DOCUMENT INTERCHANGE ARCHITECTURE: CONCEPTS AND STRUCTURES

SC23-0759-0

DOCUMENT INTERCHANGE ARCHITECTURE: CONCEPTS AND STRUCTURES

SC23-0759-0

First Edition (June 1983)

Changes are made periodically to the information herein; any such changes will be reported in subsequent revisions or Technical Newsletters.

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 D31, 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.

PREFACE

Document Interchange Architecture (DIA) is a program-to-program communication architecture which 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 interchange data in an office system network. DIA logically divides into several parts: an information interchange base and various DIA application services.

This manual describes the DIA information interchange base which consists of the DIA concepts, protocols, data structures, and session services. The facilities described within the DIA information interchange base are common throughout the various DIA application services. Common topics such as function subsetting, session control, and error recovery are also presented in this manual.

This manual is intended for data processing managers, system analysts, designers, system programmers, and application programmers, as well as systems engineers and product support representatives.

RELATED PUBLICATIONS

- Office Information Architectures: Concepts, GC23-0765-0
- Document Interchange Architecture: Document Distribution Services Reference, SC23-0762
- Document Interchange Architecture: Document Library Services Reference, SC23-0760
- Document Interchange Architecture: Application Processing Services Reference, SC23-0761
- 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
- SNA Concepts and Products, GC30-3072.

CONTENTS

Chapter 1. Introduction 1
Office Information Architectures 1
  Today's Office System Requirements 1
  Document Interchange Architecture Services 2
  Benefits of Document Interchange Architecture 3
DIA Goals 4

Chapter 2. DIA Concepts and Services 5
Logical Components of an Office System Network 5
DIA Services 6
  DIA Session Services 6
  Document Library Services 7
  Document Distribution Services 8
  Application Processing Services 11

Chapter 3. Request/Reply Protocols 13
DIA Command Classes 14
DIA Request/Reply Correlation 14
DIA Command Class Protocol Rules 16
DIA Request-Reply Protocols Rules 16

Chapter 4. Function Sets 19
Function Set Negotiation 19
Function Set Definition 19
  DIA Session Services 19
  Document Distribution Services Commands 20
  Document Library Services Commands 24
  Application Processing Services Commands 25

Chapter 5. Document Interchange Unit 27
Document Interchange Unit (DIU) Components 27
DIU Structured Field 27
  DIU Introducer 28
  DIU Introducer Extension (ISS) 29
  Structured Field Documentation Conventions 29
Document Interchange Unit (DIU) Structure 30
  DIU Prefix 31
  Command Sequence 31
    Command 32
    Operands 33
  Data Units 34
  Document Units 34
    Document Unit Identifier 35
    Document Profile 36
    Document Content Introducer 37
  DIU Suffix 38
Structured Field Segmentation 38
Summary of DIU Syntax Rules 40
GCID Encodings 42

Chapter 6. Session Services 45
DIA Session 45
Session Services Command Descriptions 46
SIGN-ON 46
SIGN-OFF 54
ACKNOWLEDGE 55
SET-CONTROL-VALUE 59

Chapter 7. Exception Detection, Classification, and Reporting 63
Exception Condition Detection 63
Exception Condition Classification 63
Exception Condition Reporting 64
Recommended Recovery Actions 67

Appendix A. Session Services Operands 69
CHARGE-CODE (Format 1) 69
CONTROL-VALUE (Format 1) 69
CORRELATION (Format 1) 71
COUNT (Format 1) 72
DOCUMENT-TYPE (Format 1) 72
EXCEPTION-CODE (Format 1) 73
FUNCTION-SET (Format 1) 73
GRAPHIC-CHARACTER-SET-ID (Format 1) 74
RECOVERY-ACTION (Format 1) 75
REPLY-DATA (Format 1) 76
SIGN-ON-ID (Format 1) 76
SIGN-ON-ID (Format 42) 77
SIGN-ON-PASSWORD (Format 1) 78

Appendix B. Code Points 79

Appendix C. DIA Document Types 85

Appendix D. Graphic Character Sets 87

Appendix E. DIU General Exception Conditions 93
Prefix Exception Conditions 93
Command Exception Conditions 94
Document Unit Exception Conditions 96
Data Unit Exception Conditions 97
Suffix Exception Conditions 97

Glossary 99

Index 105

FIGURES

1. Function Set 10 20
2. Function Set 2 21
3. Function Set 3 22
4. Function Set 4 22
5. Function Set 5 23
6. Function Set 6 23
7. Function Set 7 24
8. Function Set 8 25
9. Function Set 9 26
10. DIU Structured Field 28
11. Document Interchange Unit Structure Overview 30
12. DIU Prefix 31
13. DIU Command Sequence 32
14. DIU Data Unit 34
15. DIA Document Unit Type 3 35
16. Document Unit Identifier 35
17. DIU Document Profile 36
18. DIU Document Content Introducer 37
19. DIU Suffix 38
20. Document Unit Segmentation Example 40
21. Summary of DIU Syntax Rules 41
22. Exception Condition Code Format 65
23. Document Type Code Assignments 85
24. Character Set 103 of Code Page 256 87
25. Character Set 337 of Code Page 256 88
26. Description of Code Page 256 Graphics. 89

CHAPTER 1. INTRODUCTION

This chapter introduces IBM's Office Information Architectures, briefly identifies the requirements of today's office systems, introduces one of these architectures—Document Interchange Architecture, and describes its benefit to the user.

OFFICE INFORMATION ARCHITECTURES

IBM's Office Information Architectures are a set of specifications for dissemination and management of information in an office system network. An office system network, in this sense, represents the collection of interconnected IBM office systems. The architectures define the form of the information transmitted on the network and, further, define the rules governing the use of the information among the systems of the network.

The term architecture refers to a set of design principles that define the relationships of and interactions between various parts of a system or network of systems. The office information architectures include Document Interchange Architecture (DIA) and Document Content Architecture (DCA). This publication introduces the basic concepts, services, and data structures of Document Interchange Architecture. The Document Content Architectures are described in Revisable-Form-Text Reference and Final-Form-Text Reference.

Document Interchange Architecture defines functions for interchanging documents and other information between separate office systems that are connected through a network. Document Interchange Architecture is considered a part of IBM's Systems Network Architecture (SNA); this book introduces only the DIA part of SNA. SNA as a whole is introduced in SNA Concepts and Products.

Today's Office System Requirements

This book uses the term document to refer to the user-created information that flows through and between office systems. The term includes messages and other kinds of information not ordinarily thought of as documents.

The typical office, whether or not it uses electronic office systems, performs some or all of these document-related activities:

- **Creating documents.** This includes preparing correspondence, reports, proposals, contracts, and manuscripts. This activity may include assembling a document from other documents that already exist.
- **Revising documents.** This may range from making minor corrections to editing or rewriting the entire document.
- **Distributing documents.** Documents may be distributed to individuals (or to files) via internal or external mail, hand delivery, or electronic means.

- **Filing and retrieving documents.** Documents may be filed in and retrieved from file cabinets, libraries, or electronic storage. These activities may include logging and tracking to promote orderly filing and retrieval.

The automation of offices is becoming a reality for an increasing number of organizations. Office automation is helping these organizations improve the productivity and effectiveness of office workers and the timeliness and accuracy of the information on which they depend.

By using computer processing, today's office systems offer the potential for many other capabilities—not just faster typing, but the ability to integrate data files with text; store and retrieve correspondence and reports electronically; distribute documents electronically; and support the day-to-day activities of administrative personnel, professionals, and managers.

While some of the benefits of electronic document processing can be realized from a single, standalone office system, a network that interconnects office systems in many parts of an organization can bring greater gains in productivity.

Physically, a network is a combination of interconnected equipment and programs used for moving information between points where it may be generated, processed, stored, and used. From the viewpoint of its users, a network is a collection of services—in the case of an office system network, services useful in creating, revising, distributing, filing, and retrieving documents.

Office systems may differ in several ways, for each offers different capabilities and answers the needs of different users. The thread that ties the systems together is information interchange. The goal is to let these dissimilar office systems communicate easily with one another in a universally understandable manner.

What is needed is a uniform structure for information that is interchanged between office systems. This structure must have an encoding scheme that is designed to convey any document, regardless of its content, from one kind of office system to another; and to communicate the intent of the person who creates or sends a document as to how it is to be processed.

The encoding scheme must also be flexible and extendable to allow it to accommodate new requirements as they arise. Rules must also be established to cause the various office systems to interpret documents uniformly and act upon them in a consistent manner.

IBM meets the challenge of information interchange between office systems with Document Interchange Architecture and Document Content Architecture.

Document Interchange Architecture Services

Document Interchange Architecture defines how documents and requests for document distribution and processing functions are to be communicated through an office system network. DIA specifies the rules and data structure that establish the discipline for unambiguous interchange of documents and other information between office systems.

DIA provides these categories of services for the interconnection of office systems:

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

Document Library Services allow users to file documents in a document library, to retrieve them or delete them from the library, and to search the library for documents that meet user-specified criteria, such as the the name of the author. These criteria are compared with document descriptors that are stored with the document. For all cases where there is a match between the search criteria and the document descriptor, the user may obtain the document or the document descriptor.

Document Distribution Services deliver documents and related information from their source to one or more recipients anywhere in the network. These services can, for example, allow a user to enter a single request to distribute a document to multiple recipients, schedule distribution by document priority, confirm delivery, and report errors. Document Distribution Services are commonly referred to as electronic document distribution.

Application Processing Services allow users to modify document descriptors used in searching a library; to invoke a program to transform documents from one format to another, for example, revisable-form-text to final-form-text; and to execute user-supplied programs.

Benefits of Document Interchange Architecture

Document Interchange Architecture supports a logical view of an office system network that allows its users to request document distribution and processing functions, address recipients, and retrieve documents from a library without having to know anything about the physical organization of the network. Specific benefits of this architecture are as follows.

Document Library Services: These services permit users to file documents in and retrieve or delete them from a document library, and to search the library for documents that meet user-specified criteria. These are important services for office systems that are limited in storage capacity, where the permanent archiving of critical information is necessary, or where documents must be obtained by many locations on a demand basis. Document library services provide an organization with a means to organize, manage, and control its information assets.

Document Distribution Services: These services, through the use of electronic document distribution in offices, achieve the timely and efficient distribution of correspondence, reports, contracts, proposals, and other information.

Application Processing Services: Services are provided to transform documents and to modify the document descriptors of stored documents. These services include an interface to user-written programs that can be developed to

accomplish specific functions and can be invoked through application processing services. An example of a user-written program might be one that searches the document library for documents containing a user-specified expiration date, deletes these documents from the library, and records the deletions in a report.

DIA GOALS

The goals of the Document Interchange Architecture are:

- Provide a consistent, durable, and extendable interface to interchange information between diverse IBM systems and products operating in an office system network
- Define the syntax, semantics, protocols, data definitions, and algorithms required for the automation of common office system application services
- Define the required and optional processing capabilities for each office system application service
- Provide for the flexible and transparent placement of office system application services in the network.

CHAPTER 2. DIA CONCEPTS AND SERVICES

This chapter describes the basic concepts of Document Interchange Architecture, defines the logical components of an office system network, and explains each of the categories of services defined by Document Interchange Architecture.

