Data-Not-Found, Operand-Value
Exception Code = X'C20709'
Exception Data = LLIDF and data of CANCEL-ACTION operand

- The CANCEL-DISTRIBUTION command specifies the SOURCE-ADDRESS operand, but the CANCEL-ACTION operand specifies the Deliver option.

Exception = Catastrophic, Syntax, Data-Not-Supported, Command-Operand
Exception Code = X'C20208'
Exception Data = LLIDF and data of SOURCE-ADDRESS operand

- The CANCEL-DISTRIBUTION command specifies the RECIPIENT-ADDRESS operand, but the CANCEL-ACTION operand specifies the COD option.

Exception = Catastrophic, Syntax, Data-Not-Supported, Command-Operand
Exception Code = X'C20208'
Exception Data = LLIDF and data of RECIPIENT-ADDRESS operand

- The CANCEL-ACTION operand contains an undefined action value.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception Data = LLIDF and data of CANCEL-ACTION operand

- The CANCEL-ACTION operand specifies a non-existent Distribution-Document Name, or the distribution is already cancelled.

Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of CANCEL-ACTION operand

- The RECIPIENT-ADDRESS operand contains an invalid name.

Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
Exception Code = X'C30309'
Exception Data = LLIDF and data of RECIPIENT-ADDRESS operand

- The distribution recipient is different from the signed-on recipient, but the CANCEL-DISTRIBUTION command does not include the RECIPIENT-ADDRESS operand.

Exception = Catastrophic, Semantic, Unauthorized-Access, Command-Operand
Exception Code = X'C30308'
Exception Data = LLIDF of RECIPIENT-ADDRESS operand

- The RECIPIENT-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception Data = LLIDF and data of RECIPIENT-PASSWORD operand

- When cancelling a personal document, the CANCEL-DISTRIBUTION command does not include the DOCUMENT-PASSWORD operand.

Exception = Catastrophic, Semantic, Unauthorized-Access, Command-Operand
Exception Code = X'C30308'
Exception Data = LLIDF of DOCUMENT-PASSWORD operand

- The DOCUMENT-PASSWORD operand contains an invalid authorization key.
 Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
 Exception Code = X'C30509'
 Exception Data = LLIDF and data of DOCUMENT-PASSWORD operand
- The SOURCE-ADDRESS operand contains an invalid name.
 Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
 Exception Code = X'C30309'
 Exception Data = LLIDF and data of SOURCE-ADDRESS operand
- The distribution source is different from the signed-on source, but the CANCEL-DISTRIBUTION command does not include the SOURCE-ADDRESS operand.
 Exception = Catastrophic, Semantic, Unauthorized-Access, Command-Operand
 Exception Code = X'C30308'
 Exception Data = LLIDF of SOURCE-ADDRESS operand
- The SOURCE-PASSWORD operand contains an invalid authorization key.
 Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
 Exception Code = X'C30509'
 Exception Data = LLIDF and data of SOURCE-PASSWORD operand
- The RECIPIENT-PASSWORD operand length exceeds the processing capacity of the receiver.
 Exception = Catastrophic, Process, Length-Invalid, Command-Operand
 Exception Code = X'C40F08'
 Exception Data = LLIDF and data of RECIPIENT-PASSWORD operand
- The DOCUMENT-PASSWORD operand length exceeds the processing capacity of the receiver.
 Exception = Catastrophic, Process, Length-Invalid, Command-Operand
 Exception Code = X'C40F08'
 Exception Data = LLIDF and data of DOCUMENT-PASSWORD operand
- The SOURCE-PASSWORD operand length exceeds the processing capacity of the receiver.
 Exception = Catastrophic, Process, Length-Invalid, Command-Operand
 Exception Code = X'C40F08'
 Exception Data = LLIDF and data of SOURCE-PASSWORD operand
- The server cannot find a status for the indicated distribution.
 Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
 Exception Code = X'C40709'
 Exception Data = None

DELIVER

<u>Command</u>	<u>Operands</u>
DELIVER	[IDENTIFIED-DATA] [,SOURCE-ADDRESS] [,ATTRIBUTE-LIST] [,RECIPIENT-ADDRESS] [,ORIGINATING-NODE-ADDRESS] [,MESSAGE] [,DISTRIBUTION-ID] [,CORRELATION]

The DELIVER command transmits distribution information from a source node or office system node to a recipient node. For example, the DELIVER command returns a document from an office system node to a recipient node in reply to an OBTAIN command.

The types of information that the DELIVER command transmits are documents, messages, or documents with an appended message.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand, if present, identifies the document to be delivered. Only Format 1 of the IDENTIFIED-DATA operand is valid in support of the DELIVER command. If the DELIVER command does not include the IDENTIFIED-DATA operand, the server is not delivering a document; therefore, the message operand becomes a required, rather than an optional operand.

The IDENTIFIED-DATA (Format 1) operand specifies that the document to be retrieved is in the DIU document unit.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS (Format 1 or Format 42) is valid. If the DELIVER command does not include this operand, the element address token of the command sender is the default value.

ATTRIBUTE-LIST

The distribution originator (source node) uses the ATTRIBUTE-LIST (Format 1) to specify the processing characteristics of the distribution. Specifically, the distribution originator can specify that (1) the distribution requires a confirmation-of-delivery message when the recipient node accepts delivery of the distribution information, (2) the information is personal, and (3) the information is for priority distribution.

RECIPIENT-ADDRESS

The RECIPIENT-ADDRESS operand specifies the element address token of the recipient; either RECIPIENT-ADDRESS (Format 1 or Format 42) applies.

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand, if present, specifies the 1- to 8-character group address token of the requester that initiated this request. If the DELIVER command does not include this operand, the default is the group address token of the command sender.

The command must include the ORIGINATING-NODE-ADDRESS operand when the recipient of the DELIVER command must route something back to the originator of the distribution request.

MESSAGE

The MESSAGE operand distributes a message to all recipients specified in the distribution request; either MESSAGE operand (Format 1 or Format 2) is valid. The DELIVER command must include the MESSAGE operand if only a message is delivered. If the DELIVER command does not include the MESSAGE operand, the DELIVER command must include the IDENTIFIED-DATA operand. The DELIVER command can also include both operands.

DISTRIBUTION-ID

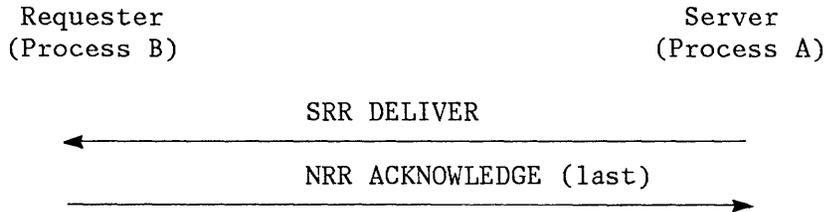
The DISTRIBUTION-ID (Format 1) operand uniquely identifies the distribution request; the unique identifier consists of the distribution document name and the date and time of the distribution request.

CORRELATION

The CORRELATION (Format 1) operand correlates a replying DELIVER command to a previously sent request, such as an OBTAIN command. The CORRELATION (Format 1) operand uniquely identifies the request to which this command replies and indicates whether this is the last reply or the not-last reply; that is, the CORRELATION operand contains a last or a not-last indicator. When the last replying command has been received, the request is complete.

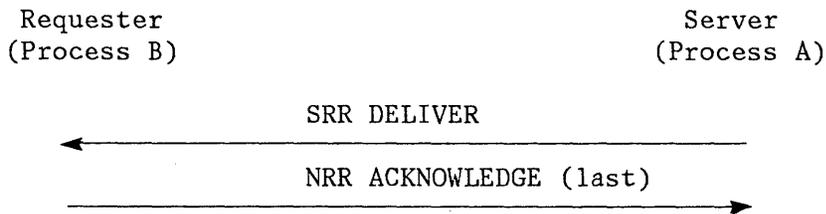
- Scenario 3 - Single Request

The following illustrates the use of the DELIVER command as a request to deliver information.



- Scenario 4 - Exception Conditions.

When exception conditions occur during the processing of the DELIVER command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The receiver rejects the DELIVER command if any of the following exception conditions exist.

- The receiver is not in a state in which it can receive and send data.
Exception = Catastrophic, Session, Intervention-Required, Unknown
Exception Code = X'C11217'
- The DELIVER command includes neither the IDENTIFIED-DATA operand nor the MESSAGE operands.
Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
- The CORRELATION operand is present, but it does not refer to a command previously sent by the receiver.
Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of CORRELATION operand
- The IDENTIFIED-DATA operand refers to non-existent data.
Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of IDENTIFIED-DATA operand
- The DELIVER command includes neither the IDENTIFIED-DATA operand nor the MESSAGE operand, and no message is in the ATTRIBUTE-LIST operand.
Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of ATTRIBUTE-LIST operand
- The Document Unit type is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit
Exception Code = X'C3020C'
Exception Data = LLIDF of Document Unit Introducer
- The Document Content Introducer type is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported,
Document-Content-Introducer
Exception Code = X'C30210'
Exception Data = LLIDF of Document Content Introducer

- The Document type in the Document Unit ID is not supported.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit ID
 Exception Code = X'C3020D'
 Exception Data = LLIDF and data of Document Unit ID
- The Document type parameter in the base subprofile is not supported.
 Exception = Catastrophic, Semantic, Data-Not-Supported,
 Document-Profile-Parameter
 Exception Code = X'C3020F'
 Exception Data = LLIDF and data of Document Type parameter
- The document profile is not supported.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Profile
 Exception Code = X'C3020E'
 Exception Data = LLIDF of Document Profile
- The MESSAGE operand format and the function sets supported are not compatible.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Command-Operand
 Exception Code = X'C30208'
 Exception Data = LLIDF of the MESSAGE operand
- The receiving processes' resources are unavailable.
 Exception = Catastrophic, Process, Resource-Not-Available, Document-Unit
 Exception Code = X'C4040C'
- The receiving process cancels the delivery of the data.
 Exception = Catastrophic, Process, Cancelled, Command
 Exception Code = X'C41407'

Support Considerations

The DELIVER command can include as many RECIPIENT-ADDRESS operands as are necessary to satisfy the requirement for multiple recipients of the same document or message.

An ATTRIBUTE-LIST operand is specified in the REQUEST-DISTRIBUTION command that initiates the distribution. The delivery characteristics specified in the ATTRIBUTE-LIST operand of the DELIVER command must be identical to those specified in the originating REQUEST-DISTRIBUTION command.

For coexistence with DIA Version 1.0 and migration to DIA Version 1.1, an office system node must be prepared to extract the message from the ATTRIBUTE-LIST operand and create a MESSAGE operand.

In a peer-to-peer connection, a DELIVER command should not include the ORIGINATING-NODE-ADDRESS, CORRELATION, DISTRIBUTION-ID and ATTRIBUTE-LIST operands. If it includes these operands, they are ignored. The MESSAGE operand is used to send a message in a peer-to-peer environment.

LIST

<u>Command</u>	<u>Operands</u>
LIST	LIST-ACTION [,RECIPIENT-ADDRESS] [,RECIPIENT-PASSWORD]

The LIST command requests delivery of distribution status information. The types of status information for this request are:

- Summary status indicators specify whether or not documents are enqueued for delivery to the recipient.
- Detailed status of documents enqueued for delivery to the recipient; this includes such items as the document distribution characteristics (for example, document type, personal, priority), status feedback (for example, confirmation of delivery and invalid recipient).

The LIST command can request status information for the requester, for a specific recipient, or for recipients on whose behalf the requester has the authority to act. See "Affinity" for a discussion of a requester acting on behalf of some other recipient. The requester can request that a summary status list, a detailed unformatted status list, or a detailed formatted status list be returned. All status returned is reported on a recipient-by-recipient basis.

The summary status list information contains indicators that specify whether or not:

- Priority documents are enqueued for delivery to the recipient.
- Non-priority documents are enqueued for delivery to the recipient.
- Personal documents are enqueued for delivery to the recipient.
- Feedback information (for example, a confirmation-of-delivery message) about a previous distribution request is enqueued for delivery.

Note: See Figure 40 on page 157 for a description of the returned summary status list that is returned.

The detailed status list information contains information about:

- A document enqueued for delivery: for example, the type of document distributed, the handling characteristics (COD requested, personal or priority) of the document distribution, or the originator of the distribution request.
- Delivered documents: for example, the date and time the document was delivered to the distribution recipient, whether or not the document was successfully delivered, or if not delivered, why it was not delivered (cancelled, invalid recipient).
- Outstanding requests for confirmation of delivery of a distribution: for example, which distribution recipients received the document, which distribution recipients have not taken delivery of the document at this time, or which distribution recipients cancelled the document.

The LIST command can request two types of detailed status list documents: unformatted and formatted. The unformatted status-list document consists of self-defining (LLIDF) parameters. See Figure 41 and Figure 42 for a description of an unformatted status document. The formatted status-list document contains the requested status information in a final-form-text document. The format of the final-form-text document is product-specific; DIA does not define this format.

Operand Descriptions

LIST-ACTION

The LIST-ACTION operand specifies the type of status information to return; either summary status information or detailed status information are valid. If the operand requests detailed status information, the requester must specify either unformatted or formatted status information. For detailed status information, the requester must select one of the following types of status:

- Documents enqueued for delivery only
- Confirmation of delivery status information only
- Distribution delivery errors only (failure to deliver a document)
- All of the above.

RECIPIENT-ADDRESS

The RECIPIENT-ADDRESS operand specifies the element address token of the recipient; either RECIPIENT-ADDRESS operand (Format 1 or Format 42) is valid. The LIST command must include the RECIPIENT-ADDRESS operand if the distribution status information to be delivered is for:

- The signed-on requester only
- A recipient other than the signed-on requester
- Recipients with affinity to the specified recipient and the specified recipient has affinity to the requester. See "Affinity" on page 17 for a discussion of a requester acting on behalf of some other recipient.

If the LIST command does not include the RECIPIENT-ADDRESS operand, the status information delivered is for the requester and all recipients that have affinity with the requester.

RECIPIENT-PASSWORD

The RECIPIENT-PASSWORD (Format 1) operand, if present, specifies a 1- to 8-character access authorization key for the specified recipient. The LIST command need not include the RECIPIENT-PASSWORD operand if the specified recipient (RECIPIENT-ADDRESS operand) is either the signed-on requester or a recipient that has affinity to the signed-on requester.

The command must include the RECIPIENT-PASSWORD operand if the information to be retrieved is for:

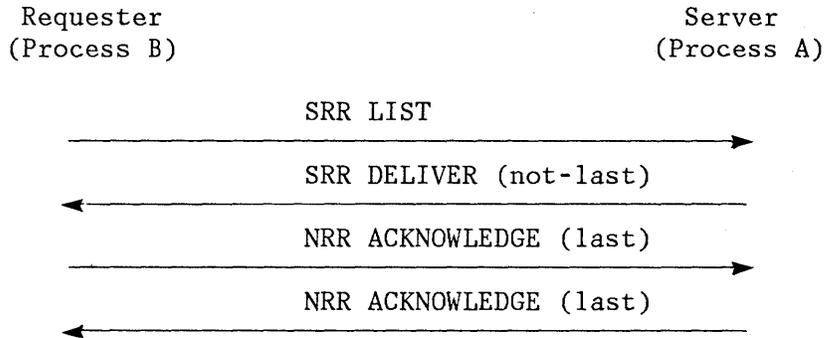
- A specific recipient other than the requester, the specified recipient does not have affinity with the requester, and the specified recipient has a password.
- Recipients with affinity to the specified recipient, and the specified recipient has affinity with the requester. See "Affinity" on page 17 for a discussion of a requester acting on behalf of some other recipient.

Request/Reply Protocol

The following scenarios illustrate possible replies to the LIST command.

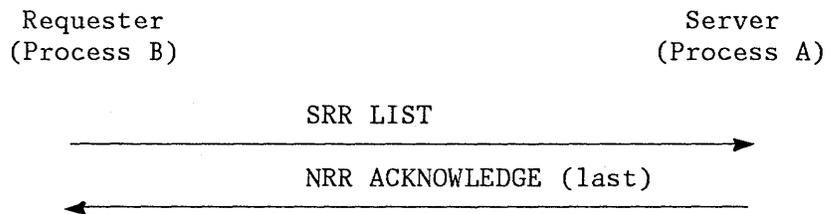
- Scenario 1 - Normal Completion with status information returned.

The normal reply to a LIST command is a DELIVER command that contains the requested status list information. After delivery of the status list document, the LIST command concludes by sending an ACKNOWLEDGE command.



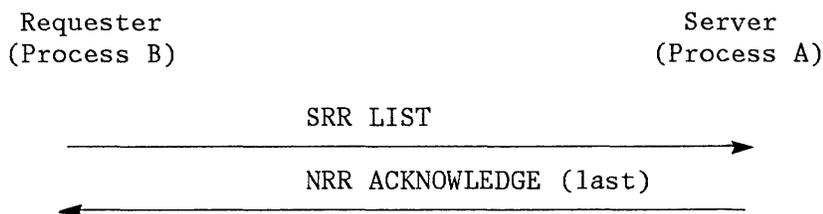
- Scenario 2 - Normal Completion with no status information returned.

When there is no status information to return, the LIST command concludes with an ACKNOWLEDGE command indicating normal completion.



- Scenario 3 - Exception Conditions.

When exception conditions occur during the processing of the LIST command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The server rejects the LIST command if any of the following conditions exist.

- The RECIPIENT-PASSWORD operand is present, but the RECIPIENT-ADDRESS operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
 Exception Code = X'C20708'
 Exception Data = LLIDF of RECIPIENT-ADDRESS operand

- The LIST command requests status for only the recipient specified in the RECIPIENT-ADDRESS operand. The RECIPIENT-ADDRESS specifies a recipient different from the signed-on recipient; the recipient does not have affinity with the signed-on recipient. The specified recipient has a password, but the RECIPIENT-PASSWORD is not specified.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
 Exception Code = X'C20708'
 Exception Data = LLIDF of RECIPIENT-PASSWORD operand

- The LIST command requests status for the recipient and recipients with affinity. The RECIPIENT-ADDRESS operand specifies a recipient different from the signed-on recipient. The specified recipient has a password, but the RECIPIENT-PASSWORD operand is not specified.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
 Exception Code = X'C20708'
 Exception Data = LLIDF of RECIPIENT-PASSWORD operand

- The RECIPIENT-ADDRESS operand contains an invalid name.
 Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
 Exception Code = X'C30309'
 Exception Data = LLIDF and data of RECIPIENT-ADDRESS operand
- The RECIPIENT-PASSWORD operand contains an invalid authorization key.
 Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
 Exception Code = X'C30509'
 Exception Data = LLIDF and data of RECIPIENT-PASSWORD operand
- The LIST-ACTION operand contains an undefined type of status information.
 Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 Exception Code = X'C30209'
 Exception Data = LLIDF and data of LIST-ACTION operand
- The server cannot return the status information in a formatted document of a document type allowable by the document type specified in the SIGN-ON command for the DIA session.
 Exception = Catastrophic, Process, Data-Not-Supported, Document-Content-Control
 Exception Code = X'C40211'
- The server cannot return formatted status in any allowable CGCSGID specified in the SIGN-ON command for the DIA session.
 Exception = Catastrophic, Process, Data-Not-Supported, Document-Content-Data
 Exception Code = X'C40212'
- The RECIPIENT-PASSWORD operand length exceeds the processing capacity of the receiver.
 Exception = Catastrophic, Process, Length-Invalid, Command-Operand
 Exception Code = X'C40F08'
 Exception Data = LLIDF and data of RECIPIENT-PASSWORD operand

Unformatted Summary Status

The following defines the format of the summary status list information document unit. The server sends the summary status-list document unit to the requester by using a replying DELIVER command.

The Summarized Status LLIDF reports all available summary status for the specified recipient.

<u>Field</u>		<u>Length</u>	<u>Value</u>
LLIDF	DOCUMENT UNIT	5	X'nnnnC90301'
	Document-Unit-ID	15	X'000A',CL13' '
LLIDF	DOCUMENT CONTENT INTRODUCER	5	X'nnnnCB0101'
LLIDF	Summarized Status	5	X'nnnnC90601'
	LLIDF Status Information	5	X'nnnnC33D01'
	STATUS	2	binary
	Reserved		B'00000000 00000000'
	Priority		B'00000000 0000xxx1'
	Non-Classified		B'00000000 0000xx1x'
	Personal		B'00000000 0000x1xx'
	Source-Status		B'00000000 00001xxx'
	Reserved		B'11111111 1111xxxx'
LLIDF	Recipient-Address	5	X'nnnnC30601'/X'nnnnC30642'
		v	depends on format

Figure 40. Unformatted Summary Status Document Unit

In the following field descriptions, the term distribution means a document, a document with an appended message, or a message only.

Field Descriptions

The Priority field indicates that priority distributions are enqueued for delivery to the specified recipient.

The Non-Classified field indicates that distributions that are not priority or personal are enqueued for delivery for the specified recipient.

The Personal field indicates that personal distributions are enqueued for delivery to the specified recipient.

The Source-Status field indicates that status information is available for distribution requests previously submitted.

The Recipient-Address field specifies the address of the recipient for which status is reported.

Unformatted Recipient Status

The Unformatted Recipient Status document unit returns detailed status information to a recipient node. The information returned includes status for document or messages enqueued for delivery to the requester.

<u>Field</u>	<u>Length</u>	<u>Value</u>
LLIDF DOCUMENT UNIT	5	X'nnnnC90301'
Document-Unit-ID	15	X'000A',CL13
LLIDF DOCUMENT CONTENT INTRODUCER	5	X'nnnnCB0101'
LLIDF Unformatted Recipient Status	5	X'nnnnC90501'
LLIDF Attribute-List	5	X'nnnnC30501'
COD	1	binary
No		X'00'
Yes		X'01'
Personal	1	binary
No		X'00'
Yes		X'01'
Priority	1	binary
No		X'00'
Yes - Highest Level		X'01'
'Retired'	1	binary
Reserved		X'01'
LLIDF Distribution ID	5	X'nnnnC34001'
Distribution-Document-Name	20	Characters
Date-Time Requested	6	Binary
Distribution-Name	1-44	Characters
LLIDF	5	X'nnnnC70601'
Document-Type	2	binary
LLIDF	5	X'nnnnC70C01'
System Code	13	Characters
LLIDF	5	X'nnnnC31101'
Originating-Node Address	1-8	Characters
LLIDF	5	X'nnnnC32301'/X'nnnnC32342'
Source-Address	v	depends on format
LLIDF	5	X'nnnnC30601'/X'nnnnC30642'
Recipient-Address	v	depends on format

Figure 41. Unformatted Recipient Status Document Unit

The status in the section from A through B (Figure 41) relates to a distribution enqueued for delivery to one or more recipients. The section from A through B is delimited by an LLIDF and contains status for only one distribution. Section A through B can be repeated to relate status for more than one distribution. The information in this section comes from operands that arrived at the OSN via the REQUEST-DISTRIBUTION command. The document unit must send the Distribution-ID field with at least the Distribution-Document-Name, the Attribute-List field, and the Recipient-Address field when it presents status in section A through B. In general, the fields can appear in any order. The Recipient-Address field can repeat several times if the preceding information still applies.

FIELD DESCRIPTIONS

A-----B

The allowable COD field values and their meanings are:

X'00' - The distribution identified in the Distribution-ID field is not COD for the specified recipients.

X'01' - The distribution identified in the Distribution-ID field is COD for the specified recipients.

X'02' - X'FF' - Reserved.

The allowable Personal field values and their meanings are:

X'00' - The distribution identified in the Distribution-ID field is not personal for the specified recipients.

X'01' - The distribution identified in the Distribution-ID field is personal for the specified recipients.

X'02' - X'FF' - Reserved

The allowable Priority field values and their meanings are:

X'00' - For the specified recipients, the LIST command does not request priority handling for the distribution identified in the Distribution-ID field.

X'01' - For the specified recipients, the LIST command requests priority handling for the distribution identified in the Distribution-ID field.

X'02' - X'FF' - Reserved for future levels of priority.

The retired field is a reserved field with a value of X'01'.

The Distribution-ID field specifies the distribution document name assigned by the office system node, the date and time of the request for the distribution, and the distribution name assigned by the source. This field is described in the DISTRIBUTION-ID operand discussion.

The Document-Type field specifies the data stream classification for the document unit. The document type encoding is listed in Appendix C, "DIA Document Types."

The System-Code field specifies the product that is the creator of the document unit.

The Originating-Node-Address field identifies the office system node that processed the REQUEST-DISTRIBUTION command.

The Source-Address field specifies the source that identifies the user or process that originated the REQUEST-DISTRIBUTION command.

The Recipient-Address field specifies a recipient that identifies the user or process at the destination-node to which the distribution was directed.

Unformatted Source Status

The Unformatted Source Status document unit returns detailed status information to a source node. The information returned includes status from previously-distributed documents or messages.

<u>Field</u>	<u>Length</u>	<u>Value</u>
LLIDF DOCUMENT UNIT	5	X'nnnnC90301'
Document-Unit-ID	15	X'000A',CL13'
LLIDF DOCUMENT CONTENT INTRODUCER	5	X'nnnnCB0101'
LLIDF Unformatted Source Status	5	X'nnnnC90701'
LLIDF	5	X'nnnnC32301'/X'nnnnC32342' C
Source-Address	v	depends on format
LLIDF Distribution Correlation	5	X'nnnnC31501'
Date-Time Delivered	6	binary
Date-Time Sent	6	binary
Distribution-Document-Name	20	Characters
Distribution-Name	1-44	Characters
LLIDF Document Status	5	X'nnnnC31601'
Notification-code	2	binary
Reserved		X'0000'
Confirmation-of-Delivery		
Delivered		X'0101'
System Error		X'0201'
Invalid Recipient		X'0301'
Invalid Document		X'0401'
Document Cancelled-All Recipients		X'0501'
Data Purged		X'0502'
Maximum List Exceeded		X'0601'
Distribution-List-Status		
Some Cancelled or Purged, Some Delivered		X'0701'
Some Delivered,Some Invalid		X'0702'
Some Cancelled or Purged, Some Invalid		X'0703'
Some Cancelled or Purged, Some Delivered,Some Invalid		X'0704'
Some Cancelled,Some Purged		X'0705'
Exception-Condition		X'FFFF'

Figure 42 (Part 1 of 2). Unformatted Source Status Document Unit

Detecting-Node-ID	1-8	Characters
LLIDF SNADS Status Data	5	X'nnnnC35741'
LT Condition Code	4	X'nn01xxxx'
LT Hop Count	6	X'nn02xxxxxxxx'
LLIDF Reporting DSU	5	X'nnnnC34041'
LT RGN	3-10	X'nn01xxxxxxxx'
LT REN	3-10	X'nn02xxxxxxxx'
LLIDF Receiving DSU	5	X'nnnnC36141'
LT RGN	3-10	X'nn01xxxxxxxx'
LT REN	3-10	X'nn02xxxxxxxx'
LLIDF Destination-Node-Address	5	X'nnnnC32F01'
DNA	1-8	Characters
LLIDF Exception Code	5	X'nnnnC32201'
Exception-Condition	v	Characters/binary
LLIDF	5	X'nnnnC30601'/X'nnnnC30642'
Recipient-Address	v	depends on format

Figure 42 (Part 2 of 2). Unformatted Source Status Document Unit

The status in the section from C through D (Figure 42) relates to a distribution that was previously sent. The section from C through D is delimited by an LLIDF and contains status for only one distribution. Section C through D can be repeated in order to report status for more than one distribution. The document unit must send the Document-Correlation field with at least the Distribution-Document-Name field, the Recipient-Address field, and the Document-Status field when it reports status in sections C through D.

Field Descriptions

C-----D

The Source-Address field specifies the source that identifies the user or process that originated the REQUEST-DISTRIBUTION command.

The Date-Time Delivered field contains the date and time when the delivery of the distribution to the recipient took place or when a routing or other error was detected. The syntax of the Date-Time Delivered field is:

Date-Time Delivered = YYMDHM
 where: YY = 2 bytes specifying the year AD (0-65535)
 M = 1 byte specifying the month (1-12)
 D = 1 byte specifying the day of the month (1-31)
 H = 1 byte specifying the hour of the day (0-23)
 M = 1 byte specifying the minute of the hour (0-59)

The Date-Time Sent field contains the date and time when the distribution was sent from the source node to the originating node. The syntax of the Date-Time Sent field is the same as the syntax of the Date-Time Delivered field.

The Distribution-Document-Name is the name that the originating node assigned to the distribution. If a missing distribution is being reported, the field is left blank.

The Distribution-Name field contains the distribution name that the source assigns.

The Notification-Code field specifies whether or not the document was delivered successfully.