Document Interchange Architecture defines the protocols and data streams necessary to interchange information such as documents and messages in a consistent, predictable manner.

Document Interchange Architecture defines a set of services which are performed by processes implemented in the uppermost layer of IBM's Systems Network Architecture. DIA specifies how these processes, located throughout the network, communicate with each other to perform required office system functions.

Each DIA service performs specified functions requested by end users. An end user is defined as an application program, a device, or a human being and, as such, represents the source or the recipient of information flowing through the office system network. Each end user of a DIA process is uniquely identified in the network by a logical address.

The information exchanged by DIA services comprises DIA commands and user information. Typical commands are: distribute a document from office system A to office systems B, C, and D; retrieve document XYZ from a document library; and search a document library for documents that satisfy search criteria J, K, L.

Document Interchange Architecture is considered a part of SNA. (Only that part of SNA is introduced by this book; SNA as a whole is described in SNA Concepts and Products.) However, DIA is not dependent on the specific presentation and transport services of the network, and is not concerned with the content of the documents being interchanged among office systems. The term document is used throughout Document Interchange Architecture in the most generic sense and is defined to be any collection of data.

LOGICAL COMPONENTS OF AN OFFICE SYSTEM NETWORK

A network of office systems based on Document Interchange Architecture contains a set of interrelated logical components that lie within the physical components of the network. The logical components are defined by DIA and are implemented by IBM products as processes executing in physical components. These logical components are:

- A source node provides DIA services, acting on behalf of one or more end users, that initiate and control the interchange of documents and messages with end users called recipients.

- A recipient node provides DIA services, acting on behalf of one or more end users (recipients), that control and receive documents and messages sent by a source node.

- An office system node (OSN) provides DIA services that receive, store, route, and deliver information for source and recipient nodes. An OSN contains storage capabilities providing the document library for attached source or recipient nodes. An office system node can also interact with an appropriately configured network to distribute information to other office system nodes.

Source nodes, recipient nodes, and office system nodes interchange documents and messages through an office system network using the transport services of the network. The nodes are uniquely identified in the network. Specifically, a source node is identified by a source address, a recipient node is identified by a recipient address, and an office system node is identified by either an originating node address or a destination node address. An OSN is an originating node when it supports a source node and is a destination node when it supports a recipient node.

Originating node addresses and destination node addresses are unique within the network. Source and recipient addresses are unique within originating nodes and destination nodes, respectively.

An OSN process can act both as an originating node and destination node concurrently. In this case, the originating node address and the destination node address are identical. Similarly, a DIA process can act in the capacity of both a source node and recipient node. In this mode, the values of the source address and recipient address are identical.

DIA SERVICES

The categories of DIA services are as follows:

- DIA Session Services

- Document Library Services

- Document Distribution Services

- Application Processing Services.

DIA Session Services

DIA processes use DIA session service commands to establish a logical connection, called a DIA session, through which they may exchange information. The DIA session exists after the two DIA processes identify themselves and agree on the scope of work that is to be performed. This agreement is necessary because not all DIA implementations 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.

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. The function sets defined for document distribution services, for example, enable documents or messages to be transferred from source nodes to office system nodes, from source nodes to recipient nodes, and from office system nodes to recipient nodes. Other function sets provide the DIA commands needed for document library services and application processing services. Function sets are defined in "Chapter 4. Function Sets" on page 19.

Document Library Services

Document library services are used for storing and retrieving documents electronically. These functions are analogous to the manual filing and retrieving of paper documents that take place in most offices.

However, document library services can also perform activities that are cumbersome in a manual system. For example, when a document is electronically filed in a document library, a set of descriptors called a document profile is filed with it. The profile contains parameters that identify the contents of the document, for example, the name under which it is filed, the authors, the subject it covers, and the date it was filed in the document library.

Document profiles are used in searching for documents in a document library. For example, a user can ask the office system to search for all documents about a particular subject and by a certain author that the library received between any two dates. Upon completing the search, the office system node would give the user a list of the documents that met the search criteria. The user could then ask the office system to retrieve a copy of a specific document on the list from the library and deliver it to the user for printing or viewing.

A document library services source node provides the following functions:

- Allows end users to file documents in, and retrieve or delete them from, the library.
- Allows authorized end users other than the ones that filed the documents to retrieve them from the library.
- Allows authorized end users to search for and retrieve documents in the library for other end users. As an example, a secretary can modify documents on behalf of those who generated them.

An office system node performing library services provides storage for the document library and performs the functions that end users request through source nodes. These functions are:

- Places documents received from source nodes into the document library
- Assigns each document it files in the document library a unique name called the library-assigned document name. This name is returned to the requestor and can be used to uniquely identify the document at some later time.

- Searches the document profile descriptors of documents in the library that the end user has authority to access and returns to the source node a list of all documents that meet the supplied search criteria.

- Delivers documents to the source node from which they were requested.

- Deletes documents from the library upon request from authorized end users.

Document Distribution Services

Document distribution services deliver information such as messages or documents, with or without an appended message, from source nodes to recipient nodes within an office system network. Documents and messages can be distributed between source and recipient nodes during a single DIA session or by routing them through office system nodes for subsequent delivery to a recipient node.

When documents or messages are delivered through an office system node, document distribution services in the source node do not establish a DIA session with document distribution services in the recipient node. Instead, the DIA session is established between the source node and the office system node. After the session is established, the information passes from the source node to the office system node. If the recipient node is located on a different office system node, the information is passed to that office system node.

When the recipient node establishes a DIA session with its office system node, it can obtain a summary list of documents and messages to be delivered, it can take delivery on any or all of the information to be delivered, or it can cancel delivery of any or all of the information.

The sender of a document or message can specify a distribution priority for it relative to other distribution requests. That is, senders can cause some information to reach their recipients faster than others.

The sender of a document or message can also request notification of delivery of a document or message to its recipients. The notification is called a confirmation-of-delivery message.

Document distribution services allow users to send a document or message to a distribution list defined in an office system node. The office system node will queue a copy of the document or message to each recipient defined on the distribution list. Each recipient can then request delivery of his individual copy.

DIA assigns to each distribution request a distribution document name that uniquely identifies the request within the office system network. DIA uses this name to correlate confirmation-of-delivery messages and error messages with their corresponding distribution requests.

A document distribution services source node provides the following functions for end users:

- Distributes information such as messages or documents, with or without an appended message, to one or more recipients located in the office system network.
- Prioritizes the distribution request so that distribution information of higher priority is delivered before information of lower priority.
- Requests that a confirmation-of-delivery message be returned to the sender of the document or message when the recipient node accepts 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 messages, for example a notification that the intended recipient is invalid, possibly due to a misspelled recipient address. Feedback messages 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 an additional authorization before receiving the distributed information. For example, a manager might distribute a personal and confidential document or message to a group of recipients authorized to receive such material, and be assured that only those recipients could receive it.
- Requests distributions on behalf of other end users.

A document distribution services recipient node provides the following functions for end users:

- Exchanges information directly while in a DIA session with the source node
- Determines what distributed information is available for delivery at the office system node
- Obtains distribution information that is ready for delivery (either all distributions or only the ones characterized by a particular class of service such as priority, nonpriority, or personal) at the office system node
- Cancels delivery of the recipient's documents or messages that are available at the office system node
- Requests delivery of documents or messages on behalf of other end users.

Document distribution services in an office system node asynchronously distribute documents or messages to recipients located in the office system network. Distributing information asynchronously means that a recipient node need not have an active DIA session with its office system node to receive the information from the source node. The information remains in the office system

node until the recipient node establishes a DIA session with the destination office system node; then it is delivered upon request.

The functions performed by document distribution services in an office system node are logically divided into two groups: originating OSN functions and destination OSN functions. Originating OSN functions are those required when a source node is in a DIA session with the office system node; destination OSN functions are those required when a recipient node is in a DIA session with the office system node. Since a node can be a source node and a recipient node within a single DIA session, the same DIA process can accommodate both attached source nodes and attached recipient nodes.

An originating OSN provides the following functions:

- Assigns and returns to source nodes a unique distribution document name for each distribution request received.
- Stores the distribution request and information to be distributed.
- Routes the distribution request and the associated information to the office system nodes that serve the specified recipients. If the destination OSN is not the same as the originating OSN, the originating OSN distributes the distribution request and information to the destination OSN that serves the specified recipients.
- Maintains a correlation table for confirmation-of-delivery messages that are currently outstanding. As confirmation-of-delivery messages are returned by destination OSNs, the originating OSN updates the correlation table. When queried by an attached source node, the originating OSN 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.

A destination OSN provides the following functions:

- Places distribution requests and information on a queue until they can be delivered to recipients. Multiple recipients can be defined to a destination OSN within a recipient distribution list, a list of one or more recipients served by the destination OSN. The OSN queues the distribution request and associated information for each recipient listed.
- Delivers distribution requests and information upon request by recipient nodes.
- Sends confirmation-of-delivery messages to the originating OSN when the recipient node takes delivery of the distribution request and information. The originating OSN returns the COD message to the source nodes that requested the confirmations.
- Lists the names of distributions contained in OSN queues for delivery to recipient nodes.
- Cancels delivery of specified distribution upon request.

Application Processing Services

Application processing services defines commands that cause an office system node to perform several additional functions. These additional functions allow end users to manipulate document profiles associated with a document (for example, to add or delete the descriptors), to invoke a program to transform documents from revisable-form-text to final-form-text, and to invoke specific application programs, procedures, or processes.

An application processing services source node provides the following functions for end users:

- Requests execution of programs within the office system node
- Requests the modification of descriptors in a document profile
- Invokes programs to format documents.

An application processing services office system node provides functions requested by end users at source nodes. These functions are:

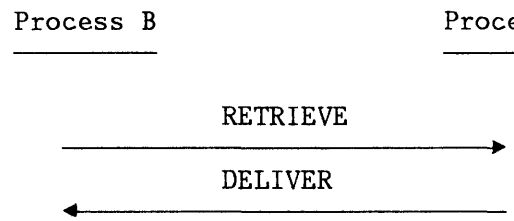
- Interprets and validates requests from the source nodes
- Modifies descriptors in the document profile specified by end users
- Schedules execution of programs and procedures requested by end users
- Executes programs to transform documents from revisable-form-text to final-form-text.

CHAPTER 3. REQUEST/REPLY PROTOCOLS

This chapter describes how DIA processes use DIA commands to exchange information. It also defines the different types of command classes used to request application services and the request/reply protocols rules for each command class.

Information exchanged between DIA processes consists of commands and user information. Typical commands are: distribute a document from one office system to other office systems; retrieve a document from the document library and return it to the requestor.

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



The above scenario illustrates the basic DIA request/reply protocol. The RETRIEVE command is the function requested and the DELIVER is the replying command to the RETRIEVE function request. The scenario also illustrates that the server (Process A) responds on demand to the requestor (Process B). The on demand protocol is one class of request/reply protocol defined by DIA that can be used by the function requestor to request the function server to perform a unit of work.