Confirmation of Delivery

X'0101' (Delivered)-indicates if the document is delivered to the recipient or to all recipients contained in the list specified by the Recipient-Address field.

X'0201' (System Error)-indicates that the destination node encountered a permanent error resulting in a failure to deliver the distribution to the specified recipients.

X'0301' (Invalid-Recipient)-indicates that the server cannot locate the recipient, or all of the recipients contained in a list specified by the Recipient-Address.

X'0401' (Invalid Document)-indicates that the destination node is unable to store the document, for example, an invalid document class is specified.

X'0501' (All)-indicates that the document has been cancelled by all of the recipients at a destination node.

X'0502' (Data-Purged)-indicates that the data has been purged at the destination OSN.

X'0601' (Maximum-List-Exceeded)-indicates that the specified recipient address is the name of a list that develops into a combined recipient list that exceeds the processing capability of the destination node.

Distribution List Status

X'0701' (Some Delivered, Some Cancelled) - indicates that the message or document has been delivered to some of the recipients specified by the recipient address and cancelled or purged for all of the other recipients.

X'0702' (Some Delivered, Some Invalid) - indicates that the message or document has been delivered to some of the recipients specified by the recipient address and all of the others are invalid.

X'0703' (Some Cancelled, Some Invalid) - indicates that the message or data has been cancelled or purged for some of the recipients specified by the recipient address and all of the others are invalid.

X'0704' (Some Cancelled, Some Delivered, Some Invalid) - indicates that the message or document has been delivered to some, cancelled or purged for some, and all of the others are invalid.

X'0705' (Some Cancelled, Some Purged) - indicates that the message or data has been cancelled by some recipients and purged for all of the other recipients.

X'FFFF' (Exception-Code) - indicates that exception conditions are being reported, and the EXCEPTION-CODE operand must be present.

The Detecting-Node-ID field specifies the address of the office system node that detected the error or is sending the COD notification.

The Destination-Node-Address field identifies the destination node for the distribution. This field contains the Destination-Node-ID of 1 to 8 bytes.

The Exception-Code field specifies the exception conditions, if any, that occurred at the indicated destination for the identified recipients.

The Recipient-Address field specifies a recipient that identifies the user or process at the destination node for the distribution.

The SNADS Status Data field specifies that an error occurred within the SNADS network. The SNADS condition code specified the type of error encountered.

The Reporting DSU specifies the name of the SNADS distribution service unit (DSU) that was sending the SNADS data when the specified error was detected.

The receiving DSU specifies the name of the SNADS distribution service unit (DSU) that was receiving the SNADS data when the specified error was detected.

In general, the fields can appear in any order. The Recipient-Address field can repeat if the preceding information is still applicable.

The unformatted recipient status (section A through B) can occur without the unformatted source status (section C through D). Likewise, the unformatted source status (section C through D) can occur without the unformatted recipient status (section A through B). Also, both the unformatted recipient status (section A through B), and the unformatted source status (section C through D) can occur together in the same document unit.

The unformatted status document unit need not include all the fields in the unformatted recipient status (section A through B) or the unformatted source status (section C through D). The document unit must omit any fields that correspond to null or omitted operands.

OBTAIN

<u>Command</u>	<u>Operands</u>
OBTAIN	OBTAIN-OPTION [,DOCUMENT-PASSWORD] [,RECIPIENT-ADDRESS] [,RECIPIENT-PASSWORD]

The OBTAIN command requests the delivery of distribution information enqueued for delivery either to the requester or to a specific recipient or recipients on whose behalf the requester has the authority to act. See "Affinity" on page 17 for a discussion of a requester acting on behalf of some other recipient.

The types of information distributed include:

- Personal and nonpersonal documents
- Personal and nonpersonal documents with appended messages
- Personal and nonpersonal messages only
- Status information about previous distribution requests.

The requester can selectively retrieve enqueued information, using one of the following options. The term distribution means a personal or nonpersonal document, a document with an appended message, or a message only.

- All distributions (priority and nonpriority)
- All priority distributions only
- A specific distribution
- All messages only (priority and nonpriority)
- All priority messages only.

To obtain personal documents or messages, the requester must supply the appropriate recipient's document password if one is defined for that requester. For a requester that does not have a defined document password, personal distributions are delivered only when the obtain-requester is the signed-on user or the OBTAIN request includes both a RECIPIENT-ADDRESS operand and a RECIPIENT-PASSWORD operand. The RECIPIENT-PASSWORD operand is not required for those users without a defined sign-on password.

Two additional OBTAIN command options are available, these are:

- Status information about previous distribution requests and all priority, nonpriority, personal, and nonpersonal messages only
- Status information about previous distribution requests and all priority, nonpriority, personal, and nonpersonal distributions.

A recipient need not have a document password to retrieve personal distributions with these options. The recipient node is responsible for ensuring that the authority to access personal documents is validated prior to delivery of those documents to the end user.

The reply to an OBTAIN command is zero or more DELIVER commands containing the information requested. The OBTAIN command concludes with a replying ACKNOWLEDGE (last) command. All replying commands correlate to the OBTAIN command.

Operand Descriptions

OBTAIN-OPTION

The OBTAIN-OPTION (Format 1) operand specifies the requested delivery option and its associated parameters. The delivery options are:

- Obtain all distributions (priority and non-priority)
- Obtain a specific distribution
- Obtain all priority distributions only
- Obtain priority messages only
- Obtain all messages only (priority and non-priority).

DOCUMENT-PASSWORD

The DOCUMENT-PASSWORD (Format 1) operand specifies a 1- to 8-byte personal document authorization key associated with the recipient. The OBTAIN command must include this operand if the requester wants to obtain personal distributions and that requester has a document password. If the requester does not have a personal document password, the operand can be omitted. In this case, the server delivers personal distributions only if the requester of the OBTAIN command is the signed-on user, or the OBTAIN command includes both a RECIPIENT-ADDRESS operand and RECIPIENT-PASSWORD operand. The RECIPIENT-PASSWORD operand is not required for a user without a defined sign-on password.

If the OBTAIN command does not include the RECIPIENT-ADDRESS operand, the DOCUMENT-PASSWORD operand must be the signed-on requester's personal document password. When the command includes RECIPIENT-ADDRESS operand, the DOCUMENT-PASSWORD operand must be the specified recipient's personal document password.

RECIPIENT-ADDRESS

The RECIPIENT-ADDRESS operand specifies the element address token of the recipient; either RECIPIENT-ADDRESS (Format 1 or Format 42) is valid. The OBTAIN command must include the RECIPIENT-ADDRESS operand if the distribution information to be obtained is for:

- A recipient other than the signed-on requester.
- Recipients have affinity to the specified recipient, and the specified recipient has affinity to the requester. See "Affinity" on page 17 for a discussion of a requester acting on behalf of some other recipient.
- The signed-on requester only.

If the command does not include the RECIPIENT-ADDRESS operand, the information is obtained for the signed-on requester and all recipients that have affinity with the requester.

RECIPIENT-PASSWORD

The RECIPIENT-PASSWORD (Format 1) operand specifies a 1- to 8-character access authorization key for the specified recipient. The OBTAIN command need not include the RECIPIENT-PASSWORD operand if the specified recipient (RECIPIENT-ADDRESS operand) is either the signed-on requester or a recipient that has affinity to the signed-on requester.

The command must include the RECIPIENT-PASSWORD operand if the information to be obtained is for either:

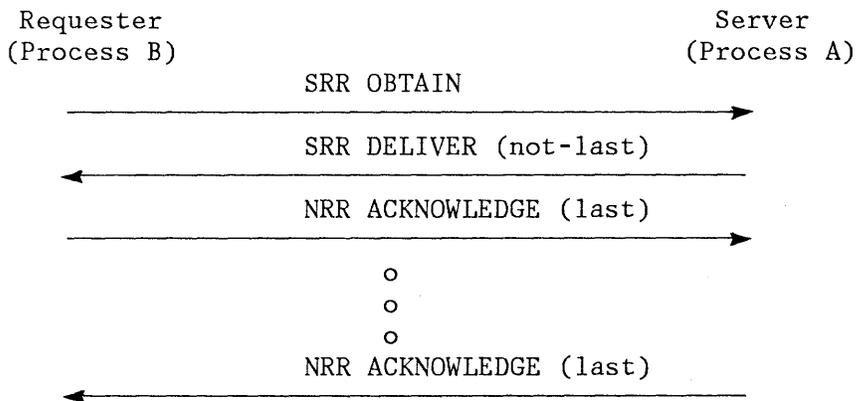
- A specific recipient other than the signed-on requester, the specified recipient does not have affinity with the signed-on requester, and the specified recipient has a password.
- Recipients with affinity to the recipient specified in the RECIPIENT-ADDRESS operand, and the specified recipient has affinity to the requester. See "Affinity" on page 17 for a discussion of a requester acting on behalf of some other recipient.

Request/Reply Protocol

The following scenarios illustrate possible replies to the OBTAIN command:

- Scenario 1 - Normal Completion with distributions returned

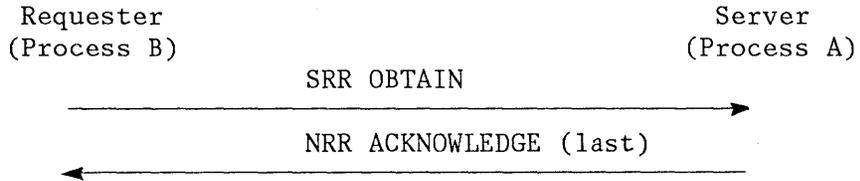
The normal reply sequence to an OBTAIN command is zero or more DELIVER commands followed by an ACKNOWLEDGE command that concludes the processing. The number of replying DELIVER commands is determined by the number of distributions obtained.



The specific OBTAIN operand values and the DOCUMENT-TYPE and GCID operands specified on the SIGN-ON command at DIA session establishment determine the types of data returned for the OBTAIN command.

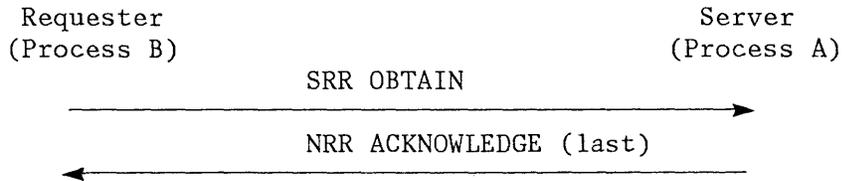
- Scenario 2 - Normal Completion with no distributions returned

When there are no distributions enqueued for delivery, the OBTAIN command concludes with an ACKNOWLEDGE command indicating normal completion.



- Scenario 3 - Exception Conditions

When exception conditions occur during the processing of the OBTAIN command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The server rejects the OBTAIN command if any of the following specific OBTAIN exception conditions are detected:

- The RECIPIENT-PASSWORD operand is present and the RECIPIENT-ADDRESS operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of RECIPIENT-ADDRESS operand

- The OBTAIN command requests distributions for only the recipient specified in the RECIPIENT-ADDRESS operand, the RECIPIENT-ADDRESS operand specifies a recipient different from the requester, the specified recipient does not have affinity with the requester, the specified recipient has a password, and the command does not include the RECIPIENT-PASSWORD.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of RECIPIENT-PASSWORD operand

- The OBTAIN command does not include the Distribution-Document-Name, and the OBTAIN-OPTION operand specifies 'Specific Distribution.'

Exception = Catastrophic, Syntax, Data-Not-Found, Operand-Value
Exception Code = X'C20709'
Exception Data = LLIDF and data of OBTAIN-OPTION operand

- The OBTAIN command requests delivery of distributions for the recipient(s) with affinity. The RECIPIENT-ADDRESS operand specifies a recipient different from the requester and the specified recipient has a password. However, the OBTAIN command does not include the RECIPIENT-PASSWORD operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of RECIPIENT-PASSWORD operand

- The RECIPIENT-ADDRESS operand contains an invalid name.

Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
Exception Code = X'C30309'
Exception Data = LLIDF and data of RECIPIENT-ADDRESS operand

- The RECIPIENT-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception Data = LLIDF and data of RECIPIENT-PASSWORD operand

- The OBTAIN command requests a personal distribution but does not include the DOCUMENT-PASSWORD operand.
 - Exception = Catastrophic, Semantic, Unauthorized-Access, Command-Operand
 - Exception Code = X'C30308'
 - Exception Data = LLIDF of DOCUMENT-PASSWORD operand
- The DOCUMENT-PASSWORD operand contains an invalid authorization key.
 - Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
 - Exception Code = X'C30509'
 - Exception Data = LLIDF and data of DOCUMENT-PASSWORD operand
- The OBTAIN-OPTION operand specifies a non-existent Distribution-Document-Name.
 - Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
 - Exception Code = X'C30709'
 - Exception Data = LLIDF and data of OBTAIN-OPTION operand
- The OBTAIN command does not include the RECIPIENT-ADDRESS operand, although the recipient neither is the signed-on user nor has affinity with the signed-on user.
 - Exception = Catastrophic, Semantic, Unauthorized-Access, Command-Operand
 - Exception Code = X'C30308'
 - Exception Data = LLIDF of RECIPIENT-ADDRESS operand
- The OBTAIN-OPTION operand contains an undefined option value.
 - Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 - Exception Code = X'C30209'
 - Exception Data = LLIDF and data of OBTAIN-OPTION operand
- The OBTAIN command requests a specific document that does not match an allowable document type specified in the SIGN-ON command.
 - Exception = Severe, Process, Data-Not-Supported, Document-Content-Control
 - Exception Code = X'840211'
 - Exception Data = none
- The OBTAIN command requests a specific document that cannot be validated or translated to meet CGCSGID usage specified by the SIGN-ON command.
 - Exception = Severe, Process, Data-Not-Supported, Document-Content-Data
 - Exception Code = X'840212'
 - Exception Data = none

- The OBTAIN command, by specifying all distributions in the OBTAIN-OPTIONS operand, requests one or more documents that do not match an allowable document type specified in the SIGN-ON command.

Exception = Warning, Process, Data-Not-Supported, Document-Content-Control
Exception Code = X'440211'
Exception Data = none

- The OBTAIN command, by specifying all distributions in the OBTAIN-OPTIONS operand, requests one or more documents that cannot be validated or translated to meet the CGCSGID usage specified by the SIGN-ON command.

Exception = Warning, Process, Data-Not-Supported, Document-Content-Data
Exception Code = X'440212'
Exception Data = none

- The RECIPIENT-PASSWORD operand length exceeds the processing capacity of the receiver.

Exception = Catastrophic, Process, Length-Invalid, Command-Operand
Exception Code = X'C40F08'
Exception Data = LLIDF and data of RECIPIENT-PASSWORD operand.

- The DOCUMENT-PASSWORD operand length exceeds the processing capacity of the receiver.

Exception = Catastrophic, Process, Length-Invalid, Command-Operand
Exception Code = X'C40F08'
Exception Data = LLIDF and data of DOCUMENT-PASSWORD operand

- The server cannot fetch the requested document.

Exception = Catastrophic, Process, Resource-Not-Available, Document-Unit
Exception Code = X'C4040C'
Exception Data = None

PROCESS-BIT-STRING

<u>Command</u>	<u>Operands</u>
PROCESS-BIT-STRING	IDENTIFIED-DATA, SCAN-DATA, BIT-STRING-REPRESENTATION

The PROCESS-BIT-STRING (PBS) command lets a facsimile device source node invoke and pass information to a device-dependent program in the office system node. The device-dependent program interprets and processes the information as a single DIA function request, such as a request to distribute a document to one or more recipients.

The DIA function request is contained in facsimile control sheets. One or more facsimile control sheets describe the DIA function to be performed.

The facsimile control sheet appears in two forms: Scan-Data and Bit-String-Representation (BSR). The PROCESS-BIT-STRING command must include the Bit-String-Representation form of the control sheet that contains the DIA function request information in binary coded format. The Scan-Data form of the control sheet is optional and contains the facsimile page image of the control sheet. The format of the control sheet is product-dependent.

The PROCESS-BIT-STRING command transports control sheet information in DIU data units: a Scan-Data data unit and a BSR data unit. If both the Scan-Data and the BSR forms of a control sheet are present, the DIU includes them as two paired data units. If there are multiple control sheets associated with the request, multiple paired data units are present.

The requester using the DELIVER and/or ACKNOWLEDGE commands return output from the invoked DIA function request to the requester.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA (Format 1) operand specifies the location of the document to be processed by this request. When the DIU includes a document, the IDENTIFIED-DATA operand value is X'01'. When the DIU does not include a document, the IDENTIFIED-DATA operand must have a value of X'00' to indicate that there is no document unit in this DIU.

SCAN-DATA

The SCAN-DATA (Format 1) operand refers to a data unit within the same DIU. The Scan-Data data unit need not be present. Because the PROCESS-BIT-STRING command must include the SCAN-DATA operand, the reference byte of the operand must be zero (X'00') whenever the Scan-Data data unit is not present. When the Scan-Data data unit is present, this operand always refers to the first data unit of the contiguous group of data units representing the control sheets.

BIT-STRING-REPRESENTATION

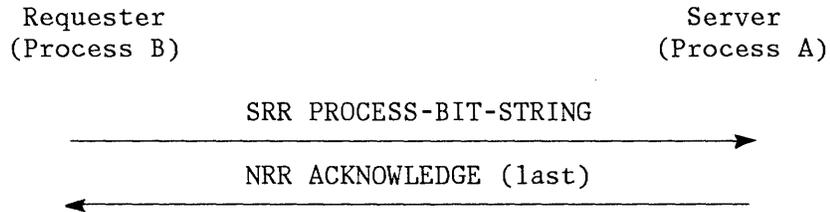
The BIT-STRING-REPRESENTATION (Format 1) operand refers to a data unit within the DIU. The BSR data unit must be present. The BIT-STRING-REPRESENTATION operand always refers to the first BSR data unit of the contiguous group of data units that represent the control sheets. The BSR data unit must follow the Scan-Data data unit.

Request/Reply Protocol

The following scenarios illustrate possible replies to the PROCESS-BIT-STRING command:

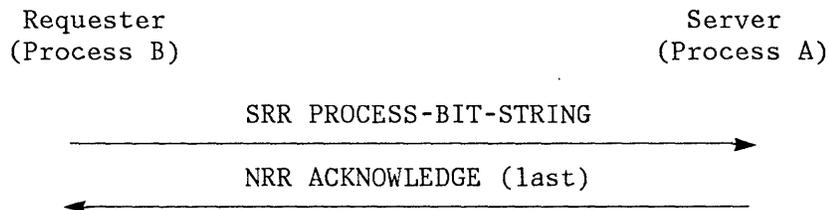
- Scenario 1 - Normal Conditions

The reply to the PROCESS-BIT-STRING command depends on the DIA function performed. The server replies with the DELIVER and/or ACKNOWLEDGE commands.



- Scenario 2 - Exception Condition.

When exception conditions occur during the processing of the PROCESS-BIT-STRING command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions."

In addition to the general exception conditions, the following exception conditions are specific to the PROCESS-BIT-STRING command.

- At least one data unit does not follow the command.

Exception = Catastrophic, Syntax, Data-Not-Found, Data-Unit
Exception Code = X'C2070A'
Exception data = LLIDF of Scan-Data data unit

- The BIT-STRING-REPRESENTATION operand contains other than a DIA command.

Exception = Catastrophic, Semantic, Data-Not-Supported, Data-Unit
Exception Code = X'C3020A'
Exception data = LLIDF of BSR data unit

Support Considerations

DIA does not define the name of the device-dependent program invoked by the PROCESS-BIT-STRING command process. The device-dependent program acts as an agent (surrogate end-user) source node and uses DIA commands to modify DIA entities, such as a DIA distribution library. Replies from the DIA command processes return to the device-dependent program. The device-dependent program, in turn, sends the DIA replying commands to the requester of the PROCESS-BIT-STRING command.

REQUEST-DISTRIBUTION

<u>Command</u>	<u>Operands</u>
REQUEST-DISTRIBUTION	ATTRIBUTE-LIST [,IDENTIFIED-DATA] [,DESTINATION-NODE-ADDRESS] ,RECIPIENT-ADDRESS [,SOURCE-ADDRESS] [,SOURCE-PASSWORD] [,DISTRIBUTION-NAME] [,MESSAGE]

The REQUEST-DISTRIBUTION command is used to distribute information to one or more recipients in the office system network.

The types of information include a document with or without an appended message, and a message only. The term document means any collection of data, such as a revisable-text document, a user's data processing payroll master file, or a DIA unformatted status document.

The document to be distributed can be submitted with the request in the DIU, or it can be distributed from the OSN document library. Messages can be distributed only in the DIU of the request.

A source node sends distribution requests to an originating office system node. The functions performed by the office system node include:

- Validation of the request.
- Safe-store of the request and information to be distributed.
- Assignment to the distribution request of a token unique within a network. The assigned token is the distribution document name.
- Schedule an asynchronous process to fan-out the information to the specified recipients.

Output from the office system node includes returning a unique distribution document name to the requester. This unique distribution document name is used to correlate subsequent status and error messages with the original distribution request.

The LIST and OBTAIN commands can obtain the status and error messages generated after the office system node accepts the REQUEST-DISTRIBUTION request.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand, if present, identifies the document to be distributed. Formats 1, 2, 3, 41, and 42 are valid in support of the REQUEST-DISTRIBUTION command. If the REQUEST-DISTRIBUTION command does not include an IDENTIFIED-DATA operand, the server is not requested to distribute a document. Consequently, a MESSAGE operand becomes required rather than optional.

The IDENTIFIED-DATA (Format 1) operand specifies that the document to be distributed is located in the DIU document unit.

The IDENTIFIED-DATA (Format 2) operand identifies the document to be distributed by means of a 1- to 44-character document name.

The IDENTIFIED-DATA (Format 3) operand identifies the document to be distributed by means of the position of the document reference in a specified Search Result List. Only the options specifying the document content and associated profile are valid for the REQUEST-DISTRIBUTION command.

The IDENTIFIED-DATA (Format 41) operand identifies the document to be distributed by means of a 1- to 44-character document name whose CGCSGID can be specified explicitly in the operand.

The IDENTIFIED-DATA (Format 42) operand identifies the document to be deleted by means of its Library-Assigned Document Name (LADN).

DESTINATION-NODE-ADDRESS

The DESTINATION-NODE-ADDRESS operand specifies the 1- to 8-character group address token of the distribution recipient. If the REQUEST-DISTRIBUTION command does not include this operand, the group address token of the command sender is used as the default value.

At least one RECIPIENT-ADDRESS operand must follow the DESTINATION-NODE-ADDRESS operand. The DESTINATION-NODE-ADDRESS operand can appear more than once in the command.

ATTRIBUTE-LIST

The requester uses ATTRIBUTE-LIST (Format 1) to specify the distribution processing characteristics. Specifically, the requester can specify: (1) a confirmation-of-delivery is to be returned when the recipient node accepts delivery of the distribution information, (2) the information is personal, requiring the recipient to specify a personal document password to receive the information, and (3) the information is for priority distribution.

RECIPIENT-ADDRESS

The RECIPIENT-ADDRESS operand specifies the element address token of the recipient; either RECIPIENT-ADDRESS operand (Format 1 or Format 42) is valid.

The operand can be repeated. The group token that applies to this element address token is the preceding DESTINATION-NODE-ADDRESS operand value. If the REQUEST-DISTRIBUTION command does not include the DESTINATION-NODE-ADDRESS operand (the recipient's group address token), the group address token of the command sender is used as the default value.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS operand (Format 1 or Format 42) is valid. If the REQUEST-DISTRIBUTION command does not include this operand, the element address token of the command sender is used as the default value.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated this request. The SOURCE-PASSWORD (Format 1) operand is required if the SOURCE-ADDRESS (Format 1) is specified. The REQUEST-DISTRIBUTION command must include the SOURCE-PASSWORD (Format 1) operand if all the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different than the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

Otherwise, the command should omit the SOURCE-PASSWORD.

DISTRIBUTION-NAME

The DISTRIBUTION-NAME operand, if present, specifies the 1- to 44-character name of the distribution as specified by the requester. The requester can use the value of the DISTRIBUTION-NAME operand as a user-correlation value for the distribution request. The server returns this DISTRIBUTION-NAME operand value in all asynchronous status messages.

MESSAGE

The MESSAGE operand is used to distribute a 1- to 256-character message to all recipients specified in the distribution request; either MESSAGE operand (Format 1 or Format 2) is valid.

Operand-Syntax

The following syntax applies to the operands of the REQUEST-DISTRIBUTION command:

- The ATTRIBUTE-LIST operand and at least one RECIPIENT-ADDRESS operand must appear in the command.
- For coexistence with DIA Version 1.0 and migration to DIA Version 1.1, an office system node must be prepared to extract the message from the ATTRIBUTE-LIST operand and create a MESSAGE operand.
- Once the DESTINATION-NODE-ADDRESS operand appears, the group address token value specified by it applies to all succeeding RECIPIENT-ADDRESS operands until the next occurrence of the DESTINATION-NODE-ADDRESS operand or the end of the command. At least one RECIPIENT-ADDRESS operand must appear between each DESTINATION-NODE-ADDRESS operand and following the last DESTINATION-NODE-ADDRESS operand.

Request/Reply Protocol

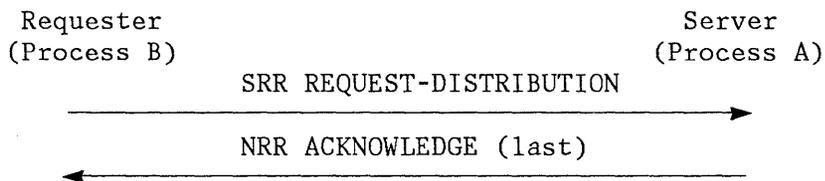
The following scenarios illustrate possible replies to the REQUEST-DISTRIBUTION command:

- Scenario 1 - Normal Completion

The reply to a REQUEST-DISTRIBUTION command is an ACKNOWLEDGE command that the server sends to the requester after the distribution request has been validated and safe-stored for further processing. The REPLY-DATA operand on the ACKNOWLEDGE command contains the distribution document name assigned to the distribution request.

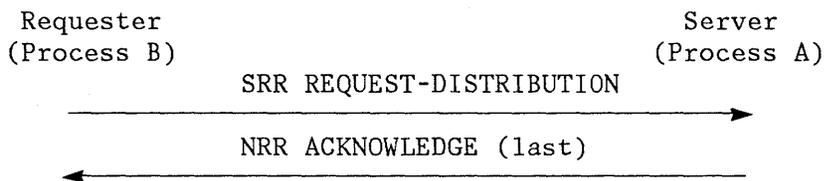
Validation of all recipient address tokens has not necessarily been performed.

After the ACKNOWLEDGE reply is sent to the REQUEST-DISTRIBUTION command, any exception conditions detected during the performance of the distribution are reported in a status document delivered in reply to a LIST or OBTAIN command.



- Scenario 2 - Exception Conditions

An ACKNOWLEDGE command that contains an exception condition in the EXCEPTION-CODE operand is the reply to any exception conditions detected during the processing of the REQUEST-DISTRIBUTION command.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The server rejects the REQUEST-DISTRIBUTION command if any of the following conditions exist.

- The SOURCE-PASSWORD operand is present, but the SOURCE-ADDRESS operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of SOURCE-ADDRESS operand

- The SOURCE-ADDRESS operand specifies a requester different from the command sender, the specified requester has a password, and the REQUEST-DISTRIBUTION command does not include the SOURCE-PASSWORD operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of SOURCE-PASSWORD operand

- An ATTRIBUTE-LIST operand does not precede the RECIPIENT-ADDRESS operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of ATTRIBUTE-LIST operand

- A RECIPIENT-ADDRESS operand does not follow the ATTRIBUTE-LIST operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of RECIPIENT-ADDRESS operand

- A RECIPIENT-ADDRESS operand does not follow the DESTINATION-NODE-ADDRESS operand.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception Data = LLIDF of RECIPIENT-ADDRESS operand

- The IDENTIFIED-DATA operand is present and does not refer to either: (1) a DIU document unit, (2) a valid document library name to which the requester has access, or (3) a valid search-result-list ID and entry number reference.

Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of IDENTIFIED-DATA operand

- The REQUEST-DISTRIBUTION command includes neither the IDENTIFIED-DATA nor the MESSAGE operands.

Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of ATTRIBUTE-LIST operand

- The ATTRIBUTE-LIST operand contains invalid attribute values.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception Data = LLIDF and data of ATTRIBUTE-LIST operand

- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
Exception Code = X'C30309'
Exception Data = LLIDF and data of SOURCE-ADDRESS operand

- All of the specified recipient addresses are invalid.

Exception = Catastrophic, Semantic, Execution-Terminated, Operand-Value
Exception Code = X'C30609'
Exception Data = LLIDF and data of last RECIPIENT-ADDRESS operand

- The SOURCE-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception Data = LLIDF and data of SOURCE-PASSWORD operand

- The RECIPIENT-ADDRESS operand contains an invalid address.

Exception = Severe, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'830209'
Exception Data = LLIDF and data of RECIPIENT-ADDRESS operand

Support Considerations

After a user has searched a DIA document library, the IDENTIFIED-DATA (Format 3) operand is used to refer to a document listed in a specified Search Result List. This format of the IDENTIFIED-DATA operand permits the requester to refer indirectly to the document to be distributed by means of its relative position in the Search Result List.

When a requester issues a REQUEST-DISTRIBUTION command, the status can be returned to the signed-on user or to someone else in the following ways:

- If the command does not include the SOURCE-ADDRESS operand, the status is returned to the signed-on user.
- If the command includes the SOURCE-ADDRESS operand, the status is returned to the user specified by the SOURCE-ADDRESS operand.

The MESSAGE operand can occur only once. The message is sent to all recipients.

STATUS-LIST

<u>Command</u>	<u>Operands</u>
STATUS-LIST	STATUS-INFORMATION

The STATUS-LIST command allows an office system node to notify:

- A recipient node that distribution information is available for delivery.
- A source node that status information about the progress of one or more outstanding distribution requests is available for delivery.

The recipient can use the OBTAIN or LIST command to obtain the distribution or status information. A source node can use the LIST command to retrieve outstanding status information.

Operand Descriptions

STATUS-INFORMATION

The STATUS-INFORMATION (Format 1) operand specifies the type of distribution information available to the signed-on source or recipient.

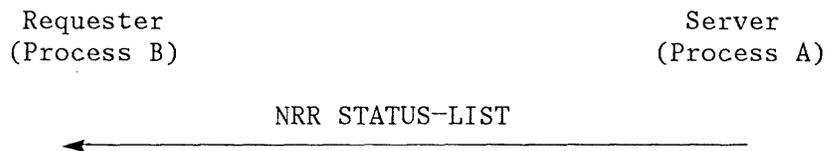
Request/Reply Protocol

The following scenarios illustrate possible replies to a STATUS-LIST command:

- Scenario 1 - Normal Completion

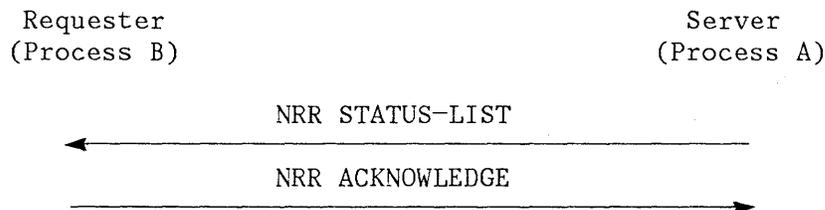
The server sends the STATUS-LIST command using the NRR command class; no replying command is expected.

The server can send the STATUS-LIST command only if the source or recipient node is in the QUIET state, a period of inactivity within the DIA session.



- Scenario 2 - Exception Condition.

When exception conditions occur during the processing of the STATUS-LIST command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." In addition to the general exception conditions, the following exception condition is specific to the STATUS-LIST command.

- The STATUS-INFORMATION operand contains an undefined type of status information.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception Data = LLIDF and data of STATUS-INFORMATION operand

CHAPTER 12. APPLICATION SERVICES

Application Processing Services Commands

Application Processing Services commands request the execution of tasks by another process. These commands are summarized here.

- The DELIVER command transports a document from a server node to a requester node.
- The EXECUTE command requests that an office system node invoke the named process for execution.
- The FORMAT command requests that an office system node execute the named formatting process using the identified document for the format input object.
- The MODIFY command requests that an office system node revise document control information fields. A field must be uniquely identified as a specific occurrence of a DIA parameter with an assigned code point and format.

The following figure shows how these commands are grouped into the Application Processing Services function set.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTER)
EXECUTE	SRR	receive	send
FORMAT	SRR	receive	send
MODIFY	SRR	receive	send
DELIVER	NRR	send	receive
ACKNOWLEDGE	NRR	send/rec	send/rec
SIGN-ON Request	SRR	receive	send
SIGN-ON Reply	NRR	send	receive
SIGN-OFF	NRR	send/rec	send/rec

Figure 43. Function Set 9

DELIVER

<u>Command</u>	<u>Operands</u>
DELIVER	IDENTIFIED-DATA, CORRELATION

The DELIVER command is the replying command that returns a document from a command server to the command requester. An example of this is returning a formatted document in reply to a FORMAT command.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be delivered. Only Format 1 of this operand is valid in support of the DELIVER command.

The IDENTIFIED-DATA (Format 1) operand specifies that the document to be delivered is located in the DIU document unit.

CORRELATION

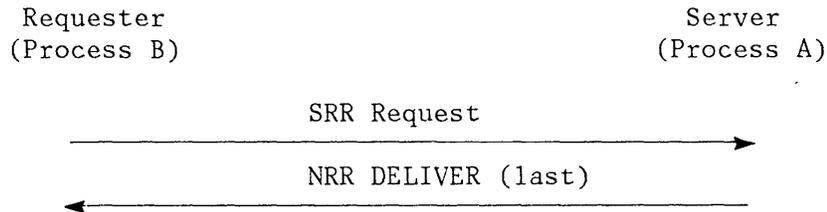
The DELIVER command uses the CORRELATION (Format 1) operand to correlate this replying command to a previously sent request. The CORRELATION (Format 1) operand identifies the request to which this DELIVER command is replying and indicates that no further replying commands will be sent.

Request/Reply Protocol

The following scenarios illustrate possible replies using the DELIVER command.

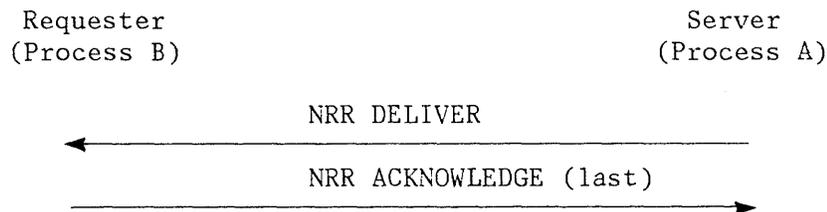
- Scenario 1 - Single Reply

The following is an example of a single request and reply.



- Scenario 2 - Exception Conditions.

When exception conditions occur during the processing of the DELIVER command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The receiver rejects the DELIVER command if any of the following conditions exist.

- The receiver is not in a state in which it can receive data.
Exception = Catastrophic, Session, Intervention-Required, Unknown
Exception Code = X'C11217'
- The IDENTIFIED-DATA operand has been omitted.
Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
- The CORRELATION operand does not refer to a command previously sent by the receiver.
Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of CORRELATION operand
- The IDENTIFIED-DATA operand refers to non-existent data.
Exception = Catastrophic, Semantic, Data-Not-Found, Operand-Value
Exception Code = X'C30709'
Exception Data = LLIDF and data of IDENTIFIED-DATA operand
- The Document Unit type is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit
Exception Code = X'C3020C'
Exception Data = LLIDF of Document Unit Introducer
- The Document Content Introducer type is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported,
Document-Content-Introducer
Exception Code = X'C30210'
Exception Data = LLIDF of Document Content Introducer
- The Document Type in the Document Unit ID is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Unit ID
Exception Code = X'C3020D'
Exception Data = LLIDF and data of Document Unit ID
- The Document Type parameter in the Base Subprofile is not supported.
Exception = Catastrophic, Semantic, Data-Not-Supported,
Document-Profile-Parameter
Exception Code = X'C3020F'
Exception Data = LLIDF and data of Document Type parameter

- The document profile is not supported.

Exception = Catastrophic, Semantic, Data-Not-Supported, Document-Profile
Exception Code = X'C3020E'
Exception Data = LLIDF of Document Profile

- The resources required of the receiving processes are unavailable.

Exception = Catastrophic, Process, Resource-Not-Available, Document-Unit
Exception Code = X'C4040C'

- The receiving process cancels the delivery of the data.

Exception = Catastrophic, Process, Cancelled, Command
Exception Code = X'C41407'

EXECUTE

<u>Command</u>	<u>Operands</u>
EXECUTE	PROCESS-NAME [, IDENTIFIED-DATA] [, ORIGINATING-NODE-ADDRESS] [, SOURCE-ADDRESS] [, SOURCE-PASSWORD] [, PROCESS-PARAMETERS] [, PROCESS-PASSWORD]

The EXECUTE command schedules a named program or procedure for asynchronous processing.

The library user has the option of passing input data to the scheduled program or procedure for processing. The input data can be included in the processing request (DIU) or in the server's document library. Additional input processing parameters can also be passed to the program or procedure when the program or procedure is invoked.

Output from the asynchronously scheduled program or procedure can be returned to the requester using the DIA Document Distribution Services or by means of non-DIA facilities.

Operand Descriptions

PROCESS-NAME

The PROCESS-NAME (Format 1) operand specifies the 1- to 32-character name of the program or procedure to be invoked.

IDENTIFIED-DATA

The IDENTIFIED-DATA operand, if present, identifies the document to be processed by the program or procedure scheduled by the EXECUTE command. Formats 1, 3, and 42 are valid in support of the EXECUTE command.

The IDENTIFIED-DATA (Format 1) operand specified that the document to be processed is in the DIU document unit.

The IDENTIFIED-DATA (Format 3) operand identifies the document to be processed by means of the position of the reference in a specified Search Result List. The operand can specify the type of document object to be passed for processing. The document object is identified in the Document Reference field of the operand. The following document object types can be referred to by this command using the IDENTIFIED-DATA (Format 3) operand:

- The document content and associated profile
- The document content only
- The document profile only.

The IDENTIFIED-DATA (Format 42) operand can identify the document to be processed by means of its Library-Assigned Document Name (LADN).

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand specifies the 1- to 8-character group address token of the requester that initiated this request. If this operand is omitted, the group address token of the command sender is assumed.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS operand (Format 1 or Format 42) is valid. If this operand is omitted, the element address token of the command sender is assumed.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated the request. The SOURCE-PASSWORD (Format 1) operand is required if the SOURCE-ADDRESS (Format 1) is specified. The SOURCE-PASSWORD (Format 1) operand is required if all the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different from the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

Otherwise, the SOURCE-PASSWORD should be omitted.

PROCESS-PARAMETERS

The PROCESS-PARAMETERS (Format 1) operand specifies 1 to 32762 bytes of input parameters that can be passed to the named program or procedure to control execution options. The named process or procedures define the syntax of the input parameters. If this operand is omitted, no input parameters are passed when the program or procedure is invoked.

PROCESS-PASSWORD

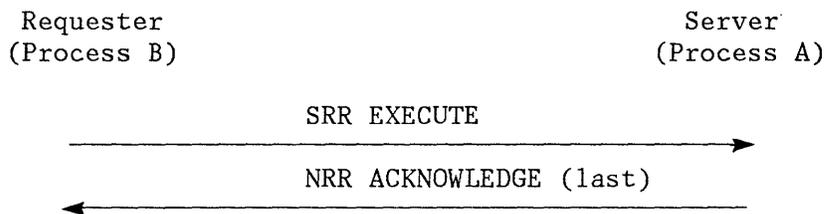
The PROCESS-PASSWORD (Format 1) operand specifies the 1- to 8-character authorization key that permits the executing of the named process.

Request/Reply Protocol

The following scenarios illustrate possible replies to the EXECUTE command:

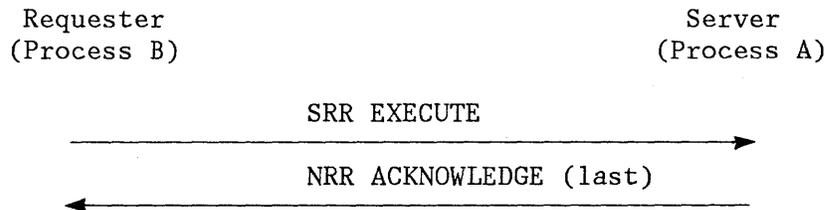
- Scenario 1 - Normal Completion

The reply command to an EXECUTE command is an ACKNOWLEDGE command that is sent to the requester when the named program or procedure is successfully scheduled for execution.



- Scenario 2 - Exception Conditions

When exception conditions occur during the processing of the EXECUTE command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exceptions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The following exception conditions are specific to the EXECUTE command. The command processor detects and reports these exception conditions in addition to the general exception conditions.

- The SOURCE-PASSWORD operand is present, and the SOURCE-ADDRESS (Format 1) operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception data = LLIDF of SOURCE-ADDRESS (Format 1) operand

- The SOURCE-ADDRESS operand identifies a requester other than the command sender; the specified requester does not have affinity with the command sender. The specified requester has a password, and the SOURCE-PASSWORD operand is not specified.

Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
Exception Code = X'C30708'
Exception data = LLIDF of SOURCE-PASSWORD operand

- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

- The ORIGINATING-NODE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of ORIGINATING-NODE-ADDRESS operand and data

- The SOURCE-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception data = LLIDF of SOURCE-PASSWORD operand and data

- The process or procedure name that the PROCESS-NAME operand specifies cannot be found.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
Exception Code = X'C40709'
Exception data = LLIDF of PROCESS-NAME operand and data

- The document specified in the IDENTIFIED-DATA operand cannot be found.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
Exception Code = X'C40709'
Exception data = LLIDF of IDENTIFIED-DATA operand and data

- The specified requester is not authorized to have access to the document specified in the IDENTIFIED-DATA operand.

Exception = Catastrophic, Process, Unauthorized-Access, Operand-Value
Exception Code = X'C40309'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

FORMAT

<u>Command</u>	<u>Operands</u>
FORMAT	IDENTIFIED-DATA, FORMATTER-NAME [,FORMATTED-DOCUMENT-NAME] [,ORIGINATING-NODE-ADDRESS] [,SOURCE-ADDRESS] [,SOURCE-PASSWORD] [,FORMAT-PARAMETERS]

The FORMAT command invokes a formatter to transform data from one format to another format, for example, a revisable-form-text document into a final-form-text document.

The data to be formatted can either be submitted with the request (DIU) or can be located in the server's document library.

The output from the program that formats the data can be either returned to the requester or filed in the server's document library.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be formatted. Formats 1, 3, and 42 are valid in support of the IDENTIFIED-DATA operand.

The IDENTIFIED-DATA (Format 1) operand specifies that the document to be delivered is in the DIU document unit.

The IDENTIFIED-DATA (Format 3) operand identifies the document to be formatted by the position of the document reference. The Document Reference field of the operand identifies the document object. The following document object types may be specified for formatting using the IDENTIFIED-DATA (Format 3) operand:

- The document content and associated profile
- The document content only.

The IDENTIFIED-DATA (Format 42) operand identifies the document to be formatted by means of its Library-Assigned Document Name (LADN).

FORMATTER-NAME

The FORMATTER-NAME (Format 1) operand specifies the 1- to 32-character name of the formatting program to be invoked.

FORMATTED-DOCUMENT-NAME

The FORMATTED-DOCUMENT-NAME (Format 1) operand specifies the 1- to 44-character name to be assigned to the formatted data that is stored in the server's document library. If the operand is omitted, the formatted data is returned to the requester using the DELIVER command.

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand specifies the 1- to 8-character group address token of the requester that initiated this request. If this operand is omitted, the group address token of the command sender is assumed.

SOURCE-ADDRESS

The SOURCE-ADDRESS operand specifies the element address token of the requester that initiated this request; either SOURCE-ADDRESS operand (Format 1 or Format 42) is valid. If this operand omitted, the element address token of the command sender is assumed.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated this request. The SOURCE-PASSWORD (Format 1) operand is required if the SOURCE-ADDRESS (Format 1) operand is specified. The SOURCE-PASSWORD (Format 1) operand is required if all the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token different than the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

Otherwise, the SOURCE-PASSWORD should be omitted.

FORMAT-PARAMETERS

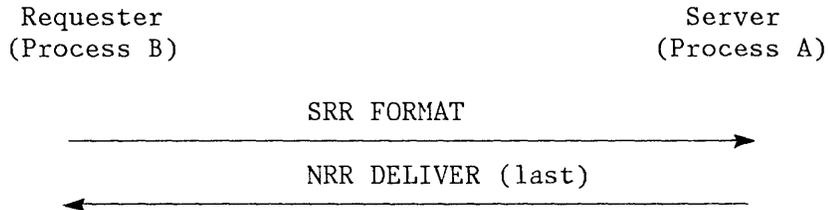
The FORMAT-PARAMETERS (Format 1) operand specifies 1 to 32762 bytes of input data parameters that are passed to the named formatting program. These input data parameters control the format processing options within the formatting program. The formatting program defines the syntax of these parameters. If this operand is omitted, the formatting program is invoked without input data parameters.

Request/Reply Protocol

The following scenarios illustrate possible replies to the FORMAT command:

- Scenario 1 - Formatted data returned to the requester

If the FORMAT command does not contain the FORMATTED-DOCUMENT-NAME operand, the output from the formatting program is returned to the requester using one or more replying DELIVER command(s). The number of replying DELIVER commands is determined by the formatting program that is invoked.

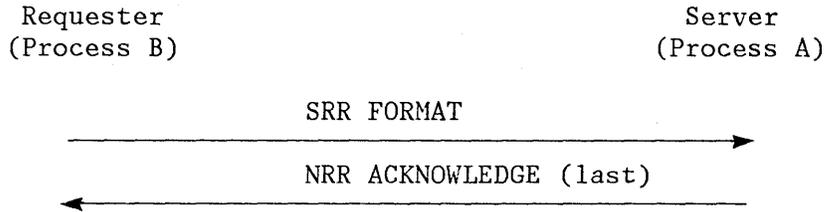


Because the FORMAT command is a specific request, the document(s) returned by the formatting program are not subject to the filtering or transformation defined when the command sender's DIA session was established.

- Scenario 2 - Formatted data filed in the document library

If the FORMATTED-DOCUMENT-NAME operand is specified, the formatted data is filed in the server's document library according to the semantics of the DIA Document Library Services FILE command. The filed document is assigned a Library-Assigned Document Name (LADN). The output is stored as a private document, and the requester of the FORMAT command is defined to be the primary owner of the document. The name supplied in the FORMATTED-DOCUMENT-NAME operand is used as the base subprofile Document Name parameter when the document is filed into the document library. The formatting program is also responsible for specifying the Document-Type and Profile CGCSGID base subprofile parameters prior to filing the document into the document library.

The replying command to a filed, formatted document is an ACKNOWLEDGE command with no exception code indications in the EXCEPTION-CODE operand and the Library-Assigned Document Name (LADN) returned in the REPLY-DATA operand. The file server generates the LADN for the formatted document by concatenating the document library node address token and the date and time that the file process was successfully completed.

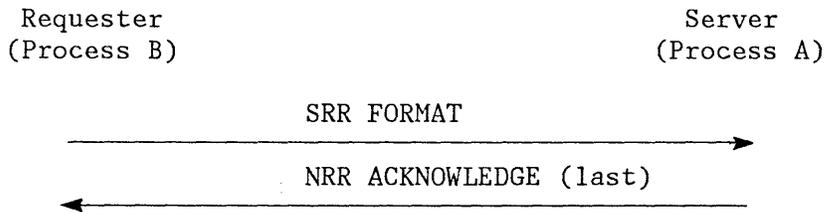


The format of the REPLY-DATA operand is defined as follows:

REPLY-DATA	LIBRARY-ASSIGNED DOCUMENT NAME
INTRODUCER	INTRODUCER
LLIDF	LLIDF - IDD (Format 42)
X'nnnnC34501'	X'nnnnC32042' (IDD (Format 42) Introducer)
	LT X'0A01' Date and Time
	X'YYMDhms'hs'
	LT X'nn02' Document Library Node address
	C'ccc . . . c' 1- to 8-byte character string

- Scenario 3 - Exception Conditions.

When exception conditions occur during the processing of the FORMAT command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The following exception conditions are specific to the FORMAT command. The command processor detects and reports these specific conditions in addition to the general exception conditions.

- The SOURCE-PASSWORD operand is present, but the SOURCE-ADDRESS (Format 1) operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception data = LLIDF of SOURCE-ADDRESS (Format 1) operand

- The SOURCE-ADDRESS operand specifies a requester that differs from the command sender; the specified requester does not have affinity with the command sender. The specified requester has a password, but the SOURCE-PASSWORD operand is not specified.

Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
Exception Code = X'C30708'
Exception data = LLIDF of SOURCE-PASSWORD operand

- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

- The ORIGINATING-NODE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of ORIGINATING-NODE-ADDRESS operand and data

- The SOURCE-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception data = LLIDF of SOURCE-PASSWORD operand and data

- The document specified in the IDENTIFIED-DATA operand cannot be found.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
Exception Code = X'C40709'
Exception data = LLIDF of IDENTIFIED-DATA operand and data

- The specified requester is neither an owner nor an owner-delegate of the document specified in the IDENTIFIED-DATA operand. The requested service cannot be executed without the requester being a designated document owner.

Exception = Catastrophic, Process, Unauthorized-Access, Operand-Value
Exception Code = X'C40309'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

- The formatting process specified in the FORMATTER-NAME operand cannot be found.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
Exception Code = X'C40709'
Exception data = LLIDF of FORMATTER-NAME operand and data

MODIFY

<u>Command</u>	<u>Operands</u>
MODIFY	IDENTIFIED-DATA, MODIFY-DATA [,ORIGINATING-NODE-ADDRESS] [,SOURCE-ADDRESS] [,SOURCE-PASSWORD]

Authorized requesters use the MODIFY command to add or delete parameters in document-related objects. The document-related objects that can be modified are document access codes and search parameters. Document profile parameters define search parameters, and the access code identifiers define the access codes.

To modify document access codes, the requester must be the primary owner of the document. Both primary and delegate owners are allowed to modify document search parameters.

Operand Descriptions

IDENTIFIED-DATA

The IDENTIFIED-DATA operand identifies the document to be modified. Formats 3 and 42 are valid in support of the MODIFY command.

The IDENTIFIED-DATA (Format 3) operand identifies a document by the position of the document reference in the specified Search Result List. The Document Reference field of the operand identifies the document object. The MODIFY command allows document object type, document profile only, to be specified via the IDENTIFIED-DATA (Format 3) operand.

The IDENTIFIED-DATA Format 42 operand identifies the document to be modified by means of its Library-Assigned Document Name (LADN).

MODIFY-DATA

The MODIFY-DATA (Format 41) operand identifies the parameters to be added or deleted within the specified DIA objects. The operand specifies the following:

- DIA object identifier
- Parameter to be modified
- Operator code - add or delete
- Value to be added or deleted.

The operand can include one or more sets of these data fields.

The DIA Object Identifier field identifies the object in which the parameter to be modified is defined. This field is required and is always 3 bytes long. The three bytes specify the IDF introducer that is assigned to each particular DIA object. For example, if the parameter to be modified is a search parameter, this field specifies the identifier (IDF) of the subprofile (such as the base subprofile or Document Library Services application subprofile), in which the search parameter is defined.

The second data field identifies the parameter to be modified. This field is required and is 3 bytes long. The three bytes specify the identifier (IDF) assigned to each specific object parameter. For example, if an author search parameter is to be added, the DIA object identifier (IDF) of the base subprofile (where the author parameter is defined) is specified in the first data field, and the parameter identifier (IDF) of the author profile parameter is specified in the second data field. If the object itself is the parameter to be modified, then the encoding of the parameter data field is defined to be X'000000'. For example, the access code DIA object is itself the parameter to be modified. Therefore, the DIA object identifier field is sufficient to uniquely identify the parameter to be modified.

The third data field, the operation code, defines the operation to be performed: X'01' for adding to the parameter or X'02' for deleting from the parameter.

The fourth data field, the parameter value, contains the 1- to 245-byte data value. For requests adding parameters, the data value is appended to the DIA object parameter. For requests deleting parameters, the data value is used to locate and delete the DIA object parameter.

ORIGINATING-NODE-ADDRESS

The ORIGINATING-NODE-ADDRESS (Format 1) operand is optional. It specifies the 1- to 8-character group address token of the requester that initiated the request. If omitted, the group address token of the command sender is assumed.

SOURCE-ADDRESS

The SOURCE-ADDRESS (Format 1) operand is optional. It specifies the element address token of the requester that initiated the request; either SOURCE-ADDRESS operand (Format 1 or Format 42) is valid. If this operand is omitted, the element address token of the command sender is assumed.

SOURCE-PASSWORD

The SOURCE-PASSWORD (Format 1) operand specifies the 1- to 8-character access authorization key of the requester that initiated the request. The SOURCE-PASSWORD (Format 1) operand is required if the SOURCE-ADDRESS (Format 1) operand is specified. The SOURCE-PASSWORD (Format 1) operand is required if all the following are true:

- The SOURCE-ADDRESS (Format 1) operand specifies an element address token that differs from the command sender.
- The specified requester does not have affinity with the command sender.
- The specified requester has a password.

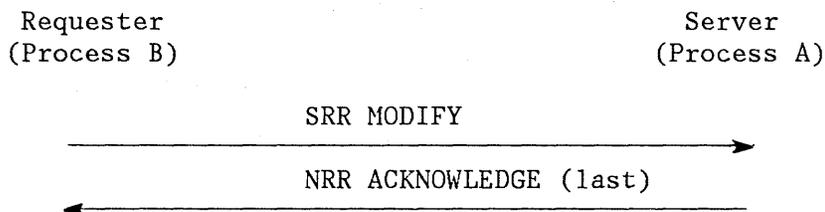
Otherwise, the SOURCE-PASSWORD should be omitted.

Request/Reply Protocol

The following scenarios illustrate possible replies to the MODIFY command.

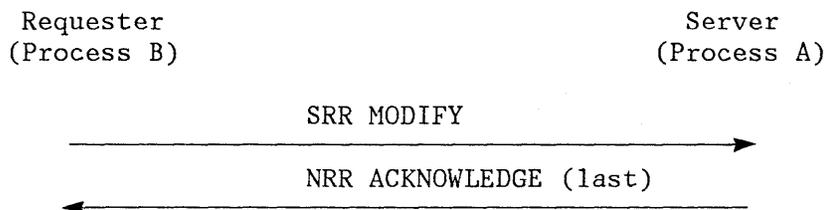
- Scenario 1 - Normal Completion

The replying command to a MODIFY command is an ACKNOWLEDGE command that is sent to the requester when the specified control object parameters have been successfully added and/or deleted.



- Scenario 2 - Exception Conditions.

When exception conditions occur during the processing of the MODIFY command, the replying ACKNOWLEDGE command contains the exception condition in the EXCEPTION-CODE operand.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in Appendix F, "DIU General Exception Conditions." The following exception conditions are specific to the MODIFY command. The command processor detects and reports specific exception conditions in addition to the general exception conditions.

- The SOURCE-PASSWORD operand is present, but the SOURCE-ADDRESS (Format 1) operand is not present.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception data = LLIDF of SOURCE-ADDRESS (Format 1) operand
- The SOURCE-ADDRESS operand specifies a requester different from the command sender; the requester does not have affinity with the command sender. The requester has a password, but the SOURCE-PASSWORD operand is not specified.

Exception = Catastrophic, Semantic, Data-Not-Found, Command-Operand
Exception Code = X'C30708'
Exception data = LLIDF of SOURCE-PASSWORD operand
- The SOURCE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of SOURCE-ADDRESS operand and data
- The ORIGINATING-NODE-ADDRESS operand contains an invalid address.

Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
Exception Code = X'C30209'
Exception data = LLIDF of ORIGINATING-NODE-ADDRESS operand and data
- The SOURCE-PASSWORD operand contains an invalid authorization key.

Exception = Catastrophic, Semantic, Password-Invalid, Operand-Value
Exception Code = X'C30509'
Exception data = LLIDF of SOURCE-PASSWORD operand and data
- The document specified in the IDENTIFIED-DATA operand cannot be found.

Exception = Catastrophic, Process, Data-Not-Found, Operand-Value
Exception Code = X'C40709'
Exception data = LLIDF of IDENTIFIED-DATA operand and data
- The requester is not a primary owner or delegate-owner of the document specified in the IDENTIFIED-DATA operand. The requested function cannot be processed without a designated document owner.

Exception = Catastrophic, Process, Unauthorized-Access, Operand-Value
Exception Code = X'C40309'
Exception data = LLIDF of SOURCE-ADDRESS operand and data

APPENDIX A. OPERANDS

This section contains a detailed discussion of each DIA operand. Each discussion includes an illustration of the operand structure.

ACCESS-CODE (FORMAT 41)

The ACCESS-CODE (Format 41) operand defines document library-user group codes that control accessibility to documents by nonowners. Members of a user group associated with a particular document are permitted read-only access to the document.