To invoke an application service function, the requestor must specify the command to be performed, any associated input data, and the request/reply protocol to be used. The type of request/reply protocol is specified by the command class, for example, a synchronous reply (command) is required. The function server responds to the function request according to the command class specified on the request. The results of the function request are returned with a DIA command. The replying command also specifies the command class (request/reply protocol) to be used in processing the function reply. A function request may result in multiple replying commands being returned.

DIA COMMAND CLASSES

The DIA command classes defined are:

- No Reply Required Command Class (NRR)

  The NRR command class is used by the requestor when the function requested does not require a replying command.

- Synchronous Reply Required Command Class (SRR)

  The SRR command class is used by the requestor when the function requested is to be performed synchronously (immediately) by the server and the results returned with one or more replying commands. The scenario above is an example of an SRR request/reply scenario.

- Asynchronous Reply Required Command Class (ARR)

  The ARR command class is used by the requestor 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 requestor 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 application services using a subset of these command classes for each application service command. These application services are called function sets and are defined in "Chapter 4. Function Sets" on page 19.

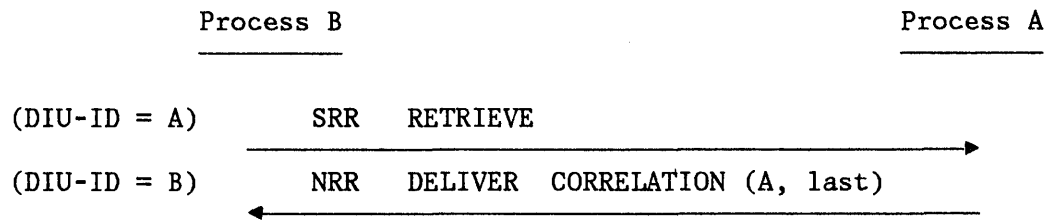
DIA REQUEST/REPLY CORRELATION

Requests and replies are each contained within a Document Interchange Unit (DIU) when they are exchanged between DIA processes. Each DIU sent may be uniquely identified with a DIU Identifier (DIU-ID).

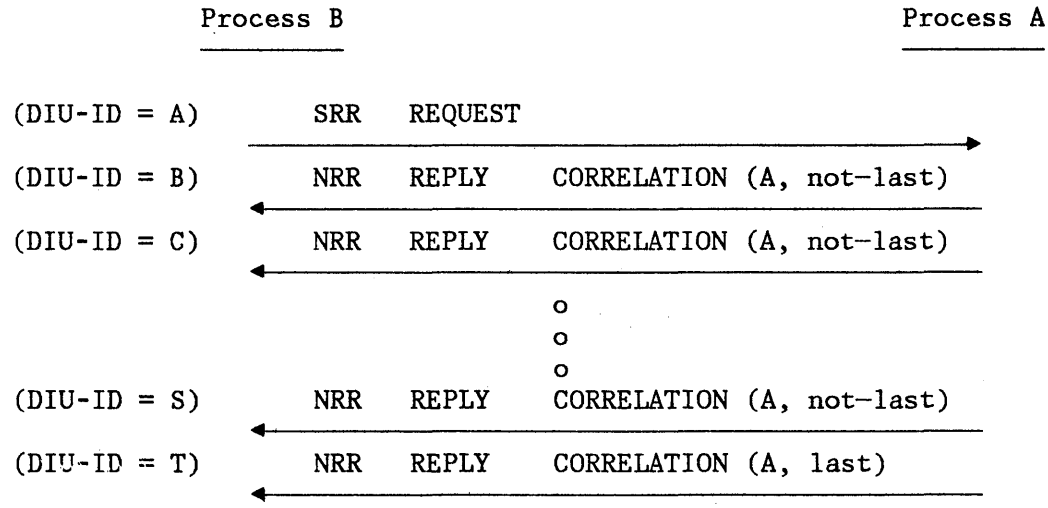
Note: DIU-ID is defined in the DIU Prefix; the definition of a DIU is found in "Chapter 5. Document Interchange Unit" on page 27.

Replying commands are correlated to a previously received function request (command) by referencing the DIU-ID of the function request and indicating whether the reply is last or not-last. This correlation information is contained in the CORRELATION operand of all replying commands. In other words, all replying commands must have a CORRELATION operand to correlate the reply to the function request to which they are replying. A definition of the CORRELATION operand may be found in "Appendix A. Session Services Operands" on page 69.

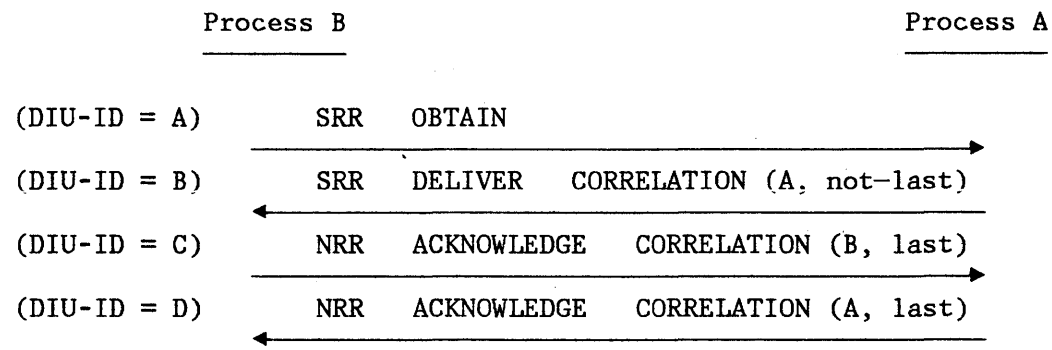
The following scenario illustrates the DIA request/reply correlation. Assume the RETRIEVE DIU has a DIU-ID = A and the DELIVER DIU has a DIU-ID = B, then the replying DELIVER command can be unambiguously correlated to the RETRIEVE request using the CORRELATION operand specifying the DIU-ID of the RETRIEVE 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 to be sent. The last replying command would have a CORRELATION operand that indicated it was the last replying command, for example:



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



In the above, the DELIVER is the last replying command to the RETRIEVE command. In turn, 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, which in turn, was a replying command to the RETRIEVE.

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 nor correlated.
  - An ACKNOWLEDGE with an exception condition code is allowed to reply to and reference this class of command. The exception condition information is for statistics or problem determination logging.
- SRR - Synchronous Reply Required Command Class
  - The SRR command requestor may not send another function request command until the last replying command has been returned. The replying command may be any command that has a CORRELATION operand correlating it with the SRR request.
- ARR - Asynchronous Reply Required Command Class

  The ARR command class is used for any command that requires a replying command 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 to ARR commands may be sent in any order.
  - Replies to ARR commands may 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.

DIA REQUEST-REPLY PROTOCOLS RULES

The flow of DIUs between two DIA processes is command-driven and complies with the command or reply protocols specified for each command defined by DIA. The following general rules apply to issuing commands within the DIA Session:

- Only one DIA process can be sending command requests at any one time.
- The DIA process which 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 in any class, an ACKNOWLEDGE command with an exception condition code will be returned in the NRR replying command class.

The mapping of DIA request/reply protocols to the SNA LU 6.2 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 implementing IBM product publications.

CHAPTER 4. 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 exists after the two DIA processes identify themselves and agree on the scope of work that is to be performed. This agreement is necessary because not all DIA implementations 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.

The negotiation includes the determination of the roles each process will play. The process that will be the command requestor is identified as Process B and the process which will be the command server is identified as Process A. In the case of symmetric interchange, for example, DIA processes capable of simultaneously acting as a requestor and a server of a DIA function, the DIA process must assume the role of both Processes A and B.

FUNCTION SET DEFINITION

Figure 1 on page 20 through Figure 9 on page 26 define how the DIA commands are grouped to form the various function sets. The figures list each command for the function set and identify the valid command class for each command and the request/reply protocol. 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 requestor and receive indicates 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.

DIA Session Services

DIA Session Services commands are used to establish, maintain, and terminate DIA sessions between DIA processes. These commands are summarized here. For the detailed definitions see "Chapter 6. Session Services" on page 45.

- The SIGN-ON command is used to establish a DIA session between two DIA processes, determine the functions of the interchange architecture to be used in that session, and enable 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 generalized replying command that is used to notify a DIA process that a requested DIU command completed normally or with exception.

- The SET-CONTROL-VALUE command provides the ability for one DIA process to establish, change, or delete the value associated with a control variable that is defined at the receiving DIA process, for example, a password.

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

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

Figure 1. Function Set 10

Document Distribution Services Commands

Document distribution services deliver information, such as messages or documents, with or without a message, from source nodes to recipient nodes within an office system network. The information can be distributed between source and recipient nodes during a single DIA session, or by routing them through office system nodes for subsequent delivery to a recipient node. A brief summary of each of these commands is included here. The detailed definition of these commands is contained in Document Distribution Services Reference.

- The CANCEL-DISTRIBUTION command cancels distribution status information or cancels the delivery of distributed documents or messages.

- The DELIVER command transports documents and messages from an office system node (OSN) to a source or recipient node. The DELIVER command may also be used to transport documents and messages directly between a source node and a recipient node without an intervening OSN.

- The LIST command requests delivery of a list of documents and messages queued for delivery at an OSN for a recipient node or a list of the status of information about previously distributed distribution requests.

- The OBTAIN command requests delivery of one or more documents and/or messages scheduled for delivery to the requestor.

- The PROCESS-BIT-STRING command requests an OSN to interpret a bit-stream representation of a DIA function request and perform the requested operation.

- The REQUEST-DISTRIBUTION command transports documents and/or messages from a source node to an OSN for distribution to the specified recipient nodes. Documents to be distributed may be submitted with the command, located in the command server's document library or in a library accessible to the command server. Messages to be distributed can only be submitted with the command.

- The STATUS-LIST command notifies the recipient node that one or more documents or messages are available from the distribution system or that information about the progress of previous distribution requests is available.

The following figures (Figure 2 through Figure 7) show how these commands are grouped into the 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; namely, the recipient node must specifically request the OSN to deliver the information.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTOR)
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 2. Function Set 2

Function Set 3 contains the DIA commands used to deliver information from an OSN to recipient node in a unsolicited environment; namely, the OSN delivers the information to the recipient node without being specifically requested.

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

Figure 3. Function Set 3

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

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTOR)
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 4. 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 (REQUESTOR)
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 5. Function Set 5

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

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTOR)
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 6. 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 (REQUESTOR)
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 7. Function Set 7

Document Library Services Commands

Document Library Services commands provide the functions for maintaining user documents in a document library. A summary of each of these commands is shown here. Details of these commands are described in Document Library Services Reference.

- The DELETE command permanently removes access to the identified document for the delete requestor. A document that has two or more owners will be removed from library storage when all of the owners have requested that it be deleted.
- The DELIVER command transports a document from a server node to a requestor 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 an authorized document requestor.
- The SEARCH command locates the documents in the library that have characteristics that match search criteria specified by the requestor of the search. It creates and preserves a named list of references or pointers to the search selected documents. The list of references may be used for retrieving the document descriptors or the document contents.

Figure 8 is a table showing how these commands are grouped into the Document Library Services function set.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTOR)
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 8. Function Set 8

Application Processing Services Commands

Application Processing Services commands request the execution of tasks by another process. These commands are summarized here. The detailed specifications of these commands is in Application Processing Services Reference.