LL	I	D	F	LT	ACCD	LT	RNGE
X'nnnn'	X'C3'	X'39'	X'41'	X'nn01'	ACCD	X'nn02'	RNGE
0	2	3	4	5	7	m	n v

The ACCD field consists of one or more access codes that are preserved in the destination document library. The ACCD field is a fixed-length, 4-byte decimal number from 0 to 2047. The ACCD field is assigned a T value of X'01'. The ACCESS-CODE operand may also be specified as a set of values that are defined as a range of codes beginning with the value specified by an ACCD field. The RNGE field is used to specify the upper boundary of the set of values that are between the values of ACCD and RNGE. The RNGE field may not be specified with ACCD values of 0 (zero) and null (4 bytes of binary zero). The permitted range of values are all of the integers beginning with ACCD and increasing, by one, up to and including the value specified by the RNGE field. The RNGE field is assigned a T value of X'02' and is specified as a 4-byte decimal value from 2 to 2047. The RNGE field must be immediately preceded by an ACCD field that has a value less than the value of the RNGE that follows it. Both individually-specified ACCD values and ranges of access code values can be specified using the RNGE field in one ACCESS-CODE operand. Multiple ranges of access codes can also be specified by orderly pairing fields of ACCD and RNGE values. For example, an ACCD value of 0001 and a RNGE value of 0099, followed by another ACCD value of 0201 and a RNGE value of 0399, would produce 298 access codes for all of the integer values from 1 to 99 and from 201 through 399. An ACCD value of 0 (zero) means that access codes do not control the identified document (that is, the document is public) and that any requester can access it. A value of null (4 bytes of binary zero) means that the identified document has no access codes (that is, the document is private) and can be accessed only by the primary owner and owner-delegates. The length of the ACCESS-CODE operand and the number of assigned access code values and ranges are determined from the LL bytes of the operand introducer.

ATTRIBUTE-LIST (FORMAT 1)

The ATTRIBUTE-LIST (Format 1) operand specifies the delivery characteristics that will be in effect while the information is being distributed.

LL	I	D	F		
X'nnnn'	X'C3'	X'05'	X'01'	ATL	
0	2	3	4	5	v

The ATL operand value has the following format:

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
COD	1	binary
No		X'00'
Yes		X'01'
Reserved		X'02' - X'FF'
Personal	1	binary
No		X'00'
Yes		X'01'
Reserved		X'02' - X'FF'
Priority	1	binary
No		X'00'
Yes - Highest level		X'01'
Reserved - Future levels		X'02' - X'FF'
'Retired'	1	binary
Reserved		X'01'
Message LLIDF	5	binary
LL		X'xxxx'
Class		X'C3'
Type		X'25'
Format		X'01'
Message	v	character

For DIA Version 1.0 coexistence and migration only, the MESSAGE operand can be contained within the data variable of the ATTRIBUTE-LIST operand, as shown, when the command is sent from a source node to an OSN. The OSN must remove the message from the ATTRIBUTE-LIST operand and create a MESSAGE operand for delivery.

Field Descriptions

The COD field must appear in the ATTRIBUTE-LIST operand. The COD field values that are allowed and their meanings are defined as follows:

- X'00' - COD is not requested when this information is delivered to the specified recipients.
- X'01' - COD is requested when this information is delivered to the specified recipients.
- X'02' - X'FF' - Reserved.

The Personal field must appear in the ATTRIBUTE-LIST operand. The Personal field values that are allowed and their meanings are defined as follows:

- X'00' - The information is not personal to the specified recipients.
- X'01' - The information is personal to the specified recipients.
- X'02' - X'FF' - Reserved

The Priority field must appear in the ATTRIBUTE-LIST operand. The Priority field values that are allowed and their meanings are defined as follows:

- X'00' - Priority handling is not required for this information, for the specified recipients.
- X'01' - Highest priority handling is required for this information, for the specified recipients.
- X'02' - X'FF' - Reserved for future levels of priority. Any future levels of priority that are assigned will be in descending order. The value X'01' represents the highest priority, X'02' the next highest priority, X'03' the next, and so on, with each succeeding value being the next lower priority.

The retired field is a reserved field with a value of X'01'.

BIT STRING-REPRESENTATION (FORMAT 1)

The BIT-STRING-REPRESENTATION (Format 1) operand refers to a data unit within the same DIU. The BSR data unit must be present. The BSR operand always refers to the first BSR data unit of the contiguous group of data units that represent the control sheets. The BSR data unit will always follow the Scan-Data data unit if it is present. The BSR data unit is a self-describing entity and will be identified by an ID byte of X'C63B'.

LL	I	D	F	
X'nnnn'	X'C4'	X'3B'	X'01'	BSR
0	2	3	4	5 6

The BSR operand value is a 1-byte reference to a data unit.

The BSR data unit is defined as follows:

BSR DATA UNIT (Bit String Representation Data Unit)

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
LLIDF	5	binary
LL		X'nnnn'
Class		X'C6'
Type		X'3B'
Format		X'x1'
Bit-String-Representation		control sheet information

Field Descriptions

The Bit-String-Representation field contains control sheet information in the form of a bit string. The receiving office system node (OSN) interprets this bit string to determine the operation to be performed, and the associated operand values to be used.

CANCEL-ACTION (FORMAT 1)

The CANCEL-ACTION (Format 1) operand identifies the information and defines the action to be taken. The Action-Code field and the Distribution-Document-Name field must appear in the CANCEL-ACTION operand. The CANCEL-ACTION operand is defined as follows:

LL	I	D	F	
X'nnnn'	X'C3'	X'17'	X'01'	CAC
0	2	3	4	5 v

The CANCEL-ACTION operand value has the following format:

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
Action-Code	1	binary
Reserved		X'00'
Delivery		X'01'
COD		X'02'
character		Distribution-Document-Name LL-6

Field Descriptions

Action-Code

Specifying Delivery cancels the delivery of information scheduled to be delivered to this recipient.

Specifying COD deletes the system retention of the status of information sent COD by this source.

Distribution-Document-Name

This field will contain the Distribution-Document-Name. The Distribution-Document-Name comprises an 8-byte originating node address and the 8-byte identifier for the distribution requester and a 4-byte sequence number that is incremented each time data is distributed for the requester identified by the first sixteen bytes of this field. This is done to ensure uniqueness while the information is being transported from the source node to the recipient node.

CHARGE-CODE (FORMAT 1)

The CHARGE-CODE (Format 1) operand identifies the account to be used to accrue any charges incurred during the DIA session.

LL	I	D	F	
X'nnnn'	X'C3'	X'0F'	X'01'	CC
0	2	3	4	5 v

The CC operand value is a variable-length character string.

CONTROL-VALUE (FORMAT 1)

The CONTROL-VALUE (Format 1) operand specifies both the variable name and the value to be associated with that variable as a control value. This operand is used when the control value is to be established, changed, or deleted.

The operand is defined as follows:

LL	I	D	F	
X'nnnn'	X'C3'	X'21'	X'01'	CV
0	2	3	4	5

The fields contained in the CV operand and their descriptions follow:

<u>FIELD</u>	<u>LENGTH</u>	<u>CONTENTS</u>
Old-Value	variable	DIA operand introducer and the old value for that operand
New-Value	variable	DIA operand introducer and the new value for that operand
Authorization Value	variable	DIA operand introducer and value for any authorization (password) that must be provided before the value of the variable specified by the Old- and New-Value fields may be set to a new value

The fields, when present, must appear in the order specified, that is, Old-Value, New-Value, then Authorization Value.

Both the Old-Value and New-Value fields must be specified for this operand. The operand identifier for both the Old-Value and New-Value must be the same and must specify a valid DIA operand. No change, addition, or deletion will be processed if the identifiers of the old and new values are not identical.

The format of the Old-Value, New-Value, and Authorization Value fields in the CV operand must be identical to their format when used as operands elsewhere in DIA (that is, they must be of the form 5-byte DIA operand introducer followed by the appropriate value).

DIA operands whose values may be changed using this command are as follows:

<u>OPERAND NAME</u>	<u>IDENTIFIER</u>	<u>AUTHORIZATION VALUE</u>
SIGN-ON-PASSWORD	X'C338'	none (see note)
DOCUMENT-PASSWORD	X'C32E'	none (see note)

Two-byte DIA operand identifiers that appear within the CONTROL-VALUE operand must indicate an immediate operand value, that is, X'C3nn'.

Note: No authorization value is used when changing the SIGN-ON-PASSWORD or DOCUMENT-PASSWORD operands for the currently signed-on user. If, however, the SET-CONTROL-VALUE command is used to change either of these passwords for a user other than the signed-on user, then an authorization value must be provided. In this case, the authorization value to change the other user's SIGN-ON-PASSWORD is the other user's SIGN-ON-ID operand. The authorization value to change the other user's DOCUMENT-PASSWORD operand is the other user's SIGN-ON-ID and SIGN-ON-PASSWORD operands. For the latter, the authorization value portion of the CONTROL-VALUE operand will contain two DIA operand introducer and value combinations. These introducer and value pairs can appear in either order.

CORRELATION (FORMAT 1)

The CORRELATION (Format 1) operand identifies the command to which this command is replying. It is used to correlate a replying command to a previously received request command. The operand is defined as follows:

LL	I	D	F	
X'nnnn'	X'C3'	X'28'	X'01'	COR
0	2	3	4	5 v

The COR operand value has the following format:

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
Reply-Indicator	1	binary
Last		X'00'
Not Last		X'01'
Reserved		X'02' - X'FF'
Command-Sequence-No.	1	binary
DIU-ID	LL-7	binary

Field Descriptions

The Reply-Indicator field specifies whether this reply is the last reply to the referenced request.

The Command-Sequence-Number field specifies a number which is equal to the position of the requesting command in the command sequence in the DIU in which the requesting command was received.

The DIU-ID field matches the DIU-ID field of the DIU Prefix in which the requesting command was received.

The combination of the DIU-ID and the Command-Sequence-Number parameters provide a unique identification by which the command can be correlated with the requesting command.

COUNT (FORMAT 1)

The COUNT (Format 1) operand is a numeric field representing a count as defined by the command in which it appears.

The operand is defined as follows:

LL	I	D	F	
X'nnnn'	X'C3'	X'3E'	X'01'	CNT
0	2	3	4	5 7

The CNT operand value is a 2-byte binary number.

DESCRIPTOR-CONTENT-DEFINITION (FORMAT 41)

The DESCRIPTOR-CONTENT-DEFINITION (Format 41) operand specifies the introducer IDF values of the document profile parameters that will be returned in the document descriptor document in a response to a RETRIEVE or SEARCH command.

LL	I	D	F	
X'nnnn'	X'C3'	X'1D'	X'41'	DCD
0	2	3	4	5 v

The DCD operand value has the following format. Each descriptor field is preceded by an LT. One or more of these descriptors may be specified in the DCD operand.

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
Descriptor LT introducer. The L is the length for one descriptor operand entry.	2	X'nn01'
Introducer IDF for the document subprofile for the profile parameters to be returned.	3	binary
Introducer IDF for the document profile parameter that will be the first parameter field in the document descriptor document. This field may be specified as 3 bytes of binary zero when a parameter introducer is not used.	3	binary
Introducer T field value of the subprofile parameter to be returned. May be zero if T values are not used in the parameter.	1	binary
o		
o		
o		
Descriptor LT introducer of next descriptor operand entry.	2	X'nn01'
Introducer IDF of the next subprofile for the parameter to be returned.	3	binary
Introducer IDF for the document profile parameter that will be the last parameter field in the document descriptor document. This field may be specified as 3 bytes of binary zero when a parameter introducer is not used.	3	binary
Introducer T field value of the subprofile parameter to be returned. May be zero if T values are not used in the parameter.	1	binary

Field Descriptions

The DCD parameter values include an LT and 3-byte fixed-length IDF of the base application, and product subprofile parameter introducers that identify the parameter values that may be retrieved in a document-descriptor document.

The ID and Format byte values of the parameters are used only to specify the profile parameters that should be included in the document-descriptor document; the length fields are not needed for this identification. The profile parameters are returned in the document descriptor document in the same order as they are specified in the DCD operand, if they are present in the profiles of the documents that the Search Result List refers to.

DESTINATION-NODE-ADDRESS (FORMAT 1)

The DESTINATION-NODE-ADDRESS (Format 1) operand specifies the OSN within the office system distribution network to which this distribution request and its related data are directed.

LL	I	D	F	
X'nnnn'	X'C3'	X'2F'	X'01'	DNID
0	2	3	4	5

The DNID operand value identifies the address of the OSN that is the destination of the distribution system function to be performed in support of the DIA command containing this operand. The DNID is a 1- to 8-byte character string.

DISTRIBUTION-IDENTIFIER (FORMAT 1)

The DISTRIBUTION-IDENTIFIER (Format 1) operand identifies the distribution request being acted upon.

LL	I	D	F	
X'nnnn'	X'C3'	X'40'	X'01'	DIID
0	2	3	4	5 v

The DIID operand value has the following format:

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
Distribution-Document-Name	20	Characters
Date-Requested (yynd)	4	binary
Time-Requested (hm)	2	binary
Distribution-Name	1-44	Characters

Field Descriptions

The Distribution-Document-Name is the name created by the originating node to provide unique identification for the information while it is the object of a given distribution request. The Distribution-Document-Name is constructed with the 8-byte originating node address and the 8-byte identifier for the requester of the distribution request, followed by a 4-byte sequence number. The sequence number is the first available 4-byte number from the series of 0001 to 9999 assigned to this requester. Each request for distribution is assigned the next higher number in the series, and when 9999 is reached, the series wraps back to one (0001).

Note: Sequence number 0000 is reserved for status.

The Date-Requested is the date of the REQUEST-DISTRIBUTION command from the source node. The Date-Requested syntax is defined as 4 bytes of discontinuous binary in the following format:

DATE-REQUESTED ::= YYMD

where: YY = 2 byte binary value of 4 digit decimal year
(for example, 1980(10): X'07BC')

M = 1 byte binary value of 2 digit decimal month
(for example, 1-12(10): X'01' - X'0C')

D = 1 byte binary value of 2 digit decimal day of
month (for example, 1-31(10): X'01' - X'1F')

The Time-Requested is the time of the REQUEST-DISTRIBUTION command from the source node. The Time-Requested syntax is defined as 2 bytes of discontinuous binary in the following format:

TIME-REQUESTED ::= hm

where: h = 1 byte binary value of 2 digit decimal hours
(for example, 0-23(10): X'00' - X'17')

m = 1 byte binary value of 2 digit decimal minutes
(for example, 0-59(10): X'00' - X'3B')

The Distribution-Name is the 1-to 44-byte name of the distribution request as known by the source. If this field is omitted, the LL will be set accordingly.

DISTRIBUTION-NAME (FORMAT 1)

The DISTRIBUTION-NAME (Format 1) operand specifies the name of the distribution request as known by the source.

LL	I	D	F	
X'nnnn'	X'C3'	X'41'	X'01'	DINA
0	2	3	4	5 v

The DINA operand value is 1-44 characters from Character Set 337. The first and last character of the name may not be a X'40' (space in Code Page 256). The allowable character set for DINA are the characters defined by CGCSGID Code Page 256 Character Set ID 337. (See Appendix D, "Graphic Character Sets" on page 279).

DOCUMENT-PASSWORD (FORMAT 1)

The DOCUMENT-PASSWORD (Format 1) operand specifies the personal document authorization key associated with the recipient's personal documents.

LL	I	D	F	
X'nnnn'	X'C3'	X'2E'	X'01'	DP
0	2	3	4	5 v

The DP operand value is a 1- to 8-byte character string.

DOCUMENT-TYPE (FORMAT 1)

The DOCUMENT-TYPE (Format 1) operand identifies the specific DIA document types that can be received during the DIA session being established by this SIGN-ON request.

LL	I	D	F	
X'mnnn'	X'C3'	X'29'	X'01'	DT
0	2	3	4	5

The DT operand value consists of a vector of one or more 2-byte document type identifiers, each of which specifies a document type identifier value for a document type that can be received.

<u>FIELD</u>	<u>LENGTH</u>	<u>CONTENTS</u>
Doc.Type ID	2	2-byte binary number specifying a document type as defined in Appendix C.

The use of this operand is restricted to SIGN-ON commands that are sent from process B nodes to process A nodes. The values that this operand specifies are applicable only to the node's receive capabilities, not to that node's send capability.

The appearance of this operand is optional for individual function sets as defined in Chapter 7, "Function Sets and Commands: Introduction." If the operand is omitted, then documents of any type can be delivered to the node during the DIA session.

When the operand is present, the document type for each document to be delivered will be checked against the specified values. If the type matches a specified value, then the document will be delivered. If there is no match, but the OSN is capable of transforming the document to a specified type, then the document will be transformed and delivered. If there is no match and document type conversion is not possible or available, then the document will not be sent. Documents enqueued for delivery to the signed-on recipient that are filtered in this session are held in the distribution queue until some action is taken to have them delivered or cancelled.

The processing exceptions that can arise during a session from the use of this option are defined in the individual command descriptions for each command requesting delivery of documents.

DIA neither requires nor specifies any mandatory document type transforms to be supported by OSN products. Conversions, when supported, must be able to insure that there is no loss of information or meaning to preserve the integrity of the distribution. There is no preferred transformation implied by the ordering of values in the DOCUMENT-TYPE operand.

This operand does not apply to the delivery of types X'0008' and X'000A'.

EXCEPTION-CODE (FORMAT 1)

The EXCEPTION-CODE (Format 1) operand contains the relevant exception indicators. The operand is defined as follows:

LL	I	D	F		
X'nnnn'	X'C3'	X'22'	X'01'	EC	
0	2	3	4	5	v

For EXCEPTION-CODE operand values, refer to Chapter 6, "Exception Detection, Classification, and Reporting."

The EXCEPTION-CODE operand is repeated on the ACKNOWLEDGE command if multiple exceptions encountered within one command of a DIU are to be reported on a single ACKNOWLEDGE command. An exception of the highest severity is to appear in the first occurrence of the EXCEPTION-CODE operand. No ordering of the exceptions is required after the first occurrence of the EXCEPTION-CODE operand.

FILE-OPTION (FORMAT 1)

The FILE-OPTION (Format 1) operand specifies the action to be taken if a duplicate document name condition exists when attempting to perform the file process.

LL	I	D	F	
X'nnnn'	X'C3'	X'31'	X'01'	FO
0	2	3	4	5 v

The FILE-OPTION operand value has the following format.

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
Options	1	binary
Reserved		X'00'
Replace		X'01'
Reject		X'02'
Reserved		X'03'-X'FF'

FIELD DESCRIPTIONS

The Options field specifies the action to be taken if a duplicate document name condition exists when attempting a file process.

X'01' specifies replace the duplicate file in the target library with the document indicated by the IDD operand.

X'02' specifies reject the FILE command if a duplicate document name is found in the target library.

FORMATTED-DOCUMENT-NAME (FORMAT 1)

The FORMATTED-DOCUMENT-NAME (Format 1) operand specifies the name of the document that is the output result of the FORMAT command. The name of the document will appear as immediate data.

LL	I	D	F	
X'nnnn'	X'C3'	X'09'	X'01'	FDN
0	2	3	4	5 v

The FORMATTED-DOCUMENT NAME operand value consists of 1 to 44 characters that identify the formatted document. The length can be determined from the LL bytes of the operand.

FORMATTER-NAME (FORMAT 1)

The FORMATTER-NAME (Format 1) operand specifies the name of a format process that will be executed using the data contained in the IDENTIFIED-DATA operand as input.

LL	I	D	F	
X'nnnn'	X'C3'	X'0C'	X'01'	FNA
0	2	3	4	5 v

The FORMATTER-NAME operand value is a 1- to 32-byte character string.

FORMAT-PARAMETERS (FORMAT 1)

The FORMAT-PARAMETERS (Format 1) operand specifies the parameters that can be specified for the named formatter to control the formatter processing options. The named formatter defines the syntax of the parameter data and preserves the FORMAT-PARAMETERS operand exactly as it is entered.

LL	I	D	F	
X'nnnn'	X'C3'	X'37'	X'01'	FPA
0	2	3	4	5 v

The FORMAT-PARAMETERS operand value contains an immediate data field. The LL bytes specify the total length of the operand.

The FORMAT-PARAMETERS operand data is the exact form of the parameters as the requester entered. The parameter data syntax is defined by each of the formatters that may be specified by the format names that support the server node supports.

FUNCTION-SET (FORMAT 1)

The FUNCTION-SET (Format 1) operand identifies the specific set of DIU functional facilities proposed for use during a DIA session and identifies the role to be assumed by each issuer of the SIGN-ON command.

LL	I	D	F	
X'nnnn'	X'C3'	X'12'	X'01'	FS
0	2	3	4	5 v

The FUNCTION-SET operand value consists of a vector of one or more 3-byte function set identifiers, each containing a 1-byte role identifier and a 2-byte function set identifier value. The function set identifiers specify which sets of DIA facilities will be used during the DIA session being established.

Support of any function set requires that the implementation for either process A or process B must recognize and process all commands designated as receive for that process. The sending of commands within any function set is determined by the product being implemented.

<u>FIELD</u>	<u>LENGTH</u>	<u>CONTENTS</u>
Role	1	X'01' = Sender assumes the role of Process A. The session partner must therefore be Process B. X'02' = Sender assumes the role of Process B. The session partner must therefore be Process A. X'03' = Sender assumes the role of both Process A and Process B. The session partner must also be both Process A and Process B. X'00' and X'04'-X'FF' Reserved.
Function Set ID	2	2-byte binary number specifying a function set as defined in Chapter 6.

GRAPHIC-CHARACTER-SET-ID (FORMAT 1)

The GRAPHIC-CHARACTER-SET-ID (Format 1) operand identifies specific character sets and code pages that can be used for data received during the DIA session being established by this sign-on request.

LL	I	D	F	
X'nnnn'	X'C3'	X'2A'	X'01'	CGCSGID
0	2	3	4	5

The CGCSGID operand value consists of a vector of one or more 4-byte Graphic Character Set Identifiers, each specifying a Character Set ID and Code Page ID combination.

<u>FIELD</u>	<u>LENGTH</u>	<u>CONTENTS</u>
Graphic Char. Set ID	4	4-byte binary number where the first two bytes specify a Graphic Character Set Global ID and the second two bytes specify the Code Page Global ID to be used with the character set.

The use of this operand is restricted to SIGN-ON commands that are sent from a process B node to a process A node. The values specified by this operand are applicable to the receive capabilities of the node and not to that node's send capability.

The appearance of this operand is optional for individual function sets as defined in Chapter 6. If the operand is omitted, then validation of CGCSGID usage in text data is not required for the DIA session and any document may be delivered that is consistent with the DOCUMENT TYPE operand specification of SIGN-ON.

When the GRAPHIC-CHARACTER-SET-ID (Format 1) operand is present in the SIGN-ON command, documents will be checked for CGCSGID usage. If the document contains no character data and is a valid document type for the session, then the document will be delivered. If the document is a valid document type and the CGCSGID usage is known to be consistent with the specified values, then the document will be delivered. If the CGCSGID usage within a valid document type is not consistent with the CGCSGIDs specified, but the OSN is capable of translating to a specified graphic character set, then the document will be translated and delivered. There is no preferred translation implied by the ordering of values in the GRAPHIC-CHARACTER-SET-ID operand.

It should be noted that validity checking for CGCSGID usage of non-DIA defined documents requires knowledge of Document Content Architectures that are outside the domain of DIA. If the data contents of the document are not known, or the OSN cannot provide validity checks on the document contents, then the document is not delivered in this DIA session.

Documents enqueued for delivery to the signed-on recipient that cannot be delivered during this session are held in the distribution queue until some action is taken to have them delivered or cancelled.

The processing exceptions that can arise from the use of this session option are defined in the individual command description for each command that requests delivery of documents.

This operand does not apply to the delivery of types X'0008' and X'000A'.

IDENTIFIED-DATA (FORMAT 1)

The IDENTIFIED-DATA (Format 1) operand specifies the location of the data being referenced by the command.

The operand refers to the DIU document unit. The operand value contains a 1-byte binary number designating the specific document unit beginning with the first document unit in the DIU, that is, the nth document unit in the DIU.

LL	I	D	F	
X'nnnn'	X'C5'	X'20'	X'01'	IDD
0	2	3	4	5 v

IDENTIFIED-DATA (FORMAT 2)

The IDENTIFIED-DATA (Format 2) operand specifies the name of the data to which the command refers.

The name of the data will appear as immediate data. This format is not allowed to refer to a document unit.

LL	I	D	F	
X'nnnn'	X'C3'	X'20'	X'02'	IDD
0	2	3	4	5 v

The IDENTIFIED-DATA operand value is 1 to 44 characters. The IDENTIFIED-DATA (Format 2) operand is used when the document to be referenced is located in a private product library (as opposed to the DIA document library).

IDENTIFIED-DATA (FORMAT 3)

The IDENTIFIED-DATA (Format 3) operand refers to a document that is a member of a specified Search Result List. The operand values are specified as immediate data.

LL	I	D	F	
X'nnnn'	X'C3'	X'20'	X'03'	IDD
0	2	3	4	5 v

The IDD operand value consists of three required data fields. The first data field is the Search Result List entry number; the second data field is the type of document reference; and the third field is the Search Result List ID. These fields have the following format:

<u>FIELD NAME</u>	<u>VALUE</u>	<u>LENGTH</u>
SRL-ENTRY-NUMBER	X'0001' - X'7FFF' X'0000' & X'8000' - X'FFFF' Reserved	2 bytes
DOCUMENT-REFERENCE	X'01' Document & document profile X'02' Document content only X'03' Profile without document X'04' Document Descriptor Document X'05' Selected document descriptors and document content X'00' & X'06' - X'FF' Reserved	1 byte
SEARCH-RESULT-LIST-ID	1 to 8 characters	1 to 8 bytes

Field Descriptions

The Search Result List-Entry-Number field is a 2-byte binary value of 1 to 32767 that specifies the number of the document result entry in the Search Result List. The list entry identifies the document that is processed by the command in which this operand is specified.

The Document-Reference field specifies the type of document object to be processed for the request. This allows reference to a library document with and without its profile, reference to only the profile, reference to a Document-Descriptor Document created from search-selected profile parameters, or reference to search-selected profile parameters and the document content. Refer to the command descriptions to determine which object types are valid for each command.

The Search-Result-List-ID (SRL-ID) field is the 1- to 8-byte search request name assigned by the requester that is the output of the SEARCH command process. The contents of the SRL-ID named object are the document references or pointers to the documents that were selected by a SEARCH command process.

IDENTIFIED-DATA (FORMAT 41)

The IDENTIFIED-DATA (Format 41) operand specifies the name of the data being referenced by the command and the character set ID and code page of the name of the data.

LL	I	D	F	LT	CGCSGID	LT	IDD	
X'nnnn'	X'C3'	X'20'	X'41'	X'0601'	CGCSGID	X'nn02'	IDD	
0	2	3	4	5	7	10	12	v

The first field of the IDENTIFIED-DATA (Format 41) operand contains the 4-byte CGCSGID of the name of the data. The second field (IDD) specifies the 1 to 44 character name of the data. Each field of the operand is assigned a T value that uniquely identifies the field. The L value preceding the T value of the field specifies the length of the data, including the LT bytes. The length of the operand is determined from the LL bytes of the operand.

IDENTIFIED-DATA (FORMAT 42)

The IDENTIFIED-DATA (Format 42) operand specifies the Library-Assigned Document Name (LADN) of the document being referenced by the command. The LADN of the document will appear as immediate data.

LL	I	D	F	LT	DTM	LT	DNID
X'nnnn'	X'C3'	X'20'	X'42'	X'0A01'	DTM	X'nn02'	DNID
0	2	3	4	5	8		

The IDD operand value consists of two fields that uniquely identify the data being referenced. The length may be determined from the LL bytes of the operand. Each field of the IDD operand is assigned a T value that designates the field that is used to qualify the named data. The L value preceding each of the fields specifies the length of the data, including the LT bytes.

Field Descriptions

The Library-Assigned Document Name (LADN) consists of the document library node address of the document library concatenated with the date and time that the Document Library Services process completed filing and naming the document DTM.DNID. The fields that are used to generate the LADN are defined in the operand by the following field descriptions:

The DTM field of the operand specifies the date and time that the library process filed the document and created the LADN. The Date-Time field is assigned a T value of X'01'. The DTM syntax is defined as 8 bytes of discontinuous binary in the following format.

DATE-TIME ::= YYMDhms

where: YY = 2-byte binary value of 4-digit decimal year
 (for example 1980(10): X'07BC')

M = 1-byte binary value of 2-digit decimal month
 (for example 1-12(10): X'01' - X'0C')

D = 1-byte binary value of 2-digit decimal day of
 month (for example 1-31(10): X'01' - X'1F')

h = 1-byte binary value of 2-digit decimal hours
 (for example 0-23(10): X'00' - X'17')

m = 1-byte binary value of 2-digit decimal minutes
 (for example 0-59(10): X'00' - X'3B')

s = 1-byte binary value of 2-digit decimal seconds
 (for example 0-59(10): X'00' - X'3B')

hs = 1-byte binary value of 2-digit decimal hundredths
 of a second (for example 0-99(10): X'00' - X'63')

The DNID field of the LADN is the node ID for the document library in which the named document resides. The DNID field is a 1- to 8-byte character string. The DNID field is assigned a T value of X'02'.

LIBRARY-NAME (FORMAT 1)

The LIBRARY-NAME (Format 1) operand specifies the name of the library in which the document being referred to can be found.

LL	I	D	F	
X'nnnn'	X'C3'	X'30'	X'01'	LIBN
0	2	3	4	5 v

The LIBRARY-NAME operand value is 1 to 44 characters.

LIBRARY-NAME (FORMAT 41)

The LIBRARY-NAME (Format 41) operand specifies the name of the library in which the document being referred to can be found, the character set ID, and the code page of the library name.

LL	I	D	F	LT		LT	
X'nnnn'	X'C3'	X'30'	X'41'	X'0601'	CGCSGID	X'nn02'	LIBN
0	2	3	4	5	7	10	12 v

The first field of the LIBRARY-NAME (Format 41) operand contains the 4-byte CGCSGID of the library name. The second field (LIBRARY-NAME) specifies the 1- to 44-character library name. Each field of the operand is assigned a T value that uniquely identifies the field. The L value preceding the T value of the field specifies the length of the data including the LT bytes. The length of the operand is determined from the LL bytes of the operand.

LIST-ACTION (FORMAT 1)

The LIST-ACTION (Format 1) operand specifies the type of status information required. The operand is defined as follows:

LL	I	D	F	LA
X'0007'	X'C3'	X'18'	X'01'	LA
0	2	3	4	5 7

The LIST-ACTION operand value has the following format:

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
List-Type	1	binary
Reserved		X'00'
Delivery Information		X'01'
Queued information only		X'02'
COD status only		X'03'
System errors only		X'04'
Invalid recipient		X'05'
Delivery Information for the recipient and recipients with affinity		X'06'
Queued information for the recipient and recipients with affinity		X'07'
COD status only for the recipient and recipients with affinity		X'08'
Routing errors only for the recipient and recipients with affinity		X'09'
Invalid recipient for the recipient and recipients with affinity		X'0A'
Reserved		X'0B' - X'FF'
Reply-Type	1	binary
Reserved		X'00'
Unformatted document unit		X'01'
Formatted document unit		X'02'
Summarized Status		X'03'
Reserved		X'04' - X'FF'

Field Descriptions

The List-Type field specifies the type of list required.

Specifying Delivery Information creates a list of the status of information in categories X'02' through X'05' for the recipients.

Specifying Queued Information creates a list of all information queued for delivery for the recipients.

Specifying COD creates a list of all distribution requests for which confirmation of delivery has been requested.

Specifying System Errors creates a list of all distribution requests for which some system error occurred.

Specifying Invalid Recipient creates a list of all distribution requests that have invalid recipients.

Specifying Delivery Information for the recipient(s) with affinity to the recipient creates a list of the status of information in categories X'07' through X'0A'.

Specifying Queued Information for the recipient(s) with affinity to the recipient creates a list of all information queued for delivery for the recipients.

Specifying COD for the recipient(s) with affinity to the recipients creates a list of all distribution requests for which confirmation of delivery has been requested.

Specifying Routing Errors for the recipient(s) with affinity to the recipient creates a list of all distribution requests for which some routing error occurred.

Specifying Invalid recipient for the recipient(s) with affinity to the recipient creates a list of all distribution requests that have invalid recipients.

The Reply-Type specifies one of the following types of reply:

- An unformatted list in a document unit
- A formatted list in a document unit
- An unformatted summarized list in a document unit.

See "Unformatted Recipient Status" on page 158 and "Unformatted Source Status" on page 160 for the definition of the unformatted status document unit.

DIA does not specify the format of the document unit for formatted list output. The 'sending' product determines the document unit type, content, and format.

Specifying Summarized Status requests the return of a document that includes the classification of all information and status that are available for delivery. The document will be returned in an unformatted summarized status document unit with a DELIVER command. See "Unformatted Summary Status" on page 157 for the definition of the unformatted summarized status document unit.

Formatted or unformatted may be specified in the Reply-Type field for any of the types of lists defined in the List-Type field of the LIST-ACTION operand.

MESSAGE (FORMAT 1)

The MESSAGE (Format 1) operand contains the message text as immediate data.

LL	I	D	F	
X'nnnn'	X'C3'	X'25'	X'01'	MSG
0	2	3	4	5 v

The MESSAGE operand value contains a 1- to 256-byte character string.

The Message operand value field may contain the graphic characters that are defined within Character Set 337 of Code Page 256 (see Figure 48 on page 286).

MESSAGE (FORMAT 2)

The MESSAGE (Format 2) operand specifies the character set and code page of the message as well as the message text as immediate data.

LL	I	D	F		
X'nnnn'	X'C3'	X'25'	X'02'	CGCSGID	MSG
0	2	3	4	5	9 v

The MESSAGE operand value is the 4-byte Graphic Character Set Identifier (CGCSGID) field followed by 1 to 256 characters of immediate data message text (MSG). The message text contains characters from the character set specified in the Graphic Character Set Identifier (CGCSGID) field. The 4-byte Graphic Character Set Identifier (CGCSGID) field specifies a 2-byte Character Set ID and a 2-byte Code Page ID.

MODIFY-DATA (FORMAT 41)

The MODIFY-DATA operand specifies the identity of the DIA object parameters and the values to be used to alter the parameter contents. The operand data consists of four data fields which identify the DIA object and parameter to be modified, specify the operation code for the desired modification process, and define the modification data values. One or more sets of these data fields may be included in the MODIFY-DATA operand. Each set of data fields must be preceded by an LT byte pair specifying a 1-byte length value and a 1-byte type value.

LL	I	D	F	
X'nnnn'	X'C3'	X'0B'	X'41'	MODIFY DATA
0	2	3	4	5 v

The total length of the operand is specified by the LL bytes.

FIELD DESCRIPTIONS

The first data field specifies the identification and format of the DIA object in which the parameters to be modified are defined. This field is required and is always 3 bytes long. The 3 bytes specify the IDF introducer assigned to each particular DIA object. For example, if the parameter to be modified is a search parameter, this field specifies the IDF of the subprofile (base subprofile, Document Library Services application subprofile, and so forth) in which the search parameter is defined.

The second data field identifies the parameter to be modified. This field is required and is 3 bytes long. The three bytes specify the IDF introducer assigned to each specific object parameter.

The third data field is the operation code that specifies either X'01' for adding to the parameter or X'02' for deleting from the parameter.

The fourth data field contains either the data value that will be added to the identified parameter or that is used to locate the data value of the identified parameter field to be deleted. The Modify-Data-Value field must include the LT identifier for all parameter data field modifications when they are specified as part of the DIA object parameter definition.

<u>LENGTH</u>	<u>TYPE</u>	<u>FIELD NAME</u>	<u>VALUE</u>
X'nn'	X'01'	MODIFY-DATA-INTRODUCER	Length and T byte introducer. The length value is total length of the modify data, including the L and T bytes.
X'iiddff'		OBJECT-INTRODUCER	The assigned IDF value for the DIA object to be modified.
X'iiddff'		PARAMETER-INTRODUCER	The assigned IDF value for the parameter to be modified. This field may be X'000000' when a parameter introducer is not used.
X'nn'		MODIFY-OPERATION-CODE	X'01' Add modify data value to identified parameter. X'02' Delete modify data value from identified parameter if present. X'00' reserved X'03' - X'FF' reserved
		MODIFY-DATA-VALUE	1- to 245-byte character string to be used to modify the identified parameter. LT identifiers must be included when defined in the parameter.

OBTAIN-OPTION (FORMAT 1)

The OBTAIN-OPTION (Format 1) defines the delivery option to be taken. The operand has the following structure:

LL	I	D	F		
X'nnnn'	X'C3'	X'1E'	X'01'	OOP	
0	2	3	4	5	v

The OOP operand value has the following format:

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
Options	1	binary
All Information		X'00'
Specific Information		X'01'
Personal Information		X'02'
All Messages (Msg. Only)		X'03'
All Information for the recipient and recipients with affinity		X'04'
Specific Information for the recipient and recipients with affinity		X'05'
All Messages (Msgs. Only) for the recipient and recipients with affinity		X'06'
Reserved		X'07' - X'FD'
All Messages (Msgs. Only) and Status		X'FE'
All Information and Status		X'FF'
Priority-Distributions	1	binary
Reserved		X'00'
Priority		X'01'
Reserved		X'02' - X'FE'
All Distributions		X'FF'
'Retired'	1	X'01'
Distribution-Document-Name	20	Characters

Field Descriptions

The Options field specifies the information to be delivered; either a specifically-named distribution request or all information enqueued for delivery can be specified. The Options field must appear in the OBTAIN-OPTION operand.

X'00' specifies All Information including personal information. Personal information is not included for users other than the requester unless the appropriate document password is supplied.

X'01' specifies Specific Information that requires that the Distribution-Document-Name for the document or message be supplied. The information is obtained for only the signed-on recipient or the specified recipient. Personal information is not delivered for users other than the requester unless the appropriate document password is supplied.

X'02' specifies Personal Information and only all personal documents or messages will be delivered. The information is obtained for only the signed-on recipient or, if the associated recipient password is also supplied, the specified recipient.

X'03' specifies All Messages (Messages only) for information that contains only a message, including information with personal messages. Personal messages are not included for users other than the requester unless the appropriate document password is supplied.

X'04' specifies All Information for the recipient(s) with affinity to the recipient, including information with personal documents or messages. Personal messages are not included for users other than the requester unless the appropriate document password is supplied.

X'05' specifies Specific Information for the recipient and recipients with affinity to the recipient which requires that the Distribution-Document-Name for the document or message be supplied. If the information is personal, the information is obtained for only the signed-on recipient or, if the associated recipient password is also supplied, the specified recipient.

X'06' specifies All Messages (Messages only) for the recipient(s) with affinity to the recipient for information that contains only a message. Personal messages are not included for users other than the requester unless the appropriate document password is supplied.

X'07' - X'FD' - Reserved

X'FE' specifies All Messages (Messages only) for information that contains only a message and all status for previously distributed documents or messages for all recipients serviced by the requesting recipient. This status will be delivered in an unformatted document unit. This option does not differentiate between personal, non-personal, priority, or non-priority distribution information. It is the responsibility of the process issuing the OBTAIN command to do document password verification for personal information.

X'FF' specifies all information and all status for previously-distributed documents or messages for all recipients serviced by the requesting recipient. This status will be delivered in an unformatted document unit. This option does not differentiate between personal, non-personal, priority, or non-priority distribution information. It is the responsibility of the process issuing the OBTAIN command to do document password verification for personal information.

Priority Distributions - these categories qualify the Options field by specifying the priority level of the documents or messages to be delivered. The Priority Distributions field must appear in the OBTAIN-OPTION operand. When Specific Information is specified in the OPTION field, the Priority-Distributions field is not used to determine the information to be delivered.

X'01' specifies that only the highest priority documents or messages will be delivered. If and when additional levels of priority are assigned, specifying a lower than higher level will cause that level and all higher levels of priority to be delivered.

X'FF' specifies that all documents or messages are to be delivered regardless of the priority level.

The 'Retired' field is a reserved field with value X'01'.

The Distribution-Document-Name field contains the name of a specific distribution request to be delivered to a recipient. When the All Information, Personal Information, All Messages, All Messages and Status, All Information and Status, All Information for the recipient and recipients with affinity, or All Messages for the recipient and recipients with affinity option is specified in the OPTIONS field, then the Distribution-Document-Name field is omitted. If the Distribution-Document-Name is specified, the option specified in the OPTIONS field takes precedence, and the Distribution-Document-Name is ignored.

ORIGINATING-NODE-ADDRESS (FORMAT 1)

The ORIGINATING-NODE-ADDRESS (Format 1) operand identifies the group address token of the command requester.

LL	I	D	F	
X'nnnn'	X'C3'	X'11'	X'01'	ONID
0	2	3	4	5

The group address token specified by the ORIGINATING-NODE-ADDRESS operand value is the node address of the OSN that originated the function to be performed in support of the command containing this operand. ORIGINATING-NODE-ADDRESS is a 1- to 8-byte character string.

PROCESS-PARAMETERS (FORMAT 1)

The PROCESS-PARAMETERS (Format 1) operand specifies the parameters that may be specified for the named process of the EXECUTE command to control execution options. The syntax of the parameter data is defined by the named process and is preserved in the operand exactly as it is entered.

LL	I	D	F	
X'nnnn'	X'C3'	X'08'	X'01'	PARM
0	2	3	4	5 v

The PROCESS-PARAMETERS operand value contains an immediate data field. The total length of the operand is specified by the LL bytes.

The PROCESS-PARAMETERS operand data is the exact form of the parameters as entered by the requester. The parameter data syntax is defined by each of the processes that can be specified by the process names supported by the server.

PROCESS-NAME (FORMAT 1)

The PROCESS-NAME (Format 1) operand specifies the name of the process, program, or procedure that is to be processed in the operating system of the server.

LL	I	D	F	
X'nnnn'	X'C3'	X'07'	X'01'	PNA
0	2	3	4	5 v

The PROCESS-NAME operand value is a 1- to 32-byte character string.

PROCESS-PASSWORD (FORMAT 1)

The PROCESS-PASSWORD (Format 1) operand specifies the process authorization key that permits executing the named process.

LL	I	D	F	
X'nnnn'	X'C3'	X'13'	X'01'	PPWD
0	2	3	4	5 v

The PROCESS-PASSWORD operand value is a 1- to 8-byte character string.

RECIPIENT-ADDRESS (FORMAT 1)

The RECIPIENT-ADDRESS (Format 1) operand specifies the element address token of the recipient.

LL	I	D	F	
X'nnnn'	X'C3'	X'06'	X'01'	RID
0	2	3	4	5

The element address token specified by the RID operand value is the recipient node address and is used at the application level to identify the user or process to which the DIA command and its related data are directed. The RID operand value is a 1- to 8-byte character string.

RECIPIENT-ADDRESS (FORMAT 42)

The RECIPIENT-ADDRESS (Format 42) operand specifies the element address token of the recipient.

LL	I	D	F	LT		LT		LT	
X'nnnn'	X'C3'	X'06'	X'42'	X'nn01'	DOMID	X'nn02'	RN	X'nn03'	GN
0	2	3	4	5					

The DOMID operand value specifies the domain ID and is used at the application level to partially identify the user or process to which the DIA command and its related data are directed. It is unique within the DNID specified by the DESTINATION-NODE-ADDRESS operand that is associated with this recipient address. If there is no associated DESTINATION-NODE-ADDRESS operand, then it must be unique within the OSN where it is received. It is a 1- to 8-byte character string.

The RN operand value specifies the recipient name and is also used at the application level to further identify the user or process. It is unique within the domain specified by the domain ID field in this RECIPIENT-ADDRESS operand. It is a 1- to 32-byte character string.

The GN field of this operand specifies the global name that is associated with the recipient. It is a 1- to 32-byte character string.

The L byte for each of these operand parts specifies the length of each construct including the two LT bytes and the 1- to 8- or 1- to 32-byte character string.

When Format 42 of the RECIPIENT-ADDRESS operand is used, it must contain one of the following combinations of the individual operand parts. All parts specified for a given combination must be present.

Domain-ID and Recipient-Name

Domain-ID and Global-Name

RECIPIENT-PASSWORD (FORMAT 1)

The RECIPIENT-PASSWORD (Format 1) operand is an access authorization key associated with a recipient.

LL	I	D	F	
X'nnnn'	X'C3'	X'0E'	X'01'	RP
0	2	3	4	5 v

The RECIPIENT-PASSWORD operand value is a 1- to 8-byte character string.

RECOVERY-ACTION (FORMAT 1)

The RECOVERY-ACTION (Format 1) operand contains an indication of the recovery action recommended by the sender of the ACKNOWLEDGE command. The operand is defined as follows:

LL	I	D	F	
X'nnnn'	X'C3'	X'27'	X'01'	RACT
0	2	3	4	5 v

For RECOVER-ACTION operand values refer to Chapter 5, "Document Interchange Unit." If the RECOVERY ACTION operand does not appear on the ACKNOWLEDGE command, the recovery action is to be determined by the receiver of the ACKNOWLEDGE command.

REPLY-DATA (FORMAT 1)

The REPLY-DATA (Format 1) operand is a command-specific operand that is used by replying commands to return data to the issuer of a reply required command.

The operand is defined as follows:

LL	I	D	F	
X'nnnn'	X'C3'	X'45'	X'01'	RPD
0	2	3	4	5 v

The REPLY-DATA operand value is a variable length field containing the reply data required by the command being replied to by the command containing this operand.

For example, the REPLY-DATA operand will be used for ACKNOWLEDGE commands that are replying to REQUEST-DISTRIBUTION commands. The REPLY-DATA operand contains the distribution document name assigned to the document to be distributed. The Distribution Document Name is a field in the Distribution-ID operand on the DELIVER command.

RETRIEVE-COUNT (FORMAT 1)

The RETRIEVE-COUNT (Format 1) operand specifies the maximum number of document descriptors that will be returned in response to a RETRIEVE or a SEARCH command.

The operand is defined as follows:

LL	I	D	F	
X'0007'	X'C3'	X'1C'	X'01'	RCNT
0	2	3	4	5 7

The RETRIEVE-COUNT operand value is a 2-byte binary number from 0 to 32767.

SCAN-DATA (FORMAT 1)

The SCAN-DATA (Format 1) operand references a data unit within the same DIU. The Scan-Data data unit does not have to be present. Because the SCAN-DATA operand is required, the reference byte of the operand must be zero whenever the Scan-Data data unit is not present. When the Scan-Data data unit is present, this operand always refers to the first data unit of the contiguous group of data units representing the control sheets. The Scan-Data data unit is a self-describing entity and will be identified by an ID of X'C63A'.

LL	I	D	F		
X'nnnn'	X'C4'	X'3A'	X'01'	SCD	
0	2	3	4	5	6

The SCAN-DATA operand value is a 1-byte reference to a data unit.

The Scan-Data data unit is defined as follows:

SCAN-DATA DATA UNIT (Non-Coded Information data unit)

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
LLIDF	5	binary
LL		X'nnnn'
ID		X'C63A'
Format		X'x1'
Scan-Data		control-sheet data

Field Descriptions

The Scan-Data data field contains the control sheet data in an image Document Content Architecture compatible form which is insertable into a document unit so that the document unit remains a valid image Document Content Architecture document.

SEARCH-DATA (FORMAT 41)

The SEARCH-DATA (Format 41) operand specifies the arguments that are used by the search process to identify documents in the document library that satisfy the requested selection criteria. The search process evaluates each of the documents within the specified OSN library. The evaluation compares the search arguments with the values of identified parameters in document profiles.

The result of each search argument and profile parameter comparison is evaluated using the first (relational) operator that is specified with each of the search arguments. The search arguments are ordered in the operand from left to right for evaluation.

The search arguments each contain a second (logical) operator that specifies the evaluation that is made with the result of the immediate predecessor argument or parameter comparison and the successor comparison.

The result of each argument or parameter comparison, and the result of each predecessor or successor comparison, is evaluated as they are completed. The references to the documents that qualify according to the search data evaluations are entered into the Search Result List.

LL	I	D	F	
X'nnnn'	X'C3'	X'33'	X'41'	SEARCH DATA
0	2	3	4	5 v

The SEARCH-DATA operand contains one or more variable length search data arguments. The total length of the operand is specified by the LL bytes.

Field Descriptions

The SEARCH-DATA operand consists of search data arguments that are preceded by a (LT) 1-byte length value and a 1-byte type value. The T byte of the LT search data introducer identifies the search data arguments that will be compared with the document profile parameters.

One or more search data arguments may be specified in the SEARCH-DATA operand. Each search data argument must be introduced by the LT byte pair.

The Interchange Document Profile (IDP) consists of subprofiles, each containing one or more parameters. These subprofiles are designated as the IDP Base, the Document Library Services Application, Product Specific, and Private User. Within each of these subprofiles, parameters are specified that describe the characteristics and document relevant information. The parameters contain data fields that provide specific document-related information.

The existence of these nested profile parameter forms requires that the search data identify the subprofile, subprofile parameters, and specific parameter data fields. The SEARCH-DATA operand comprises the fields to identify these profile structures, the search data arguments, and the operators for comparing them.

The first data field of the SEARCH-DATA operand specifies the identification and format of the subprofile in which the parameters to be searched on can appear. This field is required and is always 3 bytes long. The three bytes specify the IDF introducer assigned to each particular subprofile.

The second data field specifies the identification and format of the parameter in the specified subprofile to be searched on. This field is required and is 3 bytes long. The three bytes specify the IDF introducer assigned to each specific profile parameter.

The third data field specifies the identification of a data field within the specified profile parameter to be searched on. This field is required and is 1 byte in length. The byte specifies the T value that is assigned to each specific data field in a profile parameter. A profile parameter that does not have assigned T value data fields within it will have this Search-Data field coded with a binary zero.

The fourth data field is a relational operator that is used to evaluate the comparison of the search argument and the document profile parameter.

The comparison of search arguments and document profile parameters can be made using any one of seven relational operators. These operators permit the comparisons to evaluate whether the parameter comparator is equal, not equal, less than, greater than, less than or equal, greater than or equal, or generic equal to the argument. The generic equal operator permits comparisons of the search data argument with a variable number of bytes in the profile parameter starting with the first byte. The length and contents of the search argument controls the comparison. When the profile parameter contains the byte string value specified in the search argument, starting with the first byte of the profile parameter, then the result of the comparison is true. The generic comparison is terminated by exhausting the count of bytes in the search data argument, by the first unequal byte comparison in the profile parameter, or when the number of bytes in the profile parameter is less than the number of bytes in the search argument.

The fifth data field is a logical operator that is used to evaluate the relation of the result of the immediate predecessor argument or parameter comparison and the result of the successor argument or parameter comparison specified by the current operand values. The first logical evaluation assumes that the initial state for the immediate predecessor is true.

The sixth data field is the search argument data value that is compared with the profile parameter specified by the IDF and optionally the T byte data fields.

<u>LENGTH</u>	<u>TYPE</u>	<u>FIELD NAME</u>	<u>VALUE</u>
X'nn'	X'01'	SEARCH-DATA-PARAMETER	Length and T byte introducer. The length value is the total length of the search data argument, including the L and T bytes.
X'iiddff'		SUBPROFILE-INTRODUCER	The assigned IDF value for the subprofile to be searched.
X'iiddff'		PROFILE-PARAMETER-INTRODUCER	The assigned IDF value for the parameter to be compared. This field may be X'000000' when a parameter introducer is not used.
X'tt'		PARAMETER-TYPE-INTRODUCER	The assigned T value for the parameter data field to be compared. This field will be X'00' when a parameter data field is not used.
X'xx'		SEARCH-RELATIONAL-OPERATOR	Characters: = ≠ ≤ ≥ < > * Corresponding to: = equal, X'7E'; ≠ not equal, X'BE'; ≤ less than or equal, X'8C'; ≥ greater than or equal, X'AE'; < less than, X'4C'; > greater than, X'6E'; * generic equal, X'5C';
X'xx'		SEARCH-LOGICAL-OPERATOR	Characters: + Corresponding to: + and (logical conjunction, X'4E'; or (logical disjunction), X'4F';
1-245		SEARCH-DATA-VALUE	Hexadecimal byte string.

Document search selection is also desirable by logically evaluating two or more document profile parameters with one or more different profile parameters.

This logical evaluation can be made by using the Search-Data-Parameter-Set (SDPS) field in the SEARCH-DATA operand. The use of the SDPS field in the SEARCH-DATA operand is similar to one level of parentheses for each set of logical evaluations. The SDPS field permits the result of evaluating two or more parameter comparisons to be evaluated with one or more different parameter comparisons. The SDPS field is specified by an LT introducer with a type code of X'02'. The SDPS L byte specifies the total length of one or more SEARCH-DATA-PARAMETERS (SDP) in one set. The SDPS logical operator is specified as one byte following the T byte. The SDPS logical operator may be AND or OR for evaluating the SDPS result with the immediate predecessor. The evaluation of the first SDPS assumes the initial state to be true. The first comparison of an SDPS argument or parameter assumes the initial state to be true.

The SDPS field supports evaluations such as the following:

```
IF (ARG1=PARAM1 AND ARG2=PARAM2) OR (ARG3=PARAM3 AND ARG4=PARAM4)
  THEN SELECT DOCUMENT;
```

The SDPS field of the SEARCH-DATA operand has the following format.

<u>LENGTH</u>	<u>DATA</u>	<u>FIELD NAME</u>	<u>DESCRIPTION</u>
X'nn'	X'02'	SEARCH-DATA-PARAMETER-SET	Length and type
	X'xx'	SDPS-LOGICAL-OPERATOR	Characters: + Corresponding to: + and (logical conjunction), X'4E'; or (logical disjunction), X'4F';
		SEARCH-DATA-PARAMETER(1)	
		.	
		.	
		SEARCH-DATA-PARAMETER(n)	

The example above, comparing PARM1 and PARM2 and evaluating that result with the result of the comparison of PARM3 and PARM4, is encoded in the SEARCH-DATA operand as follows.

<u>VALUE</u>	<u>FIELD NAME</u>	<u>DESCRIPTION</u>
X'nnnn'	Operand length	Total length of SEARCH-DATA
X'C3'	Operand class	Immediate operand data
X'33'	Operand type	SEARCH-DATA operand
X'41'	Operand format	Operand LT data fields
X'nn'	Field length	Total length of SDPS1 field
X'02'	Field type	SDPS type data field
X'4E'	SDPS logical operator	AND with predecessor (true)
X'nn'	Field length	Total length of SDP1
X'01'	Field type	Search-Data-Parameter
X>IDF'	Subprofile-Introducer	Subprofile identifying PARM1
X>IDF'	Parameter-Introducer	PARM1 introducer
X'tt'	Parameter-Type-Field	Parameter data field
X'7E'	Search-Relational-Op	Compare PARM1 EQUAL-TO ARG1
X'4E'	Search-Logical-Op	Logical AND with predecessor (true)
XLn	Search-Data-Value	Search ARG1 data
X'nn'	Field length	Total length of SDP2
X'01'	Field type	Search-Data-Parameter
X>IDF'	Subprofile-Introducer	Subprofile identifying PARM2
X>IDF'	Parameter-Introducer	PARM2 introducer
X'tt'	Parameter-Type-Field	Parameter data field
X'7E'	Search-Relational-Op	Compare PARM2 EQUAL-TO ARG2
X'4E'	Search-Logical-Op	Logical AND with predecessor (SDP1)
XLn	Search-Data-Value	Search ARG2 data
X'nn'	Field length	Total length of SDPS2 field
X'02'	Field type	SDPS type data field
X'4F'	SDPS logical operator	OR with predecessor(SDPS1)
X'nn'	Field length	Total length of SDP3
X'01'	Field type	Search-Data-Parameter
X>IDF'	Subprofile-Introducer	Subprofile identifying PARM3
X>IDF'	Parameter-Introducer	PARM3 introducer
X'tt'	Parameter-Type-Field	Parameter data field
X'7E'	Search-Relational-Op	Compare PARM3 EQUAL-TO ARG3
X'4E'	Search-Logical-Op	Logical AND with predecessor (true)
XLn	Search-Data-Value	Search ARG3 data
X'nn'	Field length	Total length of SDP4
X'01'	Field type	Search-Data-Parameter
X>IDF'	Subprofile-Introducer	Subprofile identifying PARM4
X>IDF'	Parameter-Introducer	PARM4 introducer
X'tt'	Parameter-Type-Field	Parameter data field
X'7E'	Search-Relational-Op	Compare PARM4 EQUAL-TO ARG4
X'4E'	Search-Logical-Op	Logical AND with predecessor (SDP3)
XLn	Search-Data-Value	Search ARG4 data

SEARCH-DATA (FORMAT 42)

The SEARCH-DATA (Format 42) operand specifies the arguments that are used by the search process to identify documents in the document library that satisfy the request selection criteria. The SEARCH-DATA (Format 42) operand permits specifying the CGCSGID in which the search data arguments are coded. The search process is exactly the same as when the SEARCH-DATA (Format 41) operand is specified.