- The DELIVER command transports a document from a server node to a requestor node.
- The EXECUTE command requests an office system node to invoke the named process for execution.
- The FORMAT command requests an office system node to execute the named formatting process using the identified document for the format input object.
- The MODIFY command requests an office system node to 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.

Figure 9 is a table showing how these commands are grouped into the Application Processing Services function set.

COMMAND	COMMAND CLASS	PROCESS A (SERVER)	PROCESS B (REQUESTOR)
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

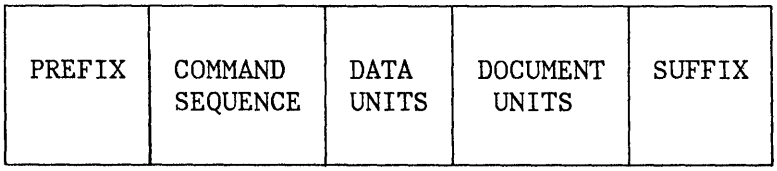
Figure 9. Function Set 9

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 introduces and identifies the DIU.
- The command sequence contains the commands that specify the functions to be performed. The command sequence may contain from 1 to 255 commands.
- A data unit contains information referenced by one or more commands in the command sequence. A DIU may contain from zero to 255 data units.
- A document unit contains the document profile descriptors and the document content. A DIU may contain from zero to 255 document units.
- A suffix terminates the DIU.

These data stream components may consist of structures called subcomponents. Examples of subcomponents are command operands and document profiles. All DIU components and their subcomponents are self-defining, variable length structured fields.

DIU STRUCTURED FIELD

The DIU structured field consists of four parts: the total length (LL) of the structured field, the structured field identifier (IDF), an optional structured field identifier extension (ISS), and an optional data variable as illustrated in Figure 10.

The structured field data variable may consist of scalar data values, fixed formatted data vectors, or self-defining fields of any form. 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 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.

The DIU structured field is defined as follows:

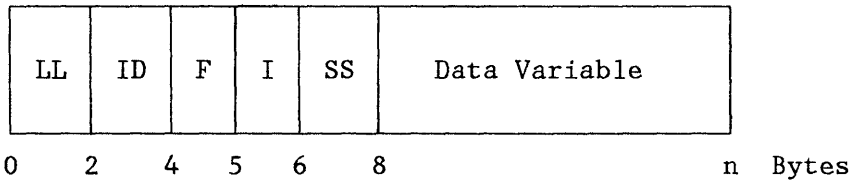


Figure 10. DIU Structured Field

LL = Structured field length

The length LL may vary from 5 to 32,767 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, the 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 ISS is present or not. The F byte is defined as follows:

```
Bit  0  -   Introducer Extension Indicator
            1 = ISS is present and follows the 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 IDF values of the DIU structured fields are defined in "Appendix B. Code Points" on page 79.

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'

The presence of an ISS introducer extension in a structured field is indicated by setting the high-order bit of the F byte to one. 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 data stream components (structured fields) may be segmented to accommodate DIA processes with limited buffer sizes, data lengths greater than 32,767 bytes, and situations where 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 the I byte, and is discussed in detail in "Structured Field Segmentation" on page 38.

Structured Field Documentation Conventions

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

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

DOCUMENT INTERCHANGE UNIT (DIU) STRUCTURE

Figure 11 illustrates the DIU data stream components and subcomponents. The DIU data stream components are: prefix, command sequence (command), data units, document units, and suffix. Each of the DIU data stream components and subcomponents are defined in the following sections.

All DIU data stream components and subcomponents 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.

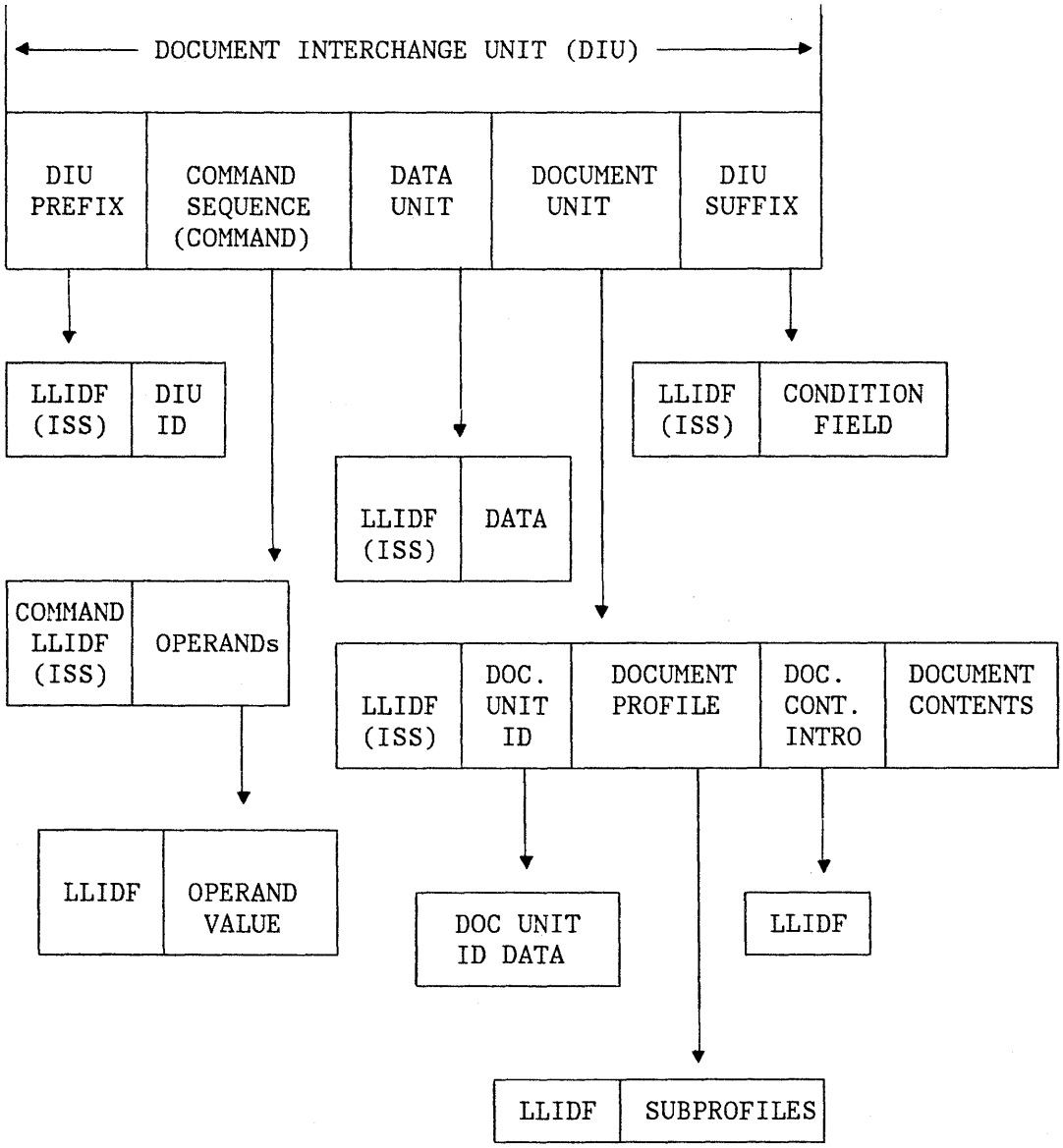


Figure 11. Document Interchange Unit Structure Overview

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 3. Request/Reply Protocols" on page 13.

The DIU-ID field is defined to be a 0- to 16-byte user-supplied value and may 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 and the type byte (D) defines the architecture version of this DIU.

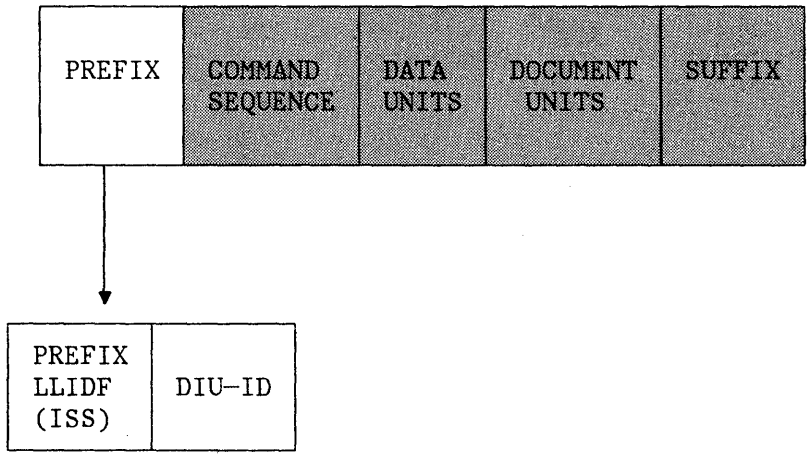


Figure 12. DIU Prefix

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 they are specified in the command sequence.

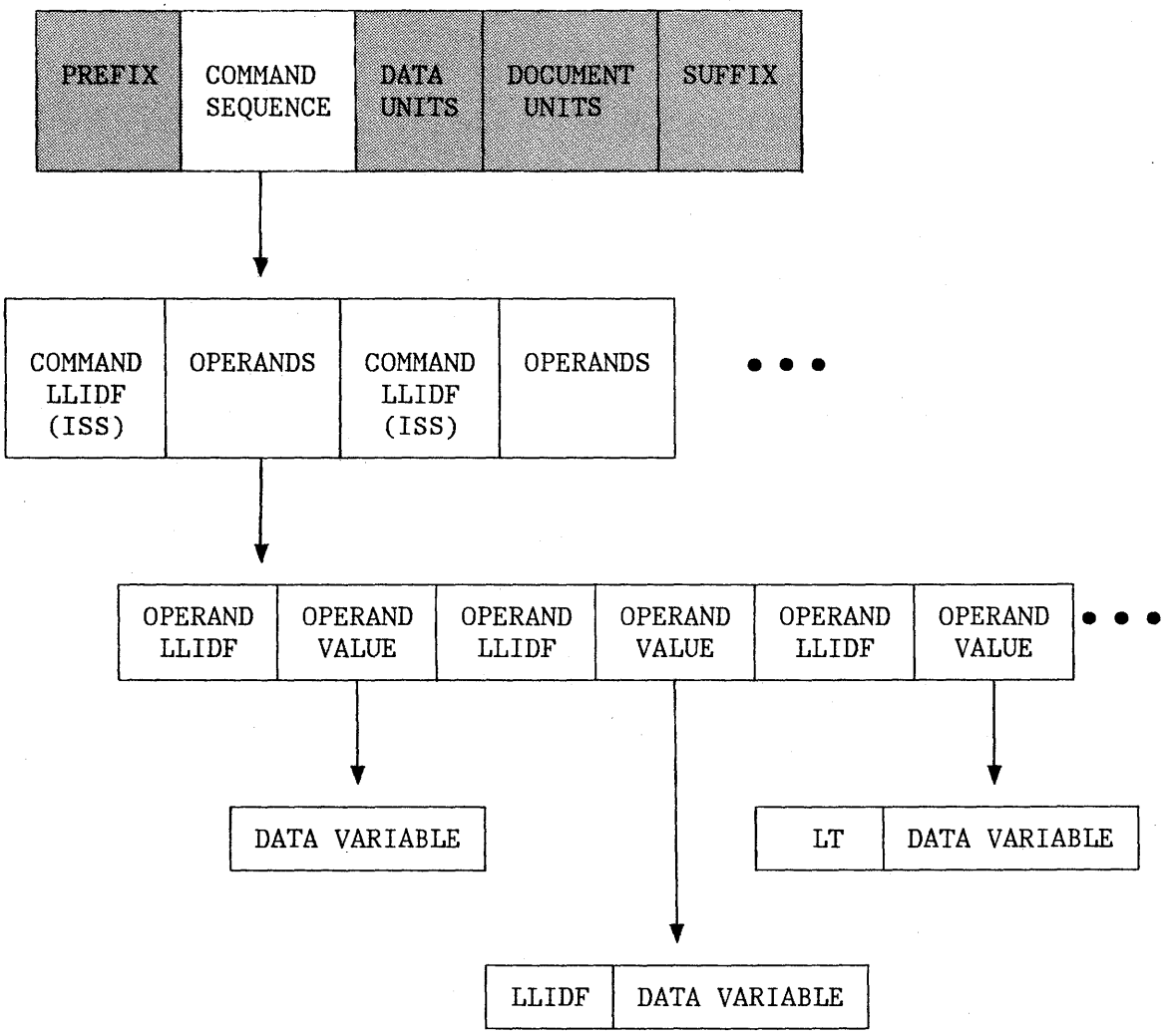


Figure 13. DIU Command Sequence

Command

Each command consists of its structured field introducer LLIDF(ISS) and zero or more operands. The command operands are totally contained in the commands structured field data variable; hence, the length (LL) of the command 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 two byte command structured field identifier (ID) defines the command class (for example, NRR, SRR, or ARR) and the specific command type (for example, a FILE command, a DELIVER command, and so on). The introducer format (F) byte defines the format and structure of the data variable. In this case, the format F byte defines the required and optional operands and order of the operands within the data variable, if any, that is required by this command. The DIA command classes are defined in "Chapter 3. Request/Reply Protocols" on page 13. The specific DIA commands are defined in the various reference manuals that describe the DIA application services, for example, Document Distribution

Services, Document Library Services, and so forth. Commands may be segmented. A description of structured field segmentation is found in "Structured Field Segmentation" on page 38.

Operands

Operands either contain or reference 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 operands 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 of the relative pointer is defined by the format F byte.
- 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 of the relative pointer is defined by the format F byte.

The operand type defines the specific type of operand, for example, a RECIPIENT-ADDRESS operand, a CORRELATION operand, and so forth.

The operand introducer format (F) byte defines the format and structure of the data variable. For immediate data operands, the operand data variable may consist of scalar data values, fixed formatted data vectors, or self-defining fields of any form. 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 F byte X'01' is defined to be a 1-byte binary number in the range from 1 to 255 whose value corresponds to the relative occurrence within the DIU of the referenced data unit or document unit.

Data Units

A data unit contains information that is referenced 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.

Data units are referenced by operands within the command sequence. The format of the data variable within the data unit corresponds to the format of the operand referencing the data unit had the operand been defined as an immediate data type operand. Data units may be segmented. A description of structured field segmentation is found in "Structured Field Segmentation" on page 38.

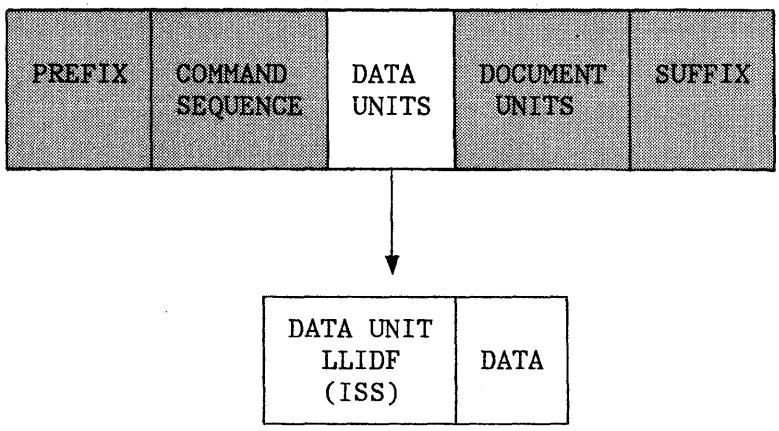


Figure 14. DIU Data Unit

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 15 on page 35.

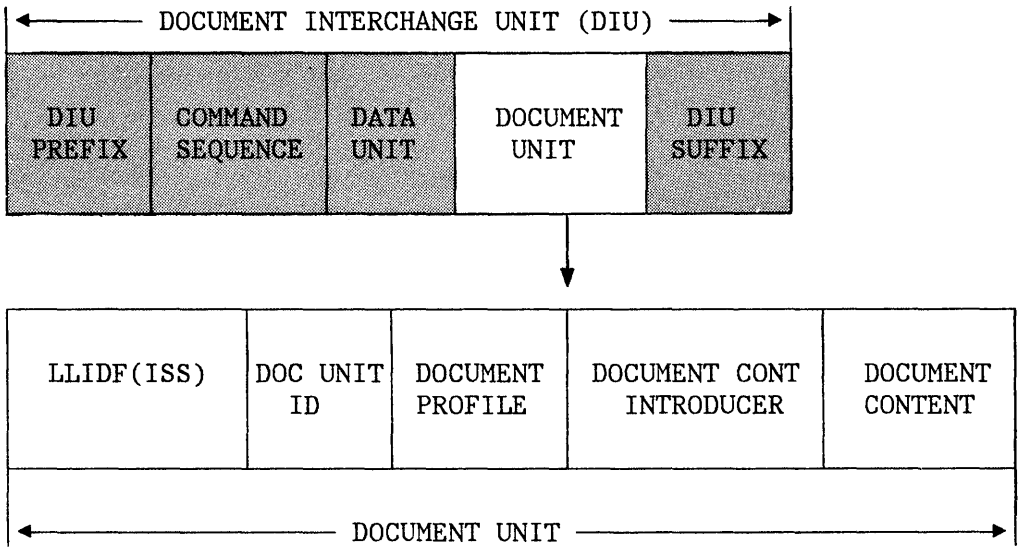


Figure 15. DIA Document Unit Type 3

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 may be found in "Structured Field Segmentation" on page 38.

Document Unit Identifier

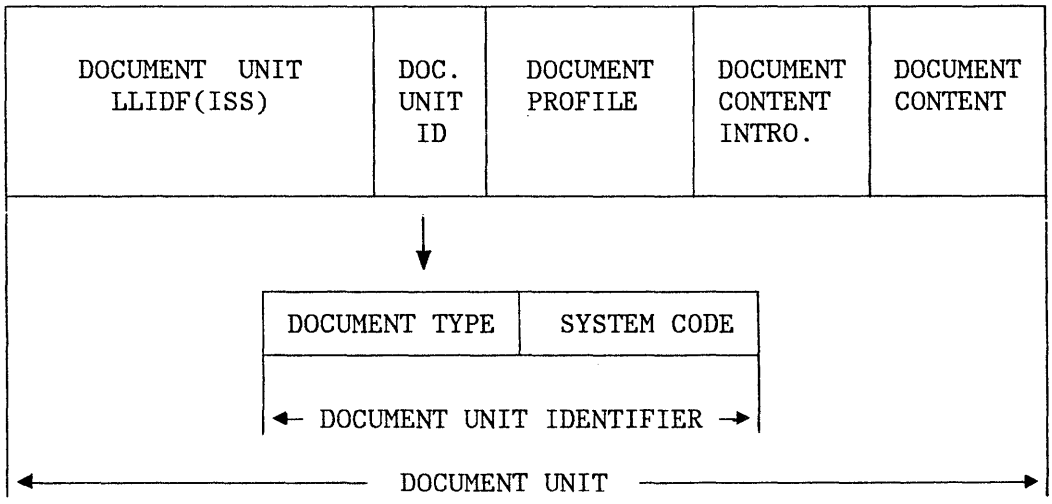


Figure 16. Document Unit Identifier

The document unit identifier for the interchange document unit type 3 is defined as a 15-byte fixed length positional data field that immediately follows the document unit structured field introducer LLIDF(ISS). 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 32,767 are used to specify interchange data stream; for example, the revisable-form-text document, final-form-text document, and so on. Interchange document types are registered by IBM. The interchange data streams are listed in "Appendix C. DIA Document Types" on page 85.

  Document Type field values in the range from 32,768 to 65,535 are used to 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 noninterchange data stream identifiers are controlled by the product identified in the system code parameter and 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, in which case the first 3 characters are IBM; or, it may contain a customer-assigned identifier, which may not 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 takes the form shown in Figure 17 .

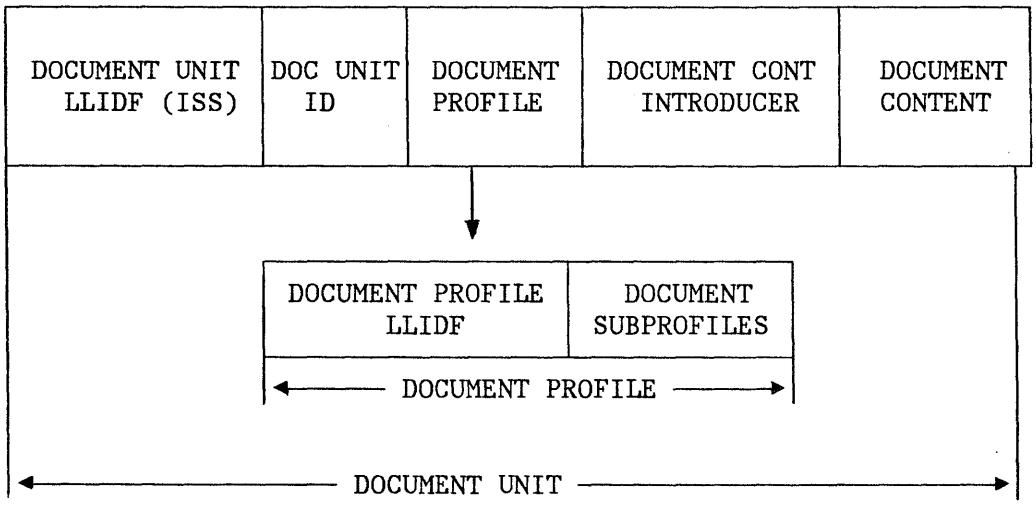


Figure 17. DIU Document Profile

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

The interchange document profile, which has an assigned an 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 Reference.

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.

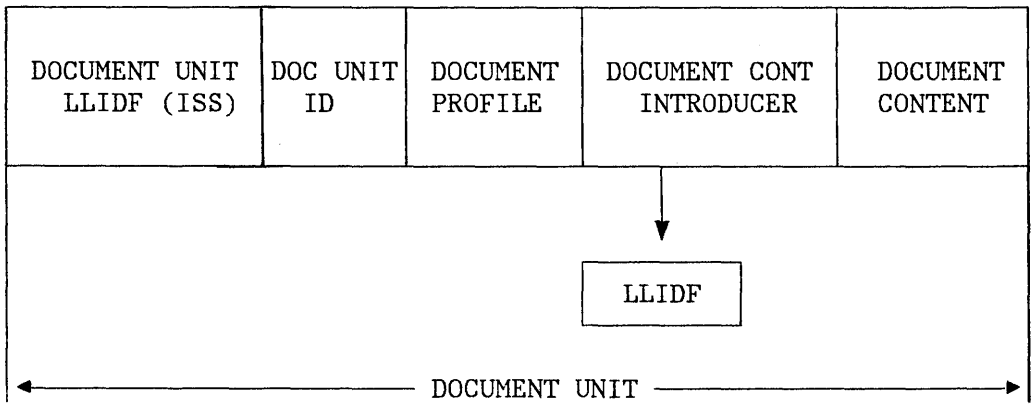


Figure 18. DIU Document Content Introducer

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 introducer 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.

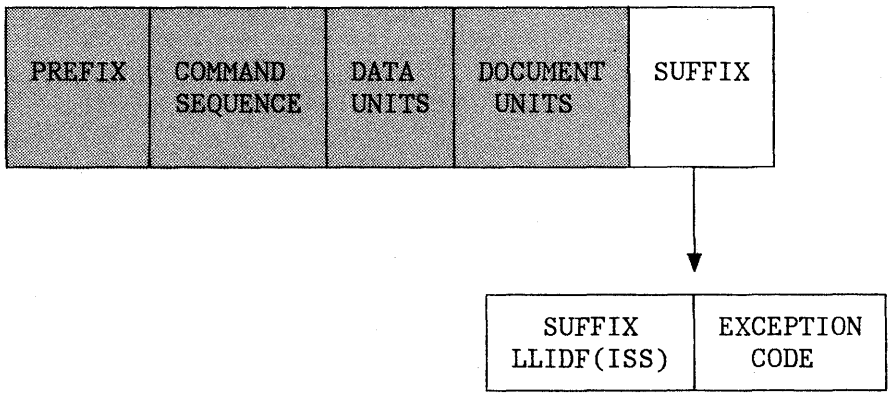


Figure 19. DIU Suffix

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. Two types of suffixes 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 field contains an exception condition code describing the DIU sender detected error which caused the termination. The format of the exception condition code is defined in "Chapter 7. Exception Detection, Classification, and Reporting" on page 63.

STRUCTURED FIELD SEGMENTATION

A structured field may be segmented to accommodate structured fields whose length exceeds 32,767 bytes, DIA processes with limited buffer space, or situations where the length of the structured field cannot be determined before the structured field must be transmitted.

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 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 or only segment or a not last segment through use of the segmentation indicator defined in the I byte of the introducer extension. This technique of segmenting a structured field requires that the last segment be coded with the last or only segment indicator, and that all previous segments are coded not last as illustrated in Figure 20 on page 40 . The introducer extension ISS may 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 entire structured field must be segmented if segmentation is performed at all. The last segment may contain a null data variable.

Segmentation of structured fields within a DIU is performed by the DIA process that generates the structured field. Segmentation of the data variable is independent of the content of the structured field data variable. It is the responsibility of the DIA process receiving the segmented structured field to reconstruct 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)" on page 29.

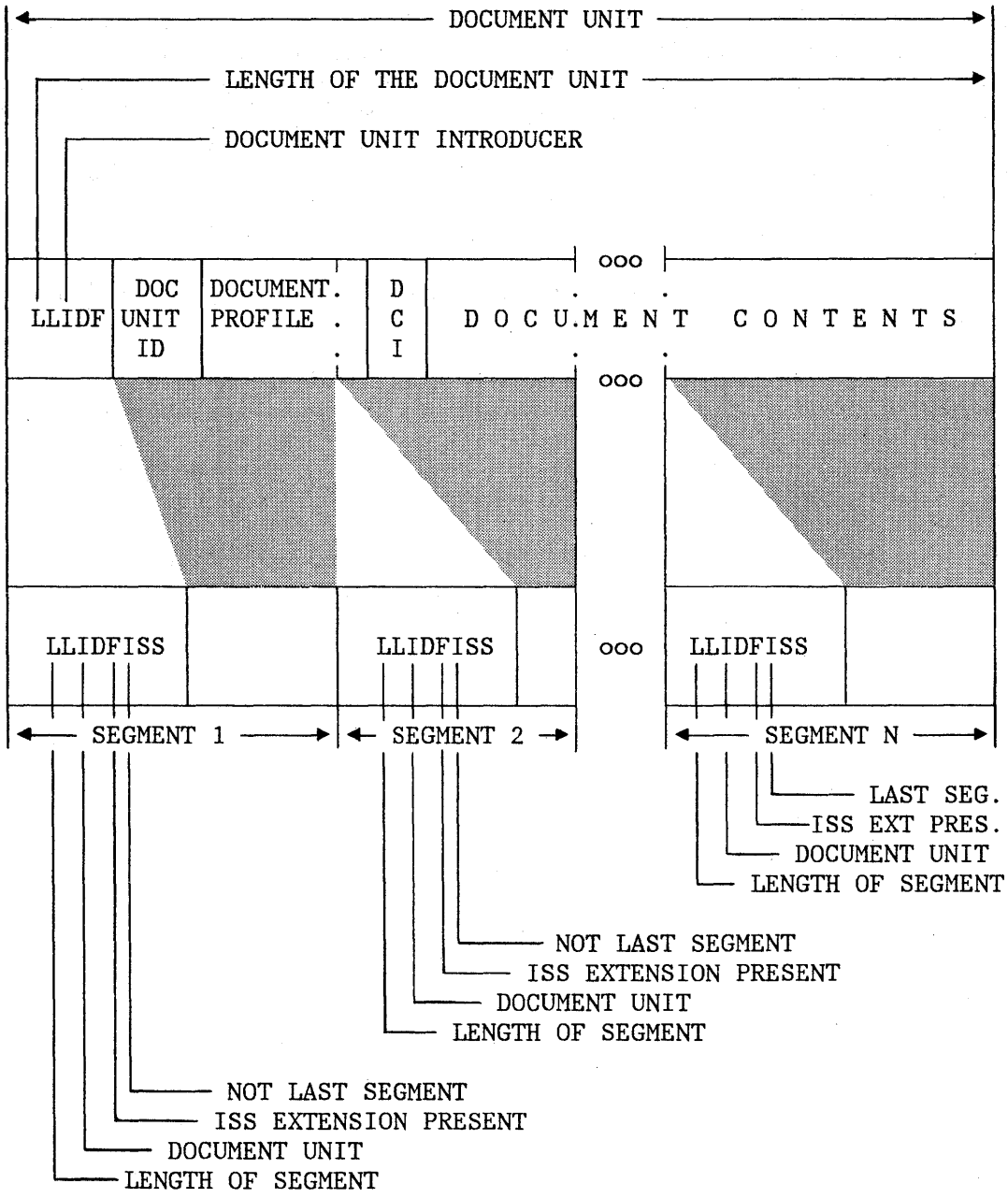


Figure 20. Document Unit Segmentation Example

SUMMARY OF DIU SYNTAX RULES

The following table presents a summary of the general syntax rules for DIU data stream components. Rows 1 and 2 specify the minimum and maximum number of times that each individual DIU data stream component can occur within a single DIU.

Rows 3, 4, 5, and 6 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 7 indicates whether the DIU component may be segmented. DIU subcomponents cannot be segmented.

	PREFIX	COMMAND	DATA UNIT	DOC UNIT	SUFFIX