LL	I	D	F		
X'nnnn'	X'C3'	X'33'	X'42'	SEARCH DATA	
0	2	3	4	5	v

The SEARCH-DATA (Format 42) operand contains one or more variable length search data arguments. The total length of the operand is specified by the LL bytes.

Field Descriptions

The SEARCH-DATA operand consists of search data arguments that are preceded by a (LT) 1-byte length value and a 1-byte type value. The T byte of the LT search data introducer identifies the search data arguments that will be compared with the document profile parameters.

One or more search data arguments may be specified in the SEARCH-DATA operand. Each search data argument must be introduced by the LT byte pair.

The Interchange Document Profile (IDP) consists of subprofiles that may each contain one or more parameters. These subprofiles are designated as the IDP Base, the Document Library Services Application, Product Specific, and Private User. Within each of these subprofiles, parameters are specified that describe the characteristics and document relevant information. The parameters contain data fields that provide specific document-related information. The profile parameters may be coded in different character sets that are specified as part of the parameter.

The existence of these nested profile parameter forms requires that the search data identify the subprofile, subprofile parameters, and specific parameter data fields. The SEARCH-DATA operand comprises the fields to identify these profile structures, the search data arguments, and the operators for comparing them.

The first data field of the SEARCH-DATA operand specifies the identification and format of the subprofile in which the parameters to be searched on can appear. This field is required and is always 3 bytes long. The three bytes specify the IDF introducer assigned to each particular subprofile.

The second data field specifies the identification and format of the parameter in the specified subprofile to be searched on. This field is required and is 3 bytes long. The three bytes specify the IDF introducer assigned to each specific profile parameter.

The third data field specifies the identification of a data field within the specified profile parameter to be searched on. This field is required and is 1 byte in length. The byte specifies the T value that is assigned to each specific data field in a profile parameter. A profile parameter that does not have assigned T value data fields within it will have this Search-Data field coded with a binary zero.

The fourth data field is a relational operator that is used to evaluate the comparison of the search argument and the document profile parameter.

The comparison of search arguments and document profile parameters can be made using any one of seven relational operators. These operators permit the comparisons to evaluate whether the parameter comparator is equal, not equal, less than, greater than, less than or equal, greater than or equal, or generic equal to the argument. The generic equal operator permits comparisons of the search data argument with a variable number of bytes in the profile parameter starting with the first byte. The comparison is controlled by the length and contents of the search argument. When the profile parameter contains the byte string value specified in the search argument, starting with the first byte of the profile parameter, then the result of the comparison is true. The generic comparison is terminated by exhausting the count of bytes in the search data argument, by the first unequal byte comparison in the profile parameter, or when the number of bytes in the profile parameter is less than the number of bytes in the search argument.

The fifth data field is a logical operator that is used to evaluate the relation of the result of the immediate predecessor argument or parameter comparison and the result of the successor argument or parameter comparison specified by the current operand values. The first logical evaluation assumes that the initial state for the immediate predecessor is true.

The sixth data field is the 4-byte binary that specifies the character set and code page used for coding the Search-Data field.

The seventh data field is the search argument data value that is compared with the profile parameter specified by the IDF and, optionally, the T byte data fields.

<u>LENGTH</u>	<u>TYPE</u>	<u>FIELD NAME</u>	<u>VALUE</u>
X'nn'	X'01'	SEARCH-DATA-PARAMETER	Length and T byte introducer. The length value is the total length of the search data argument, including the L and T bytes.
X'iiddff'		SUBPROFILE-INTRODUCER	The assigned IDF value for the subprofile to be searched.
X'iiddff'		PROFILE-PARAMETER-INTRODUCER	The assigned IDF value for the parameter to be compared. This field may be X'000000' when a parameter introducer is not used.
X'tt'		PARAMETER-TYPE-INTRODUCER	The assigned T value for the parameter data field to be compared. This field will be X'00' when a parameter data field is not used.
X'xx'		SEARCH-RELATIONAL-OPERATOR	Characters: = # ≤ ≥ < > * Corresponding to: = equal, X'7E'; # not equal, X'BE'; ≤ less than or equal, X'8C'; ≥ greater than or equal, X'AE'; < less than, X'4C'; > greater than, X'6E'; * generic equal, X'5C';
X'xx'		SEARCH-LOGICAL-OPERATOR	Characters: + Corresponding to: + and (logical conjunction), X'4E'; or (logical disjunction), X'4F';
X'xxxxxxxx'		CGCSGID	Character set and code page of Search-Data field coding
1-245		SEARCH-DATA-VALUE	Hexadecimal byte string.

SEARCH-OPTION (FORMAT 1)

The SEARCH-OPTION (Format 1) operand defines the scope of the search to be taken. The operand has the following structure:

LL	I	D	F	
X'nnnn'	X'C3'	X'32'	X'01'	S0
0	2	3	4	5 v

The SEARCH-OPTION operand value has the following format:

<u>FIELD</u>	<u>LENGTH</u>	<u>VALUE</u>
Options	1	binary
Reserved		X'00'
Owned		X'01'
Owned or Accessible		X'02'
Reserved		X'03'-X'FF'

Field Descriptions

The Options field specifies the scope of the documents to be searched.

X'01' specifies Owned, which requires that the search include all documents which are owned or delegate-owned by the requester.

X'02' specifies Owned or Accessible, which requires that the search include all documents which are owned, delegate-owned, or accessible by the requester.

SEARCH-REQUEST-NAME (FORMAT 1)

The SEARCH-REQUEST-NAME (Format 1) operand specifies the user-assigned name that the Document Library Services server should name the SRL that is created by the SEARCH command process.

LL	I	D	F	
X'nnnn'	X'C3'	X'1F'	X'01'	SRN
0	2	3	4	5 v

The SEARCH-REQUEST-NAME operand value is a 1- to 8-byte character string.

SELECT-LIMIT (FORMAT 1)

The SELECT-LIMIT (Format 1) operand specifies the maximum number of documents that can be selected by the search process that is executed by the Document Library Services server.

The operand is defined as follows:

LL	I	D	F	
X'0007'	X'C3'	X'1B'	X'01'	SLIM
0	2	3	4	5 7

The SEARCH-LIMIT operand value is a 2-byte binary number from 1 to 32767.

SIGN-ON-ID (FORMAT 1)

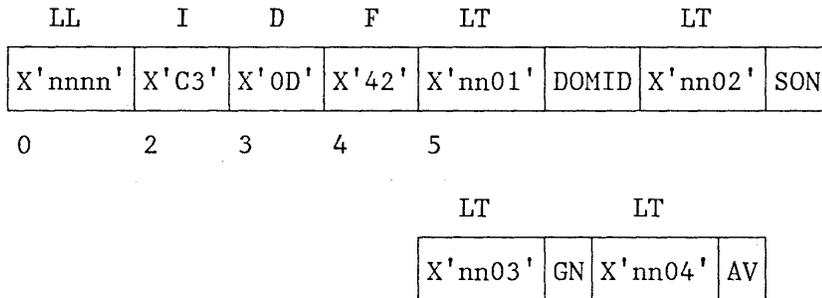
The SIGN-ON-ID (Format 1) operand identifies a DIA end user participating in a DIA session.

LL	I	D	F	
X'nnnn'	X'C3'	X'0D'	X'01'	SOI
0	2	3	4	5 v

The SIGN-ON-ID operand value is a 1- to 8-byte character string.

SIGN-ON-ID (FORMAT 42)

The SIGN-ON-ID (Format 42) operand identifies a DIA end user participating in a DIA session. The SIGN-ON-ID (Format 42) operand is used by products which cannot support standard Format 1 addressing and require the receiver to provide translation of addresses from Format 42 to Format 1 for distribution.



DOMID specifies the domain-ID, and is used at the application level to partially identify the user or process that is establishing this DIA session. It is unique within the host ONID specified for this user or process. It is a 1- to 8-byte character string.

SON specifies the sign-on name and is also used at the application level to further identify the signing-on user or process. It is unique within the domain specified by the domain-ID in this SIGN-ON-ID operand. It is a 1- to 32-byte character string.

GN specifies the global name that is associated with the signing-on user or process. It is a 1- to 32-byte character string.

AV is an authorization value that is associated with the user or process identified by either the SON or the GN value. The AV appears in lieu of the Format 1 SIGN-ON-PASSWORD operand. It is an 8-byte character string.

The L byte for each of these operand parts specifies the length of each construct including the two LT bytes and the 1- to 8- or 1- to 32-byte character string.

When format 42 of the SIGN-ON-ID operand is used, it must contain one of the following combinations of the individual operand parts. All parts specified for a given combination must be present.

Domain ID, Source Name, and Authorization Value

Domain ID, Global Name, and Authorization Value

SIGN-ON-PASSWORD (FORMAT 1)

The SIGN-ON-PASSWORD (Format 1) operand is an access authorization key associated with a DIA user participating in a DIA session.

LL	I	D	F	
X'nnnn'	X'C3'	X'38'	X'01'	SOP
0	2	3	4	5 v

The SIGN-ON-PASSWORD operand value is a 1- to 8-byte character string. The maximum length for this operand is 8 bytes.

SOURCE-ADDRESS (FORMAT 1)

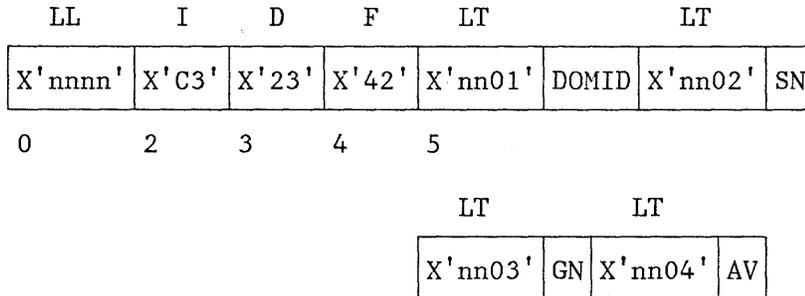
The SOURCE-ADDRESS (Format 1) operand specifies the element address token of the requester source node.

LL	I	D	F	
X'nnnn'	X'C3'	X'23'	X'01'	SID
0	2	3	4	5

The element address token specified by the SID operand value is the requester's source node address and is used at the application level to identify the user or process that is the source of the DIA command and its related data. SID is a 1- to 8-byte character string.

SOURCE-ADDRESS (FORMAT 42)

The SOURCE-ADDRESS (Format 42) operand specifies the element address token of the requester source node.



The DOMID part of this operand specifies the domain ID and is used at the application level to partially identify the user or process that is the requester of the DIA command and its related data. The domain ID is unique within the ONID specified by the ORIGINATING-NODE-ADDRESS operand that is associated with this SOURCE-ADDRESS operand. If there is no associated ORIGINATING-NODE-ADDRESS operand, then DOMID must be unique within the OSN where it is received. DOMID is a 1- to 8-byte character string.

The SN operand value specifies the source name and is also used at the application level to further identify the user or process. The source name is unique within the domain specified by the domain ID in this SOURCE-ADDRESS operand. SN is a 1- to 32-byte character string.

The GN part of this operand specifies the global name that is associated with the requester. GN is a 1- to 32-byte character string.

The AV operand value is an authorization value that is associated with the user or process identified by either the SN or the GN value. The AV part of this operand may appear only when this operand is being used in a command from a source node to an originating office system node. AV is a 1- to 8-byte character string.

The L byte for each of these operand parts specifies the length of each construct including the two LT bytes and the 1- to 8- or 1- to 32-byte character string.

When Format 42 of the SOURCE-ADDRESS operand is used, it must contain one of the following combinations of the individual operand parts. All parts specified for a given combination must be present.

Domain ID, Source Name, and Authorization Value

Domain ID and Global Name

Domain ID, Global Name, and Authorization Value

SOURCE-PASSWORD (FORMAT 1)

The SOURCE-PASSWORD (Format 1) operand is an access authorization key associated with a requester.

LL	I	D	F	
X'nnnn'	X'C3'	X'0E'	X'01'	SP
0	2	3	4	5 v

The SOURCE-PASSWORD operand value is a 1- to 8-byte character string.

STATUS-INFORMATION (FORMAT 1)

The STATUS-INFORMATION (Format 1) operand specifies the type of status information that is now available to the recipient.

The operand structure is as follows:

LL	I	D	F	
X'nnnn'	X'C3'	X'3D'	X'01'	STI
0	2	3	4	5 7

The STATUS-INFORMATION operand value is a 2-byte encoded bit string.

The defined types of status and their bit encoding follows:

- Reserved 00000000 00000000
- Priority 00000000 00000001
- Non-priority 00000000 00000010
- Personal 00000000 00000100
- COD 00000000 00001000
- System Error 00000000 00010000
- Invalid Recipient 00000000 00100000
- Reserved 01111111 11000000

Any one or any combination of codes may be used for notification of the status that is available.

Field Descriptions

Priority is specified when the status pertains to documents or messages that were distributed with the priority distribution option. This applies both to documents and messages queued for delivery.

Non-Priority is specified when the status pertains to documents or messages that were distributed without specifying priority, personal, or COD distribution options. This applies to both documents and messages that are queued for delivery.

Personal is specified when the status pertains to documents or messages that were distributed with the personal document distribution option. This applies to both documents and messages that are queued for delivery.

COD is specified when the status pertains to documents or messages that were distributed requesting confirmation of delivery. This applies to both documents and messages that were previously distributed.

System Error is specified when the status pertains to documents or messages that could not be delivered due to some system error. This applies to both documents and messages that were previously distributed.

Invalid Recipient is specified when the recipient is unknown at the destination OSN.

Status is given for the signed-on recipient only.

TIME-LIMIT (FORMAT 1)

The TIME-LIMIT (Format 1) operand specifies the maximum number of minutes that the search process should be allowed to execute by the Document Library Services server.

The operand is defined as follows:

LL	I	D	F	
X'0007'	X'C3'	X'1A'	X'01'	TLIM
0	2	3	4	5 7

The TIME-LIMIT operand value is a 2-byte binary number from 1 to 1440.

APPENDIX B. CODE POINTS

This appendix consists of tables that list the code points defined for the DIA constructs. Each table gives the description of the construct and the code point that corresponds to the structured-field identifier, that is, the IDF. The I corresponds to the construct class, and the D corresponds to the construct type. Any code point not listed in one of the tables is reserved.

The code point tables are grouped by construct class (such as Prefix, SRR Command, or Immediate Data Operand) according to the following list:

CLASS	CODE POINT CLASS	TABLE
DIU PREFIX	X'C0'	1
COMMAND - NO REPLY REQUIRED	X'C1'	2
OPERAND - IMMEDIATE	X'C3'	3
OPERAND - DATA UNIT REFERENCE	X'C4'	4
OPERAND - DOCUMENT UNIT REFERENCE	X'C5'	5
PROFILE PARAMETERS	X'C7'	6
DOCUMENT UNIT	X'C9'	7
DOCUMENT PROFILE	X'CA'	8
DOCUMENT CONTENT INTRODUCER	X'CB'	9
COMMAND - SYNC REPLY REQUIRED	X'CD'	10
DIU SUFFIX	X'CF'	11

NOTES FOR DIA CODE POINT ASSIGNMENT TABLES

All TYPE CODES of X'00' are RESERVED.

A lowercase x is used to indicate the value for bits 0 - 3 of the format byte. The actual setting of these bits must be in accordance with the rules for the specific construct in which the bits appear.

The LT form of the data field is only valid in operands and document profile parameters.

TABLE 1. DIU PREFIX ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU PREFIX	C0	
DIU PREFIX, INTERCHANGE FORM	C001	x2

TABLE 2. NO REPLY REQUIRED COMMAND ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU COMMAND - NO-REPLY-REQUIRED (NRR)	C1	
NRR - ACKNOWLEDGE	C101	x1
NRR - Assigned by SNADS	C105	x1
NRR - SIGN-ON	C10C	x1
NRR - SIGN-OFF	C10D	x1
NRR - Assigned by SNADS	C10F	x1
NRR - Assigned by SNADS	C117	x2
NRR - DELIVER	C119	x1
NRR - STATUS-LIST	C11E	x1
NRR - Assigned by SNADS	C123	x1
NRR - Assigned by SNADS	C124	x1

TABLE 3. IMMEDIATE DATA OPERAND ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
IMMEDIATE DATA OPERAND (IMMED DATA OPND)	C3	
IMMED DATA OPND ATTRIBUTE-LIST	C305	01
IMMED DATA OPND RECIPIENT-ADDRESS	C306	01
IMMED DATA OPND RECIPIENT-ADDRESS	C306	42
IMMED DATA OPND PROCESS-NAME	C307	01
IMMED DATA OPND PROCESS-PARAMETERS	C308	01
IMMED DATA OPND FORMATTED-DOCUMENT-NAME	C309	01
IMMED DATA OPND MODIFY-DATA	C30B	41
IMMED DATA OPND FORMATTER-NAME	C30C	01
IMMED DATA OPND SIGN-ON-ID	C30D	01
IMMED DATA OPND SIGN-ON-ID	C30D	42
IMMED DATA OPND SOURCE/RECIPIENT-PASSWORD	C30E	01
IMMED DATA OPND CHARGE-CODE	C30F	01
IMMED DATA OPND RESERVED	C310	01
IMMED DATA OPND ORIGINATING-NODE-ADDRESS	C311	01
IMMED DATA OPND FUNCTION-SET	C312	01
IMMED DATA OPND PROCESS-PASSWORD	C313	01
IMMED DATA OPND CANCEL-ACTION	C317	01
IMMED DATA OPND LIST-ACTION	C318	01
IMMED DATA OPND TIME-LIMIT	C31A	01
IMMED DATA OPND SELECT-LIMIT	C31B	01
IMMED DATA OPND RETRIEVE-COUNT	C31C	01
IMMED DATA OPND DESCRIPTOR-CONTENT-DEFNTN	C31D	01
IMMED DATA OPND OBTAIN-OPTION	C31E	01
IMMED DATA OPND Assigned by SNADS	C31E	02

TABLE 3. IMMEDIATE DATA OPERAND ENCODINGS (Cont.)

CODE POINT DESCRIPTION	GDS ID	F
IMMEDIATE DATA OPERAND (IMMED DATA OPND)	C3	
IMMED DATA OPND SEARCH-REQUEST-NAME	C31F	01
IMMED DATA OPND IDENTIFIED-DATA	C320	02
IMMED DATA OPND IDENTIFIED-DATA	C320	03
IMMED DATA OPND IDENTIFIED-DATA	C320	41
IMMED DATA OPND IDENTIFIED-DATA	C320	42
IMMED DATA OPND CONTROL-VALUE	C321	01
IMMED DATA OPND EXCEPTION-CODE	C322	01
IMMED DATA OPND SOURCE-ADDRESS	C323	01
IMMED DATA OPND SOURCE-ADDRESS	C323	42
IMMED DATA OPND MESSAGE	C325	01
IMMED DATA OPND MESSAGE	C325	02
IMMED DATA OPND RECOVERY-ACTION	C327	01
IMMED DATA OPND CORRELATION	C328	01
IMMED DATA OPND DOCUMENT-TYPE	C329	01
IMMED DATA OPND GRAPHIC-CHARACTER-SET-ID	C32A	01
IMMED DATA OPND DOCUMENT-PASSWORD	C32E	01
IMMED DATA OPND DESTINATION-NODE-ADDRESS	C32F	01
IMMED DATA OPND LIBRARY-NAME	C330	01
IMMED DATA OPND LIBRARY-NAME	C330	41
IMMED DATA OPND FILE-OPTION	C331	01
IMMED DATA OPND SEARCH-OPTION	C332	01
IMMED DATA OPND SEARCH-ARGUMENTS	C333	01
IMMED DATA OPND SEARCH-DATA	C333	41
IMMED DATA OPND SEARCH-DATA	C333	42
IMMED DATA OPND FORMAT-PARAMETERS	C337	01
IMMED DATA OPND SIGN-ON-PASSWORD	C338	01
IMMED DATA OPND ACCESS-CODES	C339	01
IMMED DATA OPND STATUS-INFORMATION	C33D	01
IMMED DATA OPND COUNT	C33E	01
IMMED DATA OPND DISTRIBUTION-IDENTIFIER	C340	01

TABLE 3. IMMEDIATE DATA OPERAND ENCODINGS (Cont.)

CODE POINT DESCRIPTION	GDS ID	F
IMMEDIATE DATA OPERAND (IMMED DATA OPND)	C3	
IMMED DATA OPND DISTRIBUTION-NAME	C341	01
IMMED DATA OPND REPLY-DATA	C345	01
IMMED DATA OPND Assigned by SNADS	C350	01
IMMED DATA OPND Assigned by SNADS	C351	01
IMMED DATA OPND Assigned by SNADS	C352	01
IMMED DATA OPND Assigned by SNADS	C353	01
IMMED DATA OPND Assigned by SNADS	C354	01
IMMED DATA OPND Assigned by SNADS	C355	01
IMMED DATA OPND Assigned by SNADS	C356	01
IMMED DATA OPND Assigned by SNADS	C357	01
IMMED DATA OPND Assigned by SNADS	C358	01
IMMED DATA OPND Assigned by SNADS	C359	01
IMMED DATA OPND Assigned by SNADS	C360	01
IMMED DATA OPND Assigned by SNADS	C361	01
IMMED DATA OPND Assigned by SNADS	C362	01
IMMED DATA OPND Assigned by SNADS	C363	01

TABLE 4. DATA UNIT REFERENCE OPERAND ENCODINGS

The assignment of a codepoint to an operand that is a Data Unit reference should correspond to the codepoint that is assigned to the Data Unit being referenced; that is, an operand whose GDS ID is X'C43A' refers to a Data Unit whose GDS ID is X'C63A'.

CODE POINT DESCRIPTION	GDS ID	F
DATA UNIT REFERENCE OPND (DATA U REF OPND)	C4	
DATA U REF OPND SCAN-DATA	C43A	01
DATA U REF OPND BIT-STRING-REPRESENTATION	C43B	01

TABLE 5. DOCUMENT UNIT REFERENCE OPERAND ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DOCUMENT UNIT REF OPND (DOC U REF OPND)	C5	
DOC U REF OPND IDENTIFIED-DATA	C520	01

TABLE 6. PROFILE PARAMETERS

CODE POINT DESCRIPTION	GDS ID	F
PROFILE PARAMETERS	C7	
PROF PARM DOCUMENT NAME	C700	01
PROF PARM PROFILE GCID	C701	01
PROF PARM OWNER - Retired	C702	01
PROF PARM OWNER - Retired	C702	41
PROF PARM PRIVATE (5520)	C703	01
PROF PARM AUTHOR	C704	01
PROF PARM AUTHOR	C704	41
PROF PARM DOCUMENT GCID	C705	01
PROF PARM DOCUMENT TYPE	C706	01
PROF PARM CREATION DATE TIME	C707	01
PROF PARM LAST CHANGED DATE TIME	C708	01
PROF PARM COPY LIST	C709	41
PROF PARM FILE CABINET REFERENCE	C70A	01
PROF PARM SUBJECT	C70B	01
PROF PARM SUBJECT	C70B	41
PROF PARM SYSTEM CODE	C70C	01
PROF PARM DOCUMENT SIZE	C70D	01
PROF PARM FILE ID	C70E	01
PROF PARM LIBRARY ASSIGNED DOCUMENT NAME	C720	41
PROF PARM DOCUMENT CLASS	C721	01
PROF PARM FILE DATE TIME STAMP	C740	01
PROF PARM OWNERSHIP - Retired	C741	01
PROF PARM KEYWORDS	C742	41
PROF PARM EXPIRATION DATE	C744	01
PROF PARM OWNER DELEGATE - Retired	C745	41
PROF PARM REVISABLE-FORM-TEXT - Retired	C770	41

TABLE 7. DOCUMENT UNIT ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU DOCUMENT UNIT	C9	
DOCUMENT UNIT, INTERCHANGE	C903	x1
DOCUMENT UNIT, DOCUMENT-DESCRIPTOR-PARMS	C904	01
DOCUMENT UNIT, UNFORMATTED-RECIP'NT STATUS	C905	01
DOCUMENT UNIT, UNFORMATTED-SUMMARY STATUS	C906	01
DOCUMENT UNIT, UNFORMATTED-SOURCE STATUS	C907	01
DOCUMENT UNIT, SNADS DATA OBJECT	C908	01
DOCUMENT UNIT, SNADS RESTART DATA OBJECT	C909	01
DOCUMENT UNIT, SNADS DATA OBJECT PROFILE	C90A	01
NOTE: DOCUMENT UNIT CODES X'C980'-X'C9FF' USER ASSIGNED		

TABLE 8. DOCUMENT PROFILE ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU DOCUMENT PROFILE	CA	
DOCUMENT PROFILE, PRIVATE, 3730	CA01	01
DOCUMENT PROFILE, PRIVATE, DISOSS	CA01	02
DOCUMENT PROFILE, PRIVATE (5520)	CA02	01
DOCUMENT PROFILE, INTERCHANGE (IDPA)	CA03	01
BASE SUBPROFILE (IDPA)	CA04	01
ARCHITECTED APPLICATION SUBPROFILE (DIA)	CA05	01
IBM 3730 SUBPROFILE - 3730	CA70	01
DISOSS SUBPROFILE - DISOSS	CA71	01
IBM 5520 SUBPROFILE - 5520	CA72	01
PROFS SUBPROFILE - DISOSS	CA73	01
PROFS SUBPROFILE - DISOSS	CA73	02
DATA FILE SUBPROFILE	CA74	01
NOTE: DOCUMENT PROFILE CODES X'CA80'-X'CAFF' USER ASSIGNED		

TABLE 9. DOCUMENT CONTENT INTRODUCER ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU DOCUMENT CONTENT INTRODUCER	CB	
DOCUMENT CONTENT INTRODUCER, W/DOCUMENT	CB01	01
DOCUMENT CONTENT INTRODUCER, WO/DOCUMENT	CB02	01

TABLE 10. SYNCHRONOUS REPLY REQUIRED
COMMAND ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU COMMAND - SYNCHRONOUS REPLY RQRD (SRR)	CD	
SRR - FILE	CD02	x1
SRR - FILE	CD02	x2
SRR - DELETE	CD03	x1
SRR - RETRIEVE	CD04	x1
SRR - RETRIEVE	CD04	x4
SRR - Assigned by SNADS	CD05	x1
SRR - SEARCH	CD06	x1
SRR - FORMAT	CD0A	x1
SRR - EXECUTE	CD0B	x1
SRR - SIGN-ON	CD0C	x1
SRR - Assigned by SNADS	CD0F	x1
SRR - CANCEL-DISTRIBUTION	CD10	x1
SRR - MODIFY	CD12	x1
SRR - LIST	CD13	x1
SRR - OBTAIN	CD17	x1
SRR - Assigned by SNADS	CD17	x2
SRR - SET-CONTROL-VALUE	CD18	x1
SRR - DELIVER	CD19	x1
SRR - REQUEST-DISTRIBUTION	CD1C	x1
SRR - PROCESS-BIT-STRING	CD1D	x1
SRR - STATUS-LIST	CD1E	x1
SRR - SNADS QUERY-RESTART	CD23	x1

TABLE 11. DIU SUFFIX ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU SUFFIX	CF	
DIU SUFFIX, NORMAL TERMINATION	CF01	x0
DIU SUFFIX, ABNORMAL TERMINATION	CF02	x1

APPENDIX C. DIA DOCUMENT TYPES

The following table lists the document types registered by Document Interchange Architecture.

INTERCHANGE DATA STREAM TYPE IDENTIFIER
CODE

Reserved	X'0000'
Reserved	X'0001'
Final-Form-Text Document	X'0002'
5520 Revisable-Form-Text Document	X'0003'
Word-Processing EBCDIC	X'0004'
Word-Processing-Information-File (WPIF)	X'0005'
Image-Data-Subset Document	X'0006'
3730 Text Data Stream	X'0007'
DIA Document Library Document Descriptor Document	X'0008'
3732 Display Document Data stream	X'0009'
DIA Defined Document Unit Content	X'000A'
Revisable-Form-Text Document	X'000B'
1403 Printer Compatible Data Stream with Variable Length, Unblocked Records.	X'000C'
Digitized ADS Audio	X'000D'
IBM PC Data File	X'000E'
Reserved	X'000F' thru X'7FFF'

Figure 44. Document Type Code Assignments

APPENDIX D. GRAPHIC CHARACTER SETS

The DIU consists of five logical components: a prefix, a command sequence, optional data units, optional document units, and a suffix. Within these components, DIA specifies certain operands and parameters as having character values. DIA further characterizes character-valued operands and parameters as either ID Type, Name Type, Password Type, or Message Type.

The characters and character-to-value mappings used in the values of these operands and parameters are defined and registered in the IBM Corporate Specification C-H 3-3220-050, the Registry of Graphic Character Sets and Code Pages. That document assigns registered values to character sets and to mappings of these character sets to code pages of hexadecimal values. DIA processes use these registered identifiers to support connectivity and character information interchange.

DIA specifies that the CGCSGID (Coded Graphic Character Set Global Identifier) in effect at the beginning of each DIU component is 00337/00256 (character set 00337, code page 00256). This default CGCSGID may be temporarily overridden during the parsing or processing of some operands or parameters by using the LT self-defining data variable encoding method or the MESSAGE Format 2 operand.

Note: The shorter and more familiar acronym GCID was formerly used instead of CGCSGID.

In order to enhance connectivity and character data exchange, DIA associates each character type operand or parameter with rules to be applied by the sender of that operand or parameter. However, receivers must allow for the possibility that senders have not followed the DIA rules specified for character-valued operands or parameters. Therefore, receivers must be prepared to handle character data that do not conform to the DIA rules.

To maximize interconnectivity and the ability to exchange character data, DIA specifies that:

- All Format 1 Identification tokens (SIGN_ON_ID, ORIGINATING_NODE_ADDRESS, DESTINATION_NODE_ADDRESS, SOURCE_ADDRESS, and RECIPIENT_ADDRESS) are coded using the ID Type rules specified later in this section
- All Document Names and Format 42 Address tokens are coded using the Name Type rules specified later in this section
- Passwords are coded using the Password Type rules specified later in this section
- Messages are coded using the Message Type rules specified later in this section
- Tokens formed from other tokens, such as Distribution Document Names, use the rules which apply to their component parts.
- All DIU character values not mentioned explicitly in the list above are to be coded using the Name Type rules.

The DIA rules for each type are provided below.

- For ID Type, the DIA rules are:
 - Leading spaces are invalid (must be stripped from the token before placing it into the DIU).
 - Imbedded spaces are valid and significant.
 - Trailing spaces are not significant.
 - Values consisting of all spaces are invalid.
 - In their handling of ID Type values, DIA processes must treat upper and lower case letters, if they exist in the current character set, as equivalent. In addition, multiple forms of special characters (such as the Space and the Hyphen) must be treated as equivalent. For example, DESTINATION-NODE-ADDRESS operand values of "ROSE", "Rose", "rose", and "roSe" must all be treated as identifying the same OSN.

Note: To ensure connectivity in a SNADS environment, DIA requires OSN implementations to guarantee that all Node ID's and all User ID's use only CGCSGID 00930/00256. In addition, an OSN is required explicitly to uppercase lowercase alphabets, and to translate all special characters to a single form of that character before sending the Node ID's (which map to SNADS DGN's) and User ID's (which map to SNADS DEN's). During this operation, the OSN must detect characters that do not map to CGCSGID 00930/00256, and return appropriate exceptions or status. In the SNADS environment, end user nodes are effectively limited to the use of CGCSGID 00946/00256 (which is converted by the OSN to CGCSGID 00930/00256). The mapping table for converting CGCSGID 00946/00256 to CGCSGID 00930/00256 is provided in Appendix D, "Graphic Character Sets."

- For Name Type, the DIA rules are:
 - Leading spaces are invalid (must be stripped from the token before placing it into the DIU).
 - Imbedded spaces are valid and significant.
 - Trailing spaces are not significant.
 - Values consisting of all spaces are invalid.
 - Products may perform the same monocasing and translating operation described for ID Type values in handling Name Type values when such values are used for search type operations, directory lookups, and similar operations.
- For Password Type character values, the DIA rules are:
 - Leading spaces are valid and significant.
 - Imbedded spaces are valid and significant.
 - Trailing spaces are valid and significant.
 - Values consisting of all spaces are valid and significant.
 - No monocasing, translation, or other operation is to be performed on the values sent from the end user.

- For Message Type character values, the DIA rules are:
 - Leading spaces are valid and significant.
 - Imbedded spaces are valid and significant.
 - Trailing spaces are valid and significant if sent.
 - Values consisting of all space characters are valid and significant if sent.
 - No monocasing, translation, or other operation is to be performed on the values sent.

Note: Not all products can send all of the characters in a particular character set. Network administrators may wish to enhance interconnectivity or data exchangeability by imposing restrictions on the end-user interface to limit characters used for certain purposes to subsets of the characters permitted by the architecture.

Column		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
Row	Bit Pattern → ↓	00				01				10				11				
		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11	
0	0000					SP01	& SM03	- SP10						{ SM11	} SM14	\ SM07	0 ND10	
1	0001							/ SP12		a LA01	j LJ01	~ SD19		A LA02	J LJ02		1 ND01	
2	0010									b LB01	k LK01	s LS01		B LB02	K LK02	S LS02	2 ND02	
3	0011									c LC01	l LL01	t LT01		C LC02	L LL02	T LT02	3 ND03	
4	0100									d LD01	m LM01	u LU01		D LD02	M LM02	U LU02	4 ND04	
5	0101									e LE01	n LN01	v LV01		E LE02	N LN02	V LV02	5 ND05	
6	0110									f LF01	o LO01	w LW01		F LF02	O LO02	W LW02	6 ND06	
7	0111									g LG01	p LP01	x LX01		G LG02	P LP02	X LX02	7 ND07	
8	1000									h LH01	q LQ01	y LY01		H LH02	Q LQ02	Y LY02	8 ND08	
9	1001								\ SD13	i LI01	r LR01	z LZ01		I LI02	R LR02	Z LZ02	9 ND09	
A	1010					[SM06] SM08	 SM65	:									
B	1011					· SP11	\$ SC03	, SP08	#									
C	1100					< SA03	* SM04	% SM02	@									
D	1101					(SP06) SP07	— SP09	'									
E	1110					+ SA01	; SP14	> SA05	=									
F	1111					! SP02	^ SD15	? SP15	"									

Figure 45. Character Set 103 of Code Page 256

Column		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Row	Bit Pattern → ↓	00				01				10				11			
		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0	0000					SP01	& SM03	- SP10	∅ LO61	∅ LO62		μ SM17					0 ND10
1	0001					SP30	é LE11	/ SP12	É LE12	a LA01	j LJ01			A LA02	J LJ02	SP31	1 ND01
2	0010					â LA15	ê LE15	Â LA16	Ê LE16	b LB01	k LK01	s LS01		B LB02	K LK02	S LS02	2 ND02
3	0011					ä LA17	ë LE17	Ä LA18	Ë LE18	c LC01	l LL01	t LT01		C LC02	L LL02	T LT02	3 ND03
4	0100					à LA13	è LE13	À LA14	È LE14	d LD01	m LM01	u LU01		D LD02	M LM02	U LU02	4 ND04
5	0101					á LA11	í LI11	Á LA12	Í LI12	e LE01	n LN01	v LV01		E LE02	N LN02	V LV02	5 ND05
6	0110					ã LA19	î LI15	Ã LA20	Î LI16	f LF01	o LO01	w LW01		F LF02	O LO02	W LW02	6 ND06
7	0111					ä LA27	ï LI17	Ä LA28	Ï LI18	g LG01	p LP01	x LX01		G LG02	P LP02	X LX02	7 ND07
8	1000					ç LC41	ï LI13	Ç LC42	Ì LI14	h LH01	q LQ01	y LY01		H LH02	Q LQ02	Y LY02	8 ND08
9	1001					ñ LN19	β LS61	Ñ LN20		i LI01	r LR01	z LZ01		I LI02	R LR02	Z LZ02	9 ND09
A	1010										a SM21			- SP32			
B	1011					SP11	§ SC03	· SP08	# SM01		ø SM20			ô LO15	û LU15	ô LO16	û LU16
C	1100								@ SM05	ð LD63	æ LA51	ð LD62		ö LO17	ü LU17	ö LO18	ü LU18
D	1101								· SP05	ý LY11		Ý LY12		ò LO13	ù LU13	ò LO14	ù LU14
E	1110									þ LT63	Æ LA52	þ LT64		ó LO11	ú LU11	ó LO12	ú LU12
F	1111													õ LO19	ý LY17	õ LO20	

Figure 46. Character Set 946 of Code Page 256

Column		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Row	Bit Pattern → ↓	00				01				10				11			
		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0	0000					SP01	& SM03	- SP10		∅ LO62		μ SM17					0 ND10
1	0001							/ SP12	Ê LE12					A LA02	J LJ02		1 ND01
2	0010							Â LA16	Ê LE16					B LB02	K LK02	S LS02	2 ND02
3	0011							Ä LA18	Ë LE18					C LC02	L LL02	T LT02	3 ND03
4	0100							À LA14	É LE14					D LD02	M LM02	U LU02	4 ND04
5	0101							Á LA12	Í LI12					E LE02	N LN02	V LV02	5 ND05
6	0110							Ã LA20	Î LI16					F LF02	O LO02	W LW02	6 ND06
7	0111							Å LA28	Ï LI18					G LG02	P LP02	X LX02	7 ND07
8	1000							Ç LC42	ì LI14					H LH02	Q LQ02	Y LY02	8 ND08
9	1001						β LS61	Ñ LN20						I LI02	R LR02	Z LZ02	9 ND09
A	1010											a SM21					
B	1011					SP11	§ SC03	· SP08	# SM01			ø SM20				ô LO16	û LU16
C	1100								@ SM05			Ð LD62				ö LO18	ü LU18
D	1101								· SP05			Ý LY12				ò LO14	ù LU14
E	1110											Æ LA52	þ LT64			ó LO12	ú LU12
F	1111														ý LY17	õ LO20	

Figure 47. Character Set 930 of Code Page 256

Column		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Row	→ Bit Pattern ↓	00				01				10				11			
		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0	0000					SP01	& SM03	- SP10	∅ LO61	∅ LO62	° SM19	μ SM17	¢ SC04	{ SM11	}	\ SM07	0 ND10
1	0001					SP30	é LE11	/ SP12	Ê LE12	a LA01	j LJ01	~ SD19	£ SC02	À LA02	J LJ02		1 ND01
2	0010					â LA15	ê LE15	Â LA16	Ê LE16	b LB01	k LK01	s LS01	¥ SC05	B LB02	K LK02	S LS02	2 ND02
3	0011					ä LA17	ë LE17	Ä LA18	Ë LE18	c LC01	l LL01	t LT01	Pts SC06	C LC02	L LL02	T LT02	3 ND03
4	0100					à LA13	è LE13	À LA14	È LE14	d LD01	m LM01	u LU01	f SC07	D LD02	M LM02	U LU02	4 ND04
5	0101					á LA11	í LI11	Á LA12	Í LI12	e LE01	n LN01	v LV01	§ SM24	E LE02	N LN02	V LV02	5 ND05
6	0110					ã LA19	î LI15	Ã LA20	Î LI16	f LF01	o LO01	w LW01	¶ SM25	F LF02	O LO02	W LW02	6 ND06
7	0111					å LA27	ï LI17	Å LA28	Ï LI18	g LG01	p LP01	x LX01	¼ NF04	G LG02	P LP02	X LX02	7 ND07
8	1000					ç LC41	ì LI13	Ç LC42	Ì LI14	h LH01	q LQ01	y LY01	½ NF01	H LH02	Q LQ02	Y LY02	8 ND08
9	1001					ñ LN19	β LS61	Ñ LN20	\ SD13	i LI01	r LR01	z LZ01	¾ NF05	I LI02	R LR02	Z LZ02	9 ND09
A	1010					[SM06] SM08	 SM65	: SP13	« SP17	á SM21	í SP03	ˆ SM66	ˆ SP32	ˆ L161	ˆ NS02	ˆ NS03
B	1011					· SP11	§ SC03	, SP08	# SM01	» SP18	◊ SM20	¿ SP16	¡ SM13	ô LO15	û LU15	ô LO16	û LU16
C	1100					< SA03	* SM04	% SM02	@ SM05	ö LD63	æ LA51	ð LD62	- SM15	ö LO17	ü LU17	ö LO18	ü LU18
D	1101					(SPO6) SPO7	- SPO9	' SPO5	ÿ LY11	→ SD41	ÿ LY12	SD17	ò LO13	ù LU13	ò LO14	ù LU14
E	1110					+ SA01	; SP14	> SA05	= SA04	þ LT63	Æ LA52	þ LT64	' SD11	ó LO11	ù LU11	ó LO12	ù LU12
F	1111					! SP02	^ SD15	? SP15	" SP04	± SA02	⊗ SC01	® SM53	= SM10	õ LO19	ÿ LY17	õ LO20	SS99

Figure 48. Character Set 337 of Code Page 256

ID	Graphic	Description
LA01	a	a Small
LA02	A	A Capital
LA11	á	a Acute Small
LA12	Á	A Acute Capital
LA13	à	a Grave Small
LA14	À	A Grave Capital
LA15	â	A Circumflex Small
LA16	Â	A Circumflex Capital
LA17	ä	a Diaeresis Small
LA18	Ä	A Diaeresis Capital
LA19	ã	a Tilde Small
LA20	Ã	A Tilde Capital
LA27	â	a Overcircle Small
LA28	Â	A Overcircle Capital
LA51	æ	ae Diphthong Small
LA52	Æ	AE Diphthong Capital
LB01	b	b Small
LB02	B	B Capital
LC01	c	c Small
LC02	C	C Capital
LC41	ç	c Cedilla Small
LC42	Ç	C Cedilla Capital
LD01	d	d Small
LD02	D	D Capital
LD62	Ð	Eth Icelandic Capital
LD63	ð	eth Icelandic Small
LE01	e	e Small
LE02	E	E Capital
LE11	é	e Acute Small
LE12	É	E Acute Capital
LE13	è	e Grave Small
LE14	È	E Grave Capital
LE15	ê	e Circumflex Small
LE16	Ê	E Circumflex Capital
LE17	ë	e Diaeresis Small

ID	Graphic	Description
LE18	Ë	E Diaeresis Capital
LF01	f	f Small
LF02	F	F Capital
LG01	g	g Small
LG02	G	G Capital
LH01	h	h Small
LH02	H	H Capital
LI01	i	i Small
LI02	I	I Capital
LI11	í	i Acute Small
LI12	Í	I Acute Capital
LI13	ì	i Grave Small
LI14	Ì	I Grave Capital
LI15	î	i Circumflex Small
LI16	Î	I Circumflex Capital
LI17	ï	i Diaeresis Small
LI18	Ï	I Diaeresis Capital
LI61	ı	i Dotless Small
LJ01	j	j Small
LJ02	J	J Capital
LK01	k	k Small
LK02	K	K Capital
LL01	l	l Small
LL02	L	L Capital
LM01	m	m Small
LM02	M	M Capital
LN01	n	n Small
LN02	N	N Capital
LN19	ñ	n Tilde Small
LN20	Ñ	N Tilde Capital
LO01	o	o Small

Figure 49 (Part 1 of 3). Description of Code Page 256 Graphics.

ID	Graphic	Description
LO02	O	O Capital
LO11	ó	o Acute Small
LO12	Ó	O Acute Capital
LO13	ò	o Grave Small
LO14	Ò	O Grave Capital
LO15	ô	o Circumflex Small
LO16	Ô	O Circumflex Capital
LO17	ö	o Diaeresis Small
LO18	Ö	O Diaeresis Capital
LO19	õ	o Tilde Small
LO20	Õ	O Tilde Capital
LO61	ø	o Slash Small
LO62	Ø	O Slash Capital
LP01	p	p Small
LP02	P	P Capital
LQ01	q	q Small
LQ02	Q	Q Capital
LR01	r	r Small
LR02	R	R Capital
LS01	s	s Small
LS02	S	S Capital
LS61	ß	Sharp s Small
LT01	t	t Small
LT02	T	T Capital
LT63	þ	Thorn Icelandic Small
LT64	Þ	Thorn Icelandic Capital
LU01	u	u Small
LU02	U	U Capital
LU11	ú	u Acute Small

ID	Graphic	Description
LU12	Ú	U Acute Capital
LU13	ù	u Grave Small
LU14	Ù	U Grave Capital
LU15	û	u Circumflex Small
LU16	Û	U Circumflex Capital
LU17	ü	u Diaeresis Small
LU18	Ü	U Diaeresis Capital
LV01	v	v Small
LV02	V	V Capital
LW01	w	w Small
LW02	W	W Capital
LX01	x	x Small
LX02	X	X Capital
LY01	y	y Small
LY02	Y	Y Capital
LY11	ý	y Acute Small
LY12	Ý	Y Acute Capital
LY17	ÿ	y Diaeresis Small
LZ01	z	z Small
LZ02	Z	Z Capital
ND01	1	One
ND02	2	Two
ND03	3	Three
ND04	4	Four
ND05	5	Five
ND06	6	Six
ND07	7	Seven
ND08	8	Eight
ND09	9	Nine
ND10	0	Zero

Figure 49 (Part 2 of 3). Description of Code Page 256 Graphics.

ID	Graphic	Description
NF01	½	One Half
NF04	¼	One Quarter
NF05	¾	Three Quarters
NS02	²	Two Superscript
NS03	³	Three Superscript
SA01	+	Plus Sign
SA02	±	Plus or Minus Sign
SA03	<	Less Than Sign
SA04	=	Equal Sign
SA05	>	Greater Than Sign
SC01	¤	International Currency Symbol
SC02	£	Pound Sign
SC03	\$	Dollar Sign
SC04	¢	Cent Sign
SC05	¥	Yen Sign
SC06	Pts	Peseta Sign
SC07	f	Florin Sign, Guilder Sign
SD11	'	Acute Accent
SD13	`	Grave Accent
SD15	^	Circumflex Accent
SD17	¨	Diaeresis or Umlaut Accent,
SD19	~	Tilde Accent
SD41	¸	Cedilla or Sedila Accent
SM01	#	Number Sign
SM02	%	Percent Sign
SM03	&	Ampersand
SM04	*	Asterisk
SM05	@	At Sign
SM06	[Left Bracket
SM07	\	Backslash
SM08]	Right Bracket
SM10	==	Double Underscore
SM11	{	Left Brace

ID	Graphic	Description
SM13		Vertical Line Unbroken, Vertical Bar,
SM14	}	Right Brace
SM15	—	Overline
SM17	μ	Micro Symbol
SM19	°	Degree Symbol
SM20	♂	Ordinal Indicator, Masculine
SM21	♀	Ordinal Indicator, Feminine
SM24	§	Section Symbol (USA), Paragraph Symbol (Europe)
SM25	¶	Paragraph Symbol (USA)
SM53	®	Registered Trademark Symbol
SM65	¦	Vertical Line Broken
SM66	¬	Logical NOT, "End of Line" Symbol
SP01		Space
SP02	!	Exclamation Point
SP03	¡	Exclamation Point Inverted
SP04	"	Quotation Marks
SP05	'	Apostrophe
SP06	(Left Parenthesis
SP07)	Right Parenthesis
SP08	,	Comma
SP09	—	Underline, Continuous Underscore
SP10	-	Hyphen, Minus Sign
SP11	.	Period, Full Stop
SP12	/	Slash
SP13	:	Colon
SP14	;	Semicolon
SP15	?	Question Mark
SP16	¿	Question Mark Inverted
SP17	«	Left Angle Quotes
SP18	»	Right Angle Quotes
SP30		Required Space
SP31		Numeric Space
SP32		Syllable Hyphen
SS99		Eight Ones

Figure 49 (Part 3 of 3). Description of Code Page 256 Graphics.

APPENDIX E. CHARACTER SET TRANSLATION

To convert CGCSGID 00946/00256 to CGCSGID 00930/00256, the tables shown in Figure 50 or their equivalents, are to be used. Wherever the conversion results in a byte with a value of X'00', the corresponding input value was not a character in Character Set 00946, and therefore could not be converted to a character in Character Set 00930.

Byte(s)	Content	Byte(s)	Content
X'00' thru X'3F'	X'00'	X'50'	X'50'
X'40'	X'40'	X'51'	X'71'
X'41'	X'40'	X'52'	X'72'
X'42'	X'62'	X'53'	X'73'
X'43'	X'63'	X'54'	X'74'
X'44'	X'64'	X'55'	X'75'
X'45'	X'65'	X'56'	X'76'
X'46'	X'66'	X'57'	X'77'
X'47'	X'67'	X'58'	X'78'
X'48'	X'68'	X'59'	X'59'
X'49'	X'69'	X'5A'	X'00'
X'4A'	X'00'	X'5B'	X'5B'
X'4B'	X'4B'	X'5C' thru X'5F'	X'00'
X'4C' thru X'4F'	X'00'		

Figure 50 (Part 1 of 6). Conversion Table (00946 to 00930)

Byte(s)	Content
X'60'	X'60'
X'61'	X'61'
X'62'	X'62'
X'63'	X'63'
X'64'	X'64'
X'65'	X'65'
X'66'	X'66'
X'67'	X'67'
X'68'	X'68'
X'69'	X'69'
X'6A'	X'00'
X'6B'	X'6B'
X'6C' thru X'6F'	X'00'

Byte(s)	Content
X'70'	X'80'
X'71'	X'71'
X'72'	X'72'
X'73'	X'73'
X'74'	X'74'
X'75'	X'75'
X'76'	X'76'
X'77'	X'77'
X'78'	X'78'
X'79'	X'00'
X'7A'	X'00'
X'7B'	X'7B'
X'7C'	X'7C'
X'7D'	X'7D'
X'7E' and X'7F'	X'00'

Figure 50 (Part 2 of 6). Conversion Table (00946 to 00930)

Byte(s)	Content
X'80'	X'80'
X'81'	X'C1'
X'82'	X'C2'
X'83'	X'C3'
X'84'	X'C4'
X'85'	X'C5'
X'86'	X'C6'
X'87'	X'C7'
X'88'	X'C8'
X'89'	X'C9'
X'8A' and X'8B'	X'00'
X'8C'	X'AC'
X'8D'	X'AD'
X'8E'	X'AE'
X'8F'	X'00'

Byte(s)	Content
X'90'	X'00'
X'91'	X'D1'
X'92'	X'D2'
X'93'	X'D3'
X'94'	X'D4'
X'95'	X'D5'
X'96'	X'D6'
X'97'	X'D7'
X'98'	X'D8'
X'99'	X'D9'
X'9A'	X'9A'
X'9B'	X'9B'
X'9C'	X'9E'
X'9D'	X'00'
X'9E'	X'9E'
X'9F'	X'00'

Figure 50 (Part 3 of 6). Conversion Table (00946 to 00930)

Byte(s)	Content
X'A0'	X'A0'
X'A1'	X'00'
X'A2'	X'E2'
X'A3'	X'E3'
X'A4'	X'E4'
X'A5'	X'E5'
X'A6'	X'E6'
X'A7'	X'E7'
X'A8'	X'E8'
X'A9'	X'E9'
X'AA' and X'AB'	X'00'
X'AC'	X'AC'
X'AD'	X'AD'
X'AE'	X'AE'
X'AF'	X'00'

Byte(s)	Content
X'B0' thru X'BF'	X'00'
X'C0'	X'00'
X'C1'	X'C1'
X'C2'	X'C2'
X'C3'	X'C3'
X'C4'	X'C4'
X'C5'	X'C5'
X'C6'	X'C6'
X'C7'	X'C7'
X'C8'	X'C8'
X'C9'	X'C9'
X'CA'	X'60'
X'CB'	X'EB'
X'CC'	X'EC'
X'CD'	X'ED'
X'CE'	X'EE'
X'CF'	X'EF'

Figure 50 (Part 4 of 6). Conversion Table (00946 to 00930)

Byte(s)	Content
X'D0'	X'00'
X'D1'	X'D1'
X'D2'	X'D2'
X'D3'	X'D3'
X'D4'	X'D4'
X'D5'	X'D5'
X'D6'	X'D6'
X'D7'	X'D7'
X'D8'	X'D8'
X'D9'	X'D9'
X'DA'	X'00'
X'DB'	X'FB'
X'DC'	X'FC'
X'DD'	X'FD'
X'DE'	X'FE'
X'DF'	X'DF'

Byte(s)	Content
X'E0'	X'00'
X'E1'	X'40'
X'E2'	X'E2'
X'E3'	X'E3'
X'E4'	X'E4'
X'E5'	X'E5'
X'E6'	X'E6'
X'E7'	X'E7'
X'E8'	X'E8'
X'E9'	X'E9'
X'EA'	X'00'
X'EB'	X'EB'
X'EC'	X'EC'
X'ED'	X'ED'
X'EE'	X'EE'
X'EF'	X'EF'

Figure 50 (Part 5 of 6). Conversion Table (00946 to 00930)

Byte(s)	Content
X'F0'	X'F0'
X'F1'	X'F1'
X'F2'	X'F2'
X'F3'	X'F3'
X'F4'	X'F4'
X'F5'	X'F5'
X'F6'	X'F6'
X'F7'	X'F7'
X'F8'	X'F8'
X'F9'	X'F9'
X'FA'	X'00'
X'FB'	X'FB'
X'FC'	X'FC'
X'FD'	X'FD'
X'FE'	X'FE'
X'FF'	X'00'

Figure 50 (Part 6 of 6). Conversion Table (00946 to 00930)

APPENDIX F. DIU GENERAL EXCEPTION CONDITIONS

This section defines general exception conditions common to most the DIA commands. Exception conditions that are specific to each of the commands are described in the detailed command descriptions.

The exception condition encoding for severity, class, condition, and the exception condition object must be specified by each implementation exactly as defined. This is necessary to ensure that interpretation of an exception condition is consistent among the various product implementations.

The encoding scheme for exception conditions permits a very large number of exception condition codes. To identify the DIU element in which the exception condition is detected, it is frequently necessary to have the LLIDF of the element. When the DIU element is specified more than once in the same DIU, it is sometimes necessary to examine the data to determine the problem. When the exception condition code by itself reports an ambiguous exception condition that does not precisely identify the faulty element, the LLIDF and the data where the exception condition is detected are required to diagnose the problem. The sender of the exception condition is not required to return the LLIDF and the data. If the LLIDF and data are returned, they must be returned as specified in the exception condition data field defined for each of the exception conditions.

PREFIX EXCEPTION CONDITIONS

The following list defines the general prefix exception conditions.

- The DIU is not processed if the prefix is not valid.
Exception = Catastrophic, Syntax, Data-Not-Supported, DIU-Prefix
Exception Code = X'C20201'
Exception data = LLIDF and data of DIU Prefix
- A DIU received that contains an invalid prefix format.
Exception = Catastrophic, Syntax, Format-Invalid, DIU-Prefix
Exception Code = X'C20E01'
Exception data = LLIDF of DIU Prefix
- A DIU Prefix ID received that is invalid.
Exception = Catastrophic, Syntax, ID-Invalid, DIU-Prefix
Exception Code = X'C20C01'
Exception data = LLIDF and data of invalid Prefix ID
- A DIU Prefix received that has an invalid length.
Exception = Catastrophic, Syntax, Length-Invalid, DIU-Prefix
Exception Code = X'C20F01'
Exception data = LLIDF of invalid DIU Prefix
- Segmentation indicated for a DIU Prefix
Exception = Catastrophic, Syntax, Segmentation, DIU-Prefix
Exception Code = X'C20801'
Exception data = LLIDF and introducer extension of invalid Prefix

COMMAND EXCEPTION CONDITIONS

The following list defines the general command exception conditions.

- A required operand is not specified in the command.

Exception = Catastrophic, Syntax, Data-Not-Found, Command-Operand
Exception Code = X'C20708'
Exception data = LLIDF of the required operand
- A command operand detected that is not supported.

Exception = Warning, Syntax, Function-Not-Supported, Command-Operand
Exception Code = X'420108'
Exception data = LLIDF of the unsupported operand.
- A command operand is not present in the required sequence.

Exception = Catastrophic, Syntax, Sequence, Command-Operand
Exception Code = X'C20A08'
Exception data = LLIDF of operand that is out of sequence
- A command detected that is not supported.

Exception = Catastrophic, Syntax, Function-Not-Supported, Command
Exception Code = X'C20107'
Exception data = LLIDF of command that is not supported
- The command is not processed if the command sequence introducer extension is not valid.

Exception = Catastrophic, Syntax, Indicator-Invalid, Command
Exception Code = X'C21007'
Exception data = LLIDF and introducer extension of command
- The command is not processed if a required operand is specified more than once when only one occurrence is permitted.

Exception = Catastrophic, Syntax, Function-Not-Supported, Command-Operand
Exception Code = X'C20108'
Exception data = LLIDF and data of command operand
- A command with multiple occurrences of the same optional operand that permits only one occurrence will be processed using the first operand occurrence, but will send a warning acknowledgement to the requester.