1 OCCURRENCE MINIMUM	1	1	0	0	1
2 MAXIMUM	1	255	255	255	1
3 ORDER NORMALLY PRECEDED BY	nothing or suffix of last DIU	prefix or another command	last command or another data unit	last command or last data unit or another doc unit	command or data unit or doc unit
4 NORMALLY FOLLOWED BY	command	another command or data unit or doc unit or type 1 suffix	another data unit or doc unit or type 1 suffix	another doc unit or type 1 suffix	nothing or prefix of next DIU
5 EXCEPTION PRECEDED BY	nothing	prefix or another command	last command or another data unit	last command or last data unit or another doc unit	not applic.
6 EXCEPTION FOLLOWED BY	type 2 suffix	type 2 suffix	type 2 suffix	type 2 suffix	not applic.
7 SEGMENTING ALLOWED	no	yes	yes	yes	no

Figure 21. Summary of DIU Syntax Rules

GCID ENCODINGS

The DIU consists of five logical entities: a prefix, a command sequence, optionally data units, optionally document units, and a suffix. These DIU entities and their data fields are coded according to the following standard. All exceptions to these coding definitions are explicitly specified in the descriptions of the DIU data fields.

- The base character set to be used for all Format 1 addresses (1- to 8-bytes) is Character Set A. Character set A (Assembler originated) consists of uppercase alphabetics (A through Z), the numbers (0 through 9), and special characters $, #, @. The base character set support can be expanded to include the characters in Character Set 337 of Code Page 256, including use of Space. See Figure 25 on page 88 for a definition of the valid characters. If the base is expanded, the following rules must be followed by the supporting OSN:

  - The following characters are to be folded to their functional equivalent.

    All alphabetics are to be folded to either uppercase or lowercase. The particular monocase is chosen by the OSN and will be consistent for all alphabetics; that is, if the OSN decides to fold to uppercase, then all alphabetics received will be folded to the appropriate uppercase.

    Space (X'40'), Required Space (X'41'), and Numeric Space (X'E1') are to be folded to one Space character chosen by the OSN. For example, the OSN may choose to fold all Space to X'40'.

    Required Hyphen (X'60') and Syllable Hyphen (X'CA') are to be folded to whichever Hyphen the OSN chooses.

  - Leading Spaces are not allowed in Format 1 addresses, and any trailing Space is not significant. The space is significant only when it is imbedded within the address. This holds for all operands with Format 1 addresses as well as the Search Result Name and the address portion of the Library Assigned Document Name (LADN).

- All DIA passwords are coded using the graphics defined by Character Set 103 of Code Page 256 and the Space control character (X'40'). See Figure 24 on page 87 for a definition of the valid characters. The Space may be included within the 8-byte maximum length of passwords and will be considered significant for DIA validation.

- All Document Names, Distribution Names, and Messages are coded using Character Set 337 of Code Page 256 and the Space control character (X'40'). The Space control character is not allowed as the first or last character of Document Names and Distribution Names. Messages, however, may contain Space in any position.

- All Format 42 Addresses are coded using Character Set 337 of Code Page 256 and the Space control character (X'40'). Format 42 addresses may not have leading Space and any trailing Space is not significant.

- All other DIU entities which do not have explicit GCID coverage are to be coded using Character Set 103 of Code Page 256 and the Space control character (X'40').

CHAPTER 6. SESSION SERVICES

This chapter defines the concepts of a DIA session. It also includes a description of the DIA Session Services commands, and the generalize replying ACKNOWLEDGE command.

DIA SESSION

A DIA session is defined as a logical connection between two DIA processes that can be activated, tailored to perform office system application services, and deactivated.

A DIA session is activated after the DIA processes negotiate and agree upon the set of application services to be performed during the time they are connected. Other functions performed during DIA session activation are:

- Process identification, authorization and validation. DIA processes, acting in the capacity of a source or recipient node, assume the identity of the end user on whose behalf they perform work; therefore, end users are identified, authorized, and validated at DIA session establishment.
- Specification of accounting information.
- Definition of the types of documents that can be received.
- Specification of the number of commands within a command sequence that the processes are capable of receiving during the session.

A DIA session is established after the DIA processes successfully perform the end user identification, authorization, validation, and negotiation of the functions (function sets) to be performed by exchanging SIGN-ON commands. The SIGN-ON command exchange protocols to establish a DIA session are described in "SIGN-ON" on page 46. A definition of DIA Function Set negotiation may be found in "Chapter 4. Function Sets" on page 19.

After a DIA session is established, all work performed (execution of DIA application service function requests) during the session by the DIA processes is done on behalf of the end users they represent.

A DIA session is deactivated by either DIA process sending a SIGN-OFF command. The session termination is unconditional whenever a SIGN-OFF command is sent or received.

When a DIA session is totally within a SNA session, an abnormal termination of the SNA session also terminates the DIA session (a SIGN-OFF command and DIA session reset are implied). The status of DIA commands being sent or received at the time of session termination is as stated in "Chapter 3. Request/Reply Protocols" on page 13. Since it is possible that either a system error or SNA communications subsystem failures can result in the loss of DIUs and the reinitialization of DIA processes, a subsequent SIGN-ON command received from

the same DIA session partner implies a SIGN-OFF command for the previous DIA session if an explicit SIGN-OFF command was not received.

SESSION SERVICES COMMAND DESCRIPTIONS

Each command description in the following sections begins with the command name and a list of the command operands. Optional operands are enclosed within brackets; required operands are shown without brackets. Operands are not repeatable unless explicitly stated in the command description.

The function of the command is explained, followed by a description of each operand. The detailed operand definitions are contained in "Appendix A. Session Services Operands" on page 69. The command structured field IDF's are defined in "Appendix B. Code Points" on page 79.

Each command description contains the request/reply protocol used between the command requestor and command server. Normal and exception condition scenarios are shown.

The command descriptions are concluded with a list of exception conditions which are specific to the command. The general exception conditions that are common to all DIA commands are described in "Appendix E. DIU General Exception Conditions" on page 93.

DIA commands unique to the defined DIA application services are described in detail in related reference manuals, that is, Document Distribution Services, Document Library Services, or Application Processing Services reference manuals.

SIGN-ON

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

The SIGN-ON command is used to request a DIA session with another DIA process.

The SIGN-ON request is used to propose the set of functions (FUNCTION-SET operand) to be used within the DIA session. The proposed function sets may represent either the full DIA receive capability of the process requesting the DIA session or the specific function sets that the process desires to use within this DIA session.

The DIA process receiving the SIGN-ON request examines the proposed FUNCTION-SET options and: 1) establishes a DIA session supporting the requested function sets by returning a SIGN-ON (NRR) replying command whose FUNCTION-SET operand value includes the same FUNCTION-SET-IDs requested, but with the supporting process role(s) specified; or 2) establishes a DIA session supporting a subset of the requested function sets by returning a SIGN-ON (NRR) replying command whose FUNCTION-SET operand value includes a subset of the same FUNCTION-SET-IDs requested, but with the supporting process role(s) specified; or 3) rejects the request to establish a DIA session by returning a negative ACKNOWLEDGE command with the appropriate exception codes.

The receiver of a negotiated SIGN-ON NRR reply may reject the DIA session request by returning a negative ACKNOWLEDGE command with exception codes.

The receipt of a SIGN-ON request within an active DIA session from the same requestor is treated as an implicit request to SIGN-OFF and begin a new DIA session with the specified parameters. For example, a DIA process may wish to 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 roles (requestor and server) to be assumed by each DIA session partner.

COUNT

The COUNT (Format 1) operand, if present, specifies a numeric value from 1 to 255 that indicates the maximum number of commands the sender of the SIGN-ON command containing this operand can receive in a DIU command sequence. If this operand is omitted, a default value of one will be used.

SIGN-ON-ID

The SIGN-ON-ID operand, if present, identifies a DIA end user participating in a DIA session; either SIGN-ON-ID (Format 1) or SIGN-ON-ID (Format 42) may be used. The SIGN-ON-ID operand value is used for identification and for the implementation of access authorization. If the operand is omitted, the value may be defaulted by products to a predefined 1- to 8-byte character name (for example, LU name or default SIGN-ON-ID operand) which is associated with one communications session and which serves the same function as the

SIGN-ON-ID operand. If no default is provided 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 which is sent in the DIA session.

SIGN-ON-PASSWORD

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

The SIGN-ON-PASSWORD operand is required for product implementation that provides password-protected access authorization. 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 which identifies the user accounting information to be used to accrue any charges incurred during the requested DIA session.

The CHARGE-CODE operand is required when the specific product implementation provides user accounting support.

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 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. If the operand is omitted, then documents of any type may be delivered to the requestor during the DIA session. The document type values specified by this operand are applicable to the receive capabilities of the requestor only, and not the requestor's send capability.

When the operand is present, the document type for each document to be delivered will be checked against the specified values. If the document type matches a specified value, then the document will be delivered. If there is no match, but the DIA session partner 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 transforms are not possible or available, then the document will not be sent. Documents queued 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 get them delivered or cancelled.

DIA neither requires nor specifies any mandatory document type transforms to be supported by OSN products. Transforms when supported, however, 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. DIA Document Types" on page 85 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 being established by this sign-on request. If the operand is omitted, then validation of GCID 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.

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. This operand does not apply to the delivery of document types X'0008' and X'000A'.

CORRELATION

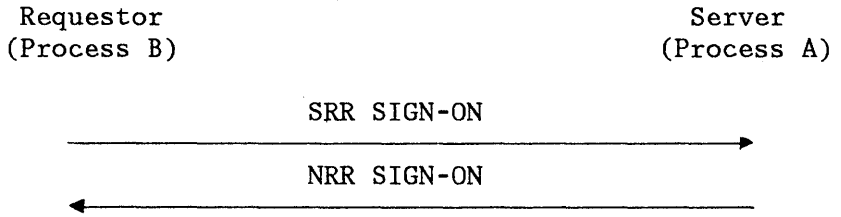
The CORRELATION (Format 1) operand, if present, is used to correlate a replying command to a previously sent request. The CORRELATION (Format 1) operand uniquely identifies the request to which the command is replying and gives an indication of whether or not additional replying commands are to be expected; that is, a last or not-last indicator is returned. When the last replying command has been received, the request is considered complete.

Request/Reply Protocol

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

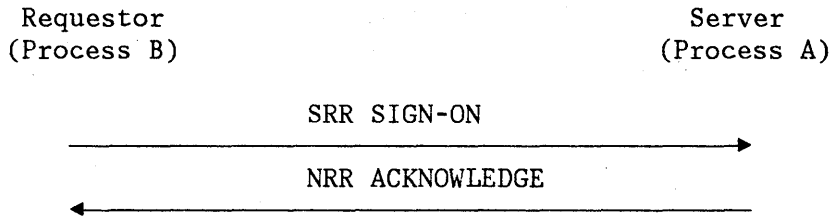
- Scenario 1 - Normal Condition

  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

The general exception conditions that are common to all DIA commands are described in "Appendix E. DIU General Exception Conditions" on page 89. The following exception conditions are specific to the SIGN-ON command and are detected and reported in addition to the general exception conditions.

- 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 value range is exceeded.

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

- The Correlation Reply-Indicator field specifies Not Last.

  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 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 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 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 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 role(s).

  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 support considerations apply, in general, 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. Reinitialization of the DIA process occurs 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 effect 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 which is returned on the replying command:

  The function set identifiers returned on the replying SIGN-ON command must be the same or a subset of the function set ID options presented in the SIGN-ON request command.

  Complementary roles must be returned for each function set selected by the sign-on receiver for use within this DIA session.

  It is invalid for the sign-on receiver to select terminal-to-terminal function sets (FS 6 & 7) for use with any other function set, except Function Set 10, within the same DIA session. If a choice of processing roles is presented in the initial sign-on request, however, the receiver of the request is free to choose which compatible subset of function sets it desires to support for the DIA session being established.

- SIGN-ON-ID

  The return of the SIGN-ON-ID operand on 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 for access authorization and the operand is not present, then the process may elect not to enter into DIA session. If the process receiving the SIGN-ON reply does not support password protection and the operand is present, then it may 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 the process may elect not to enter into 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 may be ignored.

  The support of user accounting procedures is a product decision.

- COUNT

  The COUNT operand should be returned on replying SIGN-ON commands when the replying DIA process is capable of receiving 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 requesting the SIGN-ON command must validate the COUNT operand value only if it proposes to send DIUs containing multiple commands. If the requesting process can not comply with the receive constraints of the replying process, then the reply must be rejected with a negative ACKNOWLEDGE command to terminate the request for DIA session.

- GCID/DOCUMENT TYPE

  These operands may only be sent by process B to process A nodes on SIGN-ON SRR requests or SIGN-ON NRR replies. If the operands appear on replying SIGN-ON commands sent or returned by Role A DIA processes, they may be ignored.

SIGN-OFF

Command	Operands
SIGN-OFF	-none-

The SIGN-OFF command is used unconditionally to terminate a DIA session whenever sent or received.

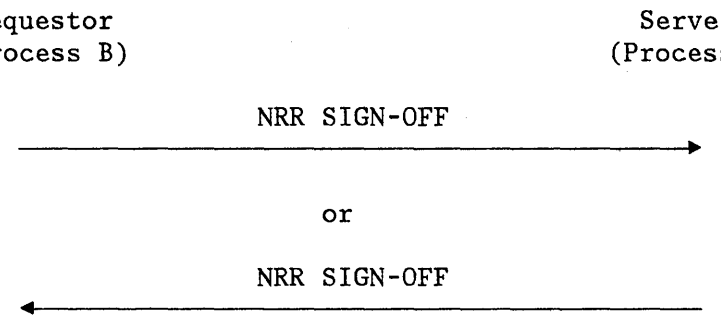
Operand Descriptions

There are no operands defined for this command.

Request/Reply Protocol

Scenario 1 - Normal Condition.

The SIGN-OFF command is an NRR command; therefore, there are no replies expected. This command can be sent by either partner in the DIA session. A SIGN-ON command is the only valid command that can follow a SIGN-OFF.



Exception Conditions

The general exception conditions that are common to all DIA commands are described in "Appendix E. DIU General Exception Conditions" on page 89. There are no specific exception conditions unique to the SIGN-OFF command.

ACKNOWLEDGE

Command	Operands
ACKNOWLEDGE	CORRELATION, EXCEPTION-CODE [,REPLY-DATA] [,RECOVERY-ACTION]

The ACKNOWLEDGE command is used as a generalized replying command to notify the requestor that a previously requested DIA command has been successfully or unsuccessfully completed.

The ACKNOWLEDGE command is used to report exception conditions detected during the processing of any DIA command. In addition to reporting exception conditions, the sender of the ACKNOWLEDGE command may also include a recommended recovery action. The receiver of the ACKNOWLEDGE may 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 requestor may use, but is not required to perform, the recommended recovery action returned on the ACKNOWLEDGE command.

The ACKNOWLEDGE command is also used as a generalized replying command to indicate that the correlated function request completed successfully. In this

case, the capability exists to pass back to the function requestor limit reply information the REPLY-DATA operand, for example, returning the Library Assigned Document Name (LADN) to a FILE command. When the ACKNOWLEDGE replying command is used 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, via the last/not-last replying command indicator, that no further replying commands 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 correlated request. Successful completions are indicated by specifying a X'00' in 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 may be reported include session errors, syntax errors, semantic errors, process errors, and sender errors. The EXCEPTION-CODE operand data variable is defined in "Chapter 7. Exception Detection, Classification, and Reporting" on page 63.