Exception = Warning, Syntax, Function-Not-Supported, Command-Operand
Exception Code = X'420108'
Exception data = LLIDF and data of the command operand

- A command received is an invalid format.
 - Exception = Catastrophic, Syntax, Format-Invalid, Command
 - Exception Code = X'C20E07'
 - Exception data = LLIDF of invalid command format
- A command operand received is an invalid format.
 - Exception = Catastrophic, Syntax, Format-Invalid, Command-Operand
 - Exception Code = X'C20E08'
 - Exception data = LLIDF of invalid operand format
- A command operand received is an invalid length.
 - Exception = Catastrophic, Syntax, Length-Invalid, Operand-Value
 - Exception Code = X'C20F09'
 - Exception data = LLIDF of invalid operand format
- Command Sequence Introducer extension contains an invalid sequence number.
 - Exception = Catastrophic, Syntax, Sequence, Command
 - Exception Code = X'C20A07'
 - Exception data = LLIDF and introducer extension of command
- A command operand detected that contains an invalid authorization value.
 - Exception = Catastrophic, Semantic, Unauthorized-Access, Operand-Value
 - Exception Code = X'C30309'
 - Exception data = LLIDF of operand containing the authorization value
- A command detected that is not permitted in the active function set.
 - Exception = Catastrophic, Semantic, Function-Not-Supported, Command
 - Exception Code = X'C30107'
 - Exception data = LLIDF of the requested command

- A command operand value detected is not supported.
 - Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
 - Exception Code = X'C30209'
 - Exception data = LLIDF and data of command operand

- A new command request has been received before an outstanding SRR command/reply sequence has been concluded.
 - Exception = Catastrophic, Process, Resource-Not-Available, Command
 - Exception Code = X'C40407'
 - Exception data = LLIDF of command being terminated

- An abnormal process condition has occurred that terminates command execution.
 - Exception = Catastrophic, Process, Execution-Terminated, Command
 - Exception Code = X'C40607'
 - Exception data = LLIDF of command being terminated

- A resource required for a DIA command process is not available.
 - Exception = Catastrophic, Process, Resource-Not-Available, Command
 - Exception Code = X'C40407'
 - Exception data = LLIDF of command being terminated

- An unrecoverable I/O error terminates command processing.
 - Exception = Catastrophic, Process, I/O-Error, Command
 - Exception data = X'C40B07'
 - Exception data = LLIDF of command being terminated

DOCUMENT UNIT EXCEPTION CONDITIONS

The following list defines the general document unit exception conditions.

- The DIU is not processed if the document unit introducer extension is not valid.

Exception = Catastrophic, Syntax, Indicator-Invalid, Document-Unit
Exception Code = X'C2100C'
Exception data = LLIDF of document unit introducer and extension

- A specified Document Unit not found.

Exception = Catastrophic, Syntax, Data-Not-Found, Document-Unit
Exception Code = X'C2070C'

- Document Unit Introducer extension contains a sequence number which is invalid.

Exception = Catastrophic, Syntax, Sequence, Document-Unit
Exception Code = X'C20A0C'
Exception data = LLIDF and introducer extension of document unit

DATA UNIT EXCEPTION CONDITIONS

The following list defines the general data unit exception conditions.

- A specified Data Unit not found.

Exception = Catastrophic, Syntax, Data-Not-Found, Data-Unit
Exception Code = X'C2070A'

- Data Unit Introducer extension contains a sequence number which is invalid.

Exception = Catastrophic, Syntax, Sequence, Data-Unit
Exception Code = X'C20A0A'
Exception data = LLIDF and introducer extension of data unit

SUFFIX EXCEPTION CONDITIONS

The following list defines the general suffix exception conditions.

- The DIU is not processed if the suffix is not valid.
Exception = Catastrophic, Syntax, Data-Not-Supported, DIU-Suffix
Exception Code = X'C20213'
Exception data = LLIDF and data of a suffix
- A DIU suffix ID received that is invalid.
Exception = Catastrophic, Syntax, ID-Invalid, DIU-Suffix
Exception Code = X'C20C13'
Exception data = LLIDF of invalid suffix ID
- A DIU suffix received that has an invalid format.
Exception = Catastrophic, Syntax, Format-Invalid, DIU-Suffix
Exception Code = X'C20E13'
Exception data = LLIDF of invalid DIU suffix
- A DIU suffix received that has an invalid length.
Exception = Catastrophic, Syntax, Length-Invalid, DIU-Suffix
Exception Code = X'C20F13'
Exception data = LLIDF of invalid DIU suffix
- Segmentation indicated for a DIU suffix.
Exception = Catastrophic, Syntax, Segmentation, DIU-Suffix
Exception Code = X'C20813'
Exception data = LLIDF and introducer extension of invalid suffix

PROCESS EXCEPTION CONDITIONS

The following list defines the general process exception conditions.

- A user process exception has been detected. The severity of the exception condition is specified in the Exception Condition Class field: catastrophic, severe, warning or information. The specific user exception condition is specified as an undifferentiated bit string in the Exception Data field.
Exception = severity, Process, User-Specified-Condition, User-Specified-Data
Exception Code = X'*40018'
Exception data = Undifferentiated bit string.

GLOSSARY

access code. A 4-byte decimal value, assigned to a document by the primary owner, that determines the set of users allowed to access the document.

address. (1) A character or group of characters that identifies a register, a particular part of storage, or some other data source or destination. (2) In DIA, a 1- to 8-byte character string that identifies the logical components of an office system network. These logical components are: source nodes, recipient nodes, and office system nodes.

affinity. A defined relationship that permits the DIA resources of a source or recipient to be accessed on his behalf by another user.

application processing services. The set of services that provide DIA functions enabling users to access processing capabilities of a remote node.

ARR. Asynchronous reply required.

asynchronous reply required (ARR). A command class that requests asynchronous processing and reply of a DIA function.

CGCSGID. Coded graphic character set global identifier.

COD. Confirmation-of-delivery.

code. A system of bit-patterns to which specific graphic or control meanings have been assigned.

coded graphic character. A graphic character, with its assigned code point.

coded graphic character set. A set of graphic characters, with their assigned code points.

coded graphic character set global identifier (CGCSGID). A 10-digit decimal identifier consisting of the concatenation of a graphic character set global identifier (GCSGID) and a code page global identifier (CPGID).

code page. A specification of code points for each graphic character in a set, or in a collection of graphic character sets. Within a given code page, a code point can have one and only one specific meaning.

code page global identifier (CPGID). A 5-digit decimal identifier assigned to a code page.

code point. One of the bit-patterns specified by a code.

command. The function to be performed by the receiving DIA process.

command sequence. A DIU data stream component containing a set of one or more commands.

communication subsystem path. A logical communication line between two users or user processes. In SNA terms, each user identifies the path by means of LUNAME(MODENAME).

condition code. Defines the specific exception condition detected by the receiver of a DIU.

confirmation-of-delivery (COD). An asynchronous message returned to the source node of a distribution request that indicates the information distributed has been delivered to the recipient node.

control variable. A DIA entity maintained by a DIA process for the

purpose of verification and authorization.

correlation value. Information used to uniquely identify and correlate the request to the reply.

data unit. A DIU data stream component that contains information referenced by operands of a command in the DIU.

data variable. A variable length collection of information contained in a structured field.

destination node. The office system node that provides services for attached source and recipient nodes.

DIA. Document interchange architecture.

DIA session. A logical connection between two DIA processes that is used to exchange information.

distribution. In general, the function provided by DIA of transporting information from a source node to one or more recipient nodes.

distribution document name. A unique identifier assigned to each distribution request.

distribution library. The collection of distribution queues and data storage provided by an office system node for the purpose of document distribution.

distribution queue. A queue of distribution and status information to be delivered to source or recipient nodes.

distribution system. The collection of office system nodes, source nodes, and recipient nodes that are interconnected to form an office system network.

DIU component. A self-defining, variable length structured field. The

DIU components are: prefix, command sequence, data unit, document unit, and suffix.

DIU subcomponent. A self-defining, variable length structured field contained within a DIU component.

DIU. Document interchange unit.

document. (1) (ISO) A data medium and the data recorded on it, that generally has permanence, and that can be read by man or machine. (2) A unified collection of information pertaining to a specific subject or related subjects.

document class. A 1- to 16-byte character string that the user designates as the class of the document being filed in the Document Library.

Document Content Architecture. A family of data stream architectures that define and specify the form of the information by describing the syntax and semantics of allowable elements in the data stream.

document content introducer. The DIU data stream subcomponent that identifies the beginning of the document content.

document descriptor. A set of profile parameters describing a document that satisfied a document library search request.

document descriptor document. A collection of one or more document descriptors.

document distribution services. The set of services that provide DIA functions enabling users to distribute information in distribution system.

Document Interchange Architecture (DIA). The specification of rules and data streams necessary to interchange information in a consistent, predictable manner.

document interchange unit (DIU). The basic unit of information exchanged between DIA processes.

document library. A repository on which documents and document related information is stored.

document library services. The set of services that provide DIA functions enabling users to manage the contents of a document library.

document type. A classification that identifies the structure and format of a document.

document unit. A DIU data stream component that contains the document and related document information.

document unit identifier. The DIU data stream subcomponent that contains the document type and system code identifier of the document.

end user. (1) The ultimate source or destination of information flowing through a system. (2) In DIA, a program, device, person, or system that uses DIA for the purpose of information interchange.

exception condition class. The type of exception condition detected by the receiver of a DIU. The exception classes are: session, syntax, semantic, process, and sender.

exception condition data. A field containing the DIU data stream component or subcomponent that caused the exception condition.

exception condition object. An identifier of the DIU component or subcomponent that caused the exception condition.

final-form text. Text that has already been formatted and is ready for presentation.

format byte. That part of the structure field introducer that defines the format and content of the structured field data variable.

function set. The set of commands that identify the scope of work. Function sets have been defined so that each set contains all commands required for a well-defined, usable, and complete set of functions for a given category of services.

GCID. Graphic character set ID.

graphic character set ID (GCID). The registry for graphic character sets and code pages. See CGCSGID.

ID. That part of the structure field introducer that defines the class and type of the structured field.

IDP. Interchange document profile.

ISS. Introducer extension.

Interchange Document Profile (IDP). A set of descriptors that identify and describe a document.

introducer. A 5-byte structured field identifier. The introducer contains a 2-byte length field, a 2-byte ID, and a format byte.

introducer extension. An optional extension to the structured field introducer used for segmentation of the structured field.

LADN. Library assigned document name.

library assigned document name (LADN). A unique name assigned to documents filed in the document library.

LUNAME. An SNA acronym for Logical Unit Name.

message. A collection of information transmitted from one point to another.

MODENAME. An SNA term referring to a particular path name supported within a SNA Logical Unit.

NRR. No reply required.

No reply required (NRR). A command class used when the function requested does not require a reply.

office system node. The DIA process that provides the services for attached source or recipient nodes.

operand. (1) (ISO) An entity to which an operation is applied. (2) A data stream subcomponent that controls the execution of the command.

originating node. The office system node that provides services for attached source nodes.

OSN. Office system node.

owner-delegate. A user that is designation by the primary owner as secondary owner of the document in the document library.

password. A characters string used for validation and authorization to gain access to a resource.

personal. A distribution class of service that requires the recipient to supply a password to receive the distributed information.

prefix. The DIU data stream component that introduces and identifies the DIU.

primary owner. The user who files the document in the document library.

priority. A distribution class of service that prioritizes the distributions so information of higher priority is delivered before information of lower priority.

process. (1) A systematic sequence of operations to produce a specified

result. (2) In DIA, a program that uses the DIA rules and data structures to interchange information.

profile parameter. A field of a subprofile that identifies and describes the document.

recipient. An end user that receives information in an office system network.

recipient node. A DIA logical component that provides services on behalf of recipients.

recovery action. The procedure recommended by the process that detected an exception condition.

reply. A command that is used to respond to a previously received request.

request. A command that specifies a function to be performed.

search argument. A search selection criteria that contains the profile parameter identifier, the search data value, and the search comparison operator.

search data parameter set. A collection of one or more search data parameters and the logical operators used to relate them.

search result list. A user named object that contains references to documents selected by the SEARCH command process.

segmentation. The division of a DIU data stream component into two or more segments.

SNA. Systems network architecture

SNADS. Systems network architecture distribution service

source. An end user that request services in an office system network.

source node. A DIA logical component that provides services on behalf of sources.

SRR. Synchronous reply required.

structured field. A self-defining, variable length field comprised of an introducer, an optional introducer extension, and a data variable.

subprofile. A set of profile parameters that describe the characteristics and attributes of a document.

suffix. The DIU data stream component that terminates the DIU.

synchronous reply required (SRR). A command class that requests synchronous processing and reply of a DIA function.

system code. An identifier associated with the originator of the document that is contained in a DIU document unit.

systems network architecture. The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through and controlling the configuration and operation of networks.

systems network architecture distribution service. An asynchronous communication service that distributes objects from, through, and to queues in distribution service units over a network of distribution connections.

user. See end user.

INDEX

A

access codes 16
ACCESS-CODE (Format 41) 209
address operands 27
address tokens 25
affinity
 affinity list 17
 affinity relationships 17
 agent 17
 principal 17
 temporary agent 17
 authorization list 17
affinity list 17
agent 17
Application Process Services 187
 DELIVER 188
 CORRELATION 188
 exception conditions 190
 IDENTIFIED-DATA 188
 request/reply protocol 189
EXECUTE 192
 exception conditions 195
 IDENTIFIED-DATA 192
 ORIGINATING-NODE-ADDRESS 193
 PROCESS-NAME 192
 PROCESS-PARAMETERS 194
 PROCESS-PASSWORD 194
 request/reply protocol 194
 SOURCE-ADDRESS 193
 SOURCE-PASSWORD 193
FORMAT 197
 exception conditions 201
 FORMAT-PARAMETERS (Format 1) 198
 FORMATTED-DOCUMENT-NAME (Format 1) 198
 FORMATTER-NAME (Format 1) 198
 IDENTIFIED-DATA 197
 ORIGINATING-NODE-ADDRESS (Format 1) 198
 request/reply protocol 199
 SOURCE-ADDRESS 198
 SOURCE-PASSWORD (Format 1) 198
MODIFY 203

exceptions conditions 207
IDENTIFIED-DATA 203
MODIFY-DATA (Format 41) 204
ORIGINATING-NODE-ADDRESS 205
request/reply protocol 206
SOURCE-ADDRESS 205
SOURCE-PASSWORD 205
Application Processing Services 11, 32
 office system node functions 33
 source node functions 33
architecture, definition 1
asynchronous reply required command
 class (ARR) 37
ATTRIBUTE-LIST (Format 1) 210
authorization list 17

B

BIT STRING-REPRESENTATION (Format 1) 212

C

CANCEL-ACTION (Format 1) 213
CHARGE-CODE (Format 1) 214
Code Points 267
command classes 37
 asynchronous reply required command
 class (ARR) 37
 no reply required command class
 (NRR) 37
 synchronous reply required command
 class (SRR) 37
command sequence 47
condition code 63
confirmation-of-delivery (COD) 23
CONTROL-VALUE (Format 1) 214
CORRELATION (Format 1) 216
COUNT (Format 1) 217

D

DESCRIPTOR-CONTENT-DEFINITION (Format 41) 217

DESTINATION-NODE-ADDRESS (Format 1) 219

distribution addressing 25

- address tokens 25
- authorization value 26
- domain 26
- global name 26
- name 26

distribution document name 23

DISTRIBUTION-IDENTIFIER (Format 1) 220

DISTRIBUTION-NAME (Format 1) 221

document 2

document access

- private documents 16
- public documents 16
- shared-private documents 16

document class 20

document content introducer 53

- document content introducer type 1 53
- document content introducer type 2 53

document descriptor document 21

document descriptors 21

Document Distribution Services 11, 23

- CANCEL-DISTRIBUTION 137
- CANCEL-ACTION 138
- DOCUMENT-PASSWORD 138
- exception conditions 141
- RECIPIENT-ADDRESS 139
- RECIPIENT-PASSWORD 139
- request/reply protocol 140
- SOURCE-ADDRESS 139
- SOURCE-PASSWORD 139

DELIVER 144

- ATTRIBUTE-LIST 145
- CORRELATION 145
- DISTRIBUTION-ID 145
- exception conditions 148
- IDENTIFIED-DATA 144
- MESSAGE 145
- ORIGINATING-NODE-ADDRESS 145
- RECIPIENT-ADDRESS 145
- request/reply protocol 146
- SOURCE-ADDRESS 144
- support considerations 150

LIST 151

- exception conditions 155
- LIST-ACTION 152

- RECIPIENT-ADDRESS 153
- RECIPIENT-PASSWORD 153
- request/reply protocol 154

OBTAIN 165

- DOCUMENT-PASSWORD 167
- exception conditions 170
- OBTAIN-OPTION (Format 1) 166
- RECIPIENT-ADDRESS 167
- RECIPIENT-PASSWORD 168
- request/reply protocol 168
- office system node functions 24

PROCESS-BIT-STRING (PBS) 173

- BIT-STRING-REPRESENTATION 174
- exception conditions 176
- IDENTIFIED-DATA 174
- request/reply protocol 175
- SCAN-DATA 174
- recipient node functions 24

REQUEST-DISTRIBUTION 177

- ATTRIBUTE-LIST 179
- DESTINATION-NODE-ADDRESS 178
- DISTRIBUTION-NAME 180
- exception conditions 182
- IDENTIFIED-DATA 178
- MESSAGE 180
- operand-syntax 180
- RECIPIENT-ADDRESS 179
- request/reply protocol 181
- SOURCE-ADDRESS 179
- SOURCE-PASSWORD 179
- support considerations 184
- source node functions 23

STATUS-LIST 185

- exception conditions 186
- request/reply protocol 186
- STATUS-INFORMATION 185

Document Interchange Architecture

- application services
- Application Processing Services 32
- Document Distribution Services 23
- Document Library Services 18
- File Transfer Services 32
- source node functions 33
- definition iii
- information interchange base iii
- process 6
 - local 6
 - remote 6
- session 6
- document interchange unit 41
 - command sequence 41
 - data unit 41
 - DIU identifier 46

- document content introducer 50
- document profile 50, 52
 - introducer 52
 - subprofiles 52
- document unit 41
- document unit identifier 51
 - document type 51
 - system code 51
- identifier 50
- introducer 42, 46
- prefix 41
- suffix 41
 - Type 1 suffix 54
 - Type 2 suffix 54
- Document Library Services 11
- FILE 99
 - ACCESS-CODE 100
 - exception conditions 102
 - IDENTIFIED-DATA 100
 - ORIGINATING-NODE-ADDRESS 100
 - SOURCE-ADDRESS 100
 - SOURCE-PASSWORD 100
- office system node functions 19
- RETRIEVE 104
 - DESCRIPTOR-CONTENT-DEFINITION 106, 115
 - exception conditions 111, 117
 - IDENTIFIED-DATA 104
 - ORIGINATING-NODE-ADDRESS 105, 114
 - RETRIEVE-COUNT 105, 115
 - SEARCH-DATA 114
 - SEARCH-OPTION 115
 - SELECT-LIMIT 115
 - SOURCE-ADDRESS 105, 114
 - SOURCE-PASSWORD 105, 114
 - support considerations 112
 - TIME-LIMIT 115
- SEARCH 113
 - SEARCH-REQUEST-NAME 114
- source node functions 19
- document ownership
 - owner-delegate 16
 - primary owner 15
- document profile 52
- Document Types 277
- document unit identifier 51
- document units 50
- DOCUMENT-PASSWORD (Format 1) 221
- DOCUMENT-TYPE (Format 1) 222

E

- exception condition classes 62
 - condition code 63
 - exception condition data 63
 - exception condition object 63
 - no-exception 62
 - process 62
 - semantic 62
 - sender 62
 - session 62
 - severity indicator 62
 - catastrophic 62
 - information 62
 - severe 62
 - warning 62
 - syntax 62
- EXCEPTION-CODE (Format 1) 223

F

- File Transfer Services 11, 32, 119
 - DELIVER 120
 - CORRELATION 120
 - exception conditions 122
 - IDENTIFIED-DATA 120
 - Request/Reply Protocol 121
 - FILE (Format 2) 124
 - exception conditions 126
 - FILE-OPTION 124
 - IDENTIFIED-DATA 124
 - LIBRARY-NAME 124
 - Request/Reply Protocol 125
 - requesting node functions 32
 - RETRIEVE (Format 4) 127
 - IDENTIFIED-DATA 127
 - LIBRARY-NAME 127
 - Request/Reply Protocol 128
 - support considerations 131
 - server node functions 32
 - FILE-OPTION (Format 1) 224
 - FORMAT-PARAMETERS (Format 1) 225
 - FORMATTED-DOCUMENT-NAME (Format 1) 225
 - FORMATTER-NAME (Format 1) 225
 - function set 10 14
 - function set 11 14
 - function set 2 14
 - function set 3 14
 - function set 4 14

function set 5 14
function set 6 14
function set 7 14
function set 8 14, 91
function set 9 14
FUNCTION-SET (Format 1) 226

G

GRAPHIC-CHARACTER-SET-ID (Format 1) 227

I

IDENTIFIED-DATA (Format 1) 228
IDENTIFIED-DATA (Format 2) 228
IDENTIFIED-DATA (Format 3) 229
IDENTIFIED-DATA (Format 41) 230
IDENTIFIED-DATA (Format 42) 231
introducer 42
 format byte 43
 introducer extension (ISS) 44
 sequence number 44
 structured-field identifier 42
 structured-field length 42
introducer extension (ISS) 44
 indicator byte 44

L

LADN
 See Library-Assigned Document Name
Library-Assigned Document Name 19
LIBRARY-NAME (Format 1) 232
LIBRARY-NAME (Format 41) 232
LIST-ACTION (Format 1) 233

M

MESSAGE (Format 1) 235
MESSAGE (Format 2) 236
MODIFY-DATA (Format 41) 236

N

no reply required command class
 (NRR) 37
node addresses
 recipient address 8
 source address 8
 uniqueness 8
nodes
 office system node 7
 recipient node 7
 source node 7

O

OBTAIN-OPTION (Format 1) 239
office information architectures 1
 list of architectures 1
office system network 1
 definition 1
office system node 7
 destination node 8
 originating node 8
operand class 48
 data unit reference 48
 document unit reference 48
 immediate data 48
operands
 ACCESS-CODE (Format 41) 209
 ATTRIBUTE-LIST (Format 1) 210
 BIT STRING-REPRESENTATION (Format
 1) 212
 CANCEL-ACTION (Format 1) 213
 CHARGE-CODE (Format 1) 214
 CONTROL-VALUE (Format 1) 214
 CORRELATION (Format 1) 216
 COUNT (Format 1) 217
 DESCRIPTOR-CONTENT-DEFINITION (Format
 41) 217
 DESTINATION-NODE-ADDRESS (Format
 1) 219
 DISTRIBUTION-IDENTIFIER (Format
 1) 220
 DISTRIBUTION-NAME (Format 1) 221
 DOCUMENT-PASSWORD (Format 1) 221
 DOCUMENT-TYPE (Format 1) 222
 EXCEPTION-CODE (Format 1) 223

FILE-OPTION (Format 1) 224
 FORMAT-PARAMETERS (Format 1) 225
 FORMATTED-DOCUMENT-NAME (Format 1) 225
 FORMATTER-NAME (Format 1) 225
 FUNCTION-SET (Format 1) 226
 GRAPHIC-CHARACTER-SET-ID (Format 1) 227
 IDENTIFIED (Format 1) 228
 IDENTIFIED-DATA (Format 2) 228
 IDENTIFIED-DATA (Format 3) 229
 IDENTIFIED-DATA (Format 41) 230
 IDENTIFIED-DATA (Format 42) 231
 LIBRARY-NAME (Format 1) 232
 LIBRARY-NAME (Format 41) 232
 LIST-ACTION (Format 1) 233
 MESSAGE (Format 1) 235
 MESSAGE (Format 2) 236
 MODIFY-DATA (Format 41) 236
 OBTAIN-OPTION (Format 1) 239
 ORIGINATING-NODE-ADDRESS (Format 1) 241
 PROCESS-NAME (Format 1) 242
 PROCESS-PARAMETERS (Format 1) 242
 PROCESS-PASSWORD (Format 1) 243
 RECIPIENT-ADDRESS (Format 1) 243
 RECIPIENT-ADDRESS (Format 42) 244
 RECIPIENT-PASSWORD (Format 1) 245
 RECOVERY-ACTION (Format 1) 245
 REPLY-DATA (Format 1) 246
 RETRIEVE-COUNT (Format 1) 246
 SCAN-DATA (Format 1) 247
 SEARCH-DATA (Format 41) 248
 SEARCH-DATA (Format 42) 254
 SEARCH-OPTION (Format 1) 258
 SEARCH-REQUEST-NAME (Format 1) 259
 SELECT-TIME (Format 1) 259
 SIGN-ON-ID (Format 1) 259
 SIGN-ON-ID (Format 42) 260
 SIGN-ON-PASSWORD (Format 1) 261
 SOURCE-ADDRESS (Format 1) 261
 SOURCE-ADDRESS (Format 42) 262
 SOURCE-PASSWORD (Format 1) 263
 STATUS-INFORMATION (Format 1) 263
 TIME-LIMIT (Format 1) 265
 ORIGINATING-NODE-ADDRESS (Format 1) 241
 owner-delegate 16

P

prefix 46
 primary owner 15
 principal 17
 private documents 16
 PROCESS-NAME (Format 1) 242
 PROCESS-PARAMETERS (Format 1) 242
 PROCESS-PASSWORD (Format 1) 243
 process, DIA 6
 local 6
 remote 6
 requester roles 6
 server roles 6
 public documents 16

R

recipient 7
 recipient address 8
 recipient node 7
 RECIPIENT-ADDRESS (Format 1) 243
 RECIPIENT-ADDRESS (Format 42) 244
 RECIPIENT-PASSWORD (Format 1) 245
 RECOVERY-ACTION (Format 1) 245
 REPLY-DATA (Format 1) 246
 requester 9
 RETRIEVE-COUNT (Format 1) 246

S

SCAN-DATA (Format 1) 247
 Search Result List 20
 SEARCH-DATA (Format 41) 248
 SEARCH-DATA (Format 42) 254
 SEARCH-OPTION (Format 1) 258
 SEARCH-REQUEST-NAME (Format 1) 259
 SELECT-LIMIT (Format 1) 259
 sequence number 44
 server 9
 session establishment 18
 Session Services 11
 ACKNOWLEDGE 82
 CORRELATION 83
 exception conditions 85
 EXCEPTION-CODE 83

RECOVERY-ACTION 83
 REPLY-DATA 83
 DELETE 92
 exception conditions 94
 IDENTIFIED-DATA 92
 ORIGINATING-NODE-ADDRESS 92
 SOURCE-ADDRESS 92
 SOURCE-PASSWORD 93
 DELIVER 95
 CORRELATION 95
 exception conditions 97
 IDENTIFIED-DATA 95
 exception conditions 76
 SET-CONTROL-VALUE 86
 CONTROL-VALUE 86
 exception conditions 88
 support considerations 89
 SIGN-OFF 81
 SIGN-ON 72
 CHARGE-CODE 73
 CORRELATION 74
 COUNT 73
 DOCUMENT-TYPE 74
 FUNCTION-SET 73
 GRAPHIC-CHARACTER-SET-ID 74
 SIGN-ON-ID 73
 SIGN-ON-PASSWORD 73
 support considerations 78
 CHARGE-CODE 79
 COUNT 80
 DOCUMENT TYPE 80
 function set 79
 GCID 80
 SIGN-ON-ID 79

SIGN-ON-PASSWORD 79
 shared-private documents 16
 SIGN-ON-ID (Format 1) 259
 SIGN-ON-ID (Format 42) 260
 SIGN-ON-PASSWORD (Format 1) 261
 solicited environment 7
 source 7
 source address 8
 source node 7
 SOURCE-ADDRESS (Format 1) 261
 SOURCE-ADDRESS (Format 42) 262
 SOURCE-PASSWORD (Format 1) 263
 STATUS-INFORMATION (Format 1) 263
 structured-field segmentation 56
 suffix 54
 synchronous reply required command class
 (SRR) 37

T

temporary agent 17
 TIME-LIMIT (Format 1) 265
 Type 1 suffix 54
 Type 2 suffix 54

U

unsolicited environment 7

Document Interchange Architecture:
Technical Reference
Order No. SC23-0781-0

**READER'S
COMMENT
FORM**

You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name, company, mailing address, and date:

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A.
(Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

Reader's Comment Form

Fold and Tape

Please Do Not Staple

Fold and Tape



NO
POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED
STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.



POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department D33, Building 803
11400 Burnet Road
Austin, TX, USA 78758

Fold and Tape

Please Do Not Staple

Fold and Tape

Document Interchange Architecture: Technical Reference Printed in U.S.A. SC23-0781-0



Document Interchange Architecture: Technical Reference Printed in U.S.A. SC23-0781-0