The EXCEPTION-CODE operand is repeatable 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 is to appear 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 is used to return data to the requestor of a command. For example, the REPLY-DATA operand is used on ACKNOWLEDGE commands that are replying to REQUEST-DISTRIBUTION command. Specifically, 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 recovery action is to be determined by the receiver of the ACKNOWLEDGE command. The RECOVER-ACTION operand data variable is defined in

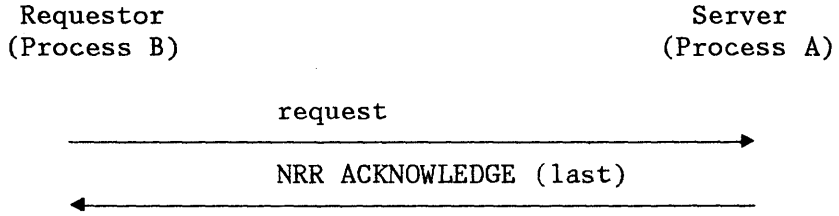
"Chapter 7. Exception Detection, Classification, and Reporting" on page 63.

Request/Reply Protocol

The following scenarios illustrate possible replies to the ACKNOWLEDGE command:

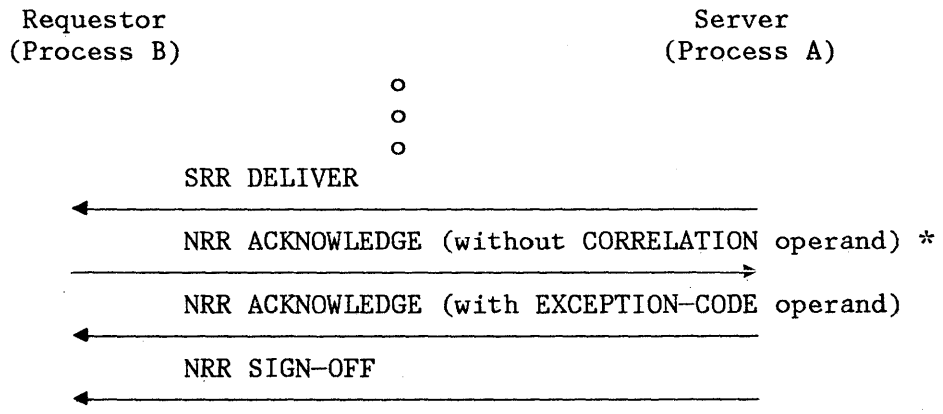
- Scenario 1 - Normal Conditions

  The ACKNOWLEDGE command is used as a replying command to a function request. The ACKNOWLEDGE command may indicate the request was successful or unsuccessful in the EXCEPTION-CODE operand. The ACKNOWLEDGE command always indicates that it is the last replying command.



- 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 condition code (refer to "Exception Condition Classification" on page 63). This ACKNOWLEDGE is followed by a SIGN-OFF command to terminate the DIA session.



* An ACKNOWLEDGE command without a CORRELATION operand is a syntax error.

Exception Conditions

The general exception conditions that are common to all DIA commands are described in "Appendix E. DIU General Exception Conditions" on page 89. The following exception conditions apply to the ACKNOWLEDGE command itself and not to the exceptions that may be 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 expect 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

Command	Operands
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, and the new-value 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 and new-value fields of the CONTROL-VALUE operand must be identical.

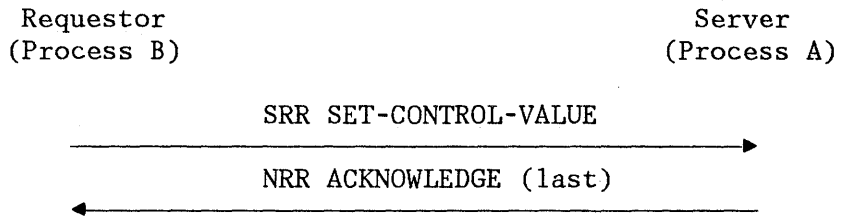
When used to delete a password, the new-value field of the CONTROL-VALUE operand must be null.

Request/Reply Protocol

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

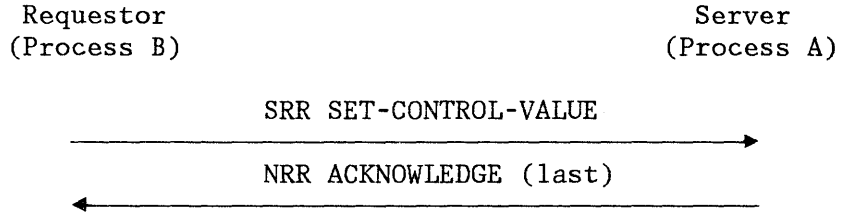
- Scenario 1 - Normal Conditions

  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 during the SET-CONTROL-VALUE command processing will be 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 all DIA commands are described in "Appendix E. DIU General Exception Conditions" on page 93. The following exception conditions are specific to the SET-CONTROL-VALUE command and are detected and reported in addition to the general exception conditions.

- The authorization is value required, but not specified in 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.

- The 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.

- The 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.

- The 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/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 does not equal the operand identifier for the old-value.

  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 the other user's SIGN-ON-PASSWORD, the other user's SIGN-ON-ID operand is required for authorization. To change the other 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 may appear in either order.

CHAPTER 7. 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 for the reliable exchange of information between DIA processes.

To satisfy this objective, the DIA process sending a DIU has the responsibility for insuring that the DIU is precise and correct. The DIA process receiving the DIU is responsible for reporting any detected exception conditions and recommending appropriate recovery actions, if any. The DIA process sending the exception condition DIU is responsible for recovery.

EXCEPTION CONDITION CLASSIFICATION

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

The exception condition classes defined are:

- No-Exception - as the name implies, 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; for example, requesting an application service outside of the negotiated functions sets.
- Syntax - this exception condition class is used to report violations of DIU syntax rules; for example, omitting a required operand for a DIA command.
- Semantic - this exception condition class is used to report conflicting parameters; for example, specifying an incorrect password.
- Process - this exception condition class is used to report exception conditions detected during function request processing; for example, insufficient resources to complete the requested function.
- Sender - this exception condition is used by the DIU sender that is unable to complete transmission of the DIU; for example, permanent disk I/O error when reading a document that is being delivered in a DIU.

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

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

The condition code defines the specific condition detected; for example, Function Not Supported, Unauthorized Access, Data Not Found, and so forth. A complete list of condition codes may be found in the tables that follow.

The exception condition object defines the DIA object that was incorrect; for example, an operand value (such as password) was incorrect, a command is not supported, and so forth. A complete list of exception condition objects may be found in the tables that follow.

The exception condition data field is used to report the object and object value that caused the exception condition. For example, the exception condition data field would contain the operand introducer LLIDF and its data variable (for example, the PASSWORD operand) that was incorrect.

EXCEPTION CONDITION REPORTING

These exception conditions are reported using either the generalized ACKNOWLEDGE replying command or, for 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, a Suffix Type 2 Exception Condition Code or EXCEPTION-CODE operand, the structured field data variable format is the same. The data variable format is defined below in Figure 22 on page 65.

When the Exception Condition Class field specifies X'00', then there was no exception condition detected, and the other operand fields may be ignored.

When the Condition Code field indicates a user-specified condition, X'00', the exception condition is defined as outside the scope of the DIU operation. Exception conditions with this code will be passed to the application process for interpretation and disposition.

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

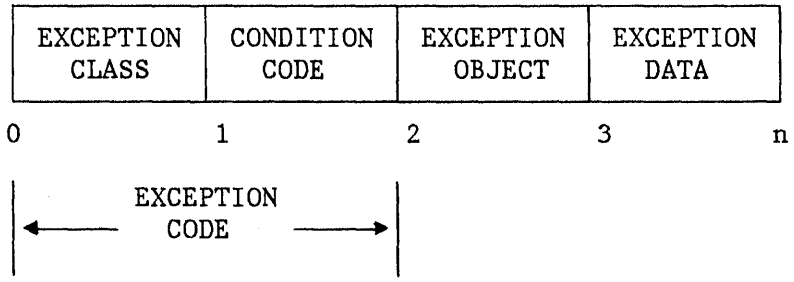


Figure 22. Exception Condition Code Format

The exception condition class byte has the following definition. The severity of the exception condition is contained in the 2 high-order bits of the exception condition class and is defined following the class values.

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

The severity of the exception condition is encoded in the 2 high-order bits of the exception condition class byte. The definition of these bits are:

Value	Severity
00..	Information - request processed to normal conclusion
01..	Warning - request result may be incorrect
10..	Severe - request concluded with exception
11..	Catastrophic - request not processed

The Condition Code byte has the following definition:

Value	Condition	Value	Condition
X'00'	User-specified condition	X'0D'	Reserved
X'01'	Function not supported	X'0E'	Format invalid
X'02'	Data not supported	X'0F'	Length invalid
X'03'	Unauthorized access	X'10'	Indicator invalid
X'04'	Resource not available	X'11'	Range exceeded
X'05'	Password invalid	X'12'	Intervention required
X'06'	Execution terminated	X'13'	Time-Out
X'07'	Data not found	X'14'	Cancelled
X'08'	Segmentation	X'15'	Subfield Length invalid
X'09'	Reserved	X'16'	Subfield Type invalid
X'0A'	Sequence	X'17'-'FF'	Reserved
X'0B'	I/O Error		
X'0C'	ID invalid		

The exception condition object byte has the following definition:

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

The Exception Condition Data field is a variable length byte string of 5 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 may recommend a form of recovery action to the sender of that DIU. This recommendation is carried on the ACKNOWLEDGE command in the RECOVERY-ACTION operand. The receiver of the RECOVERY-ACTION operand is not bound to the recommended action. If the recommended exception condition action is not done, the subsequent recovery may be unacceptable to the sender of the exception condition action and, therefore, cause the session to be terminated. The recovery actions that may be recommended in the RECOVERY-ACTION operand and their encoding are as follows:

Value	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 resend that DIU as the next DIU sent after receiving the ACKNOWLEDGE command containing this value.
X'02'	Skip and resend. The sender should resend the offending DIU after all other DIUs scheduled for reply to the current request have been sent.
X'03'	Postpone. The offending DIU should not be resent until requested on this or a subsequent DIA session.
X'04'	Cancel. The offending DIU should never be resent.
X'05'	Terminate the exchange. Stop sending the offending DIU and terminate the command that requested it. Only send the offending DIU or any other scheduled DIUs after a subsequent command is issued to request them.

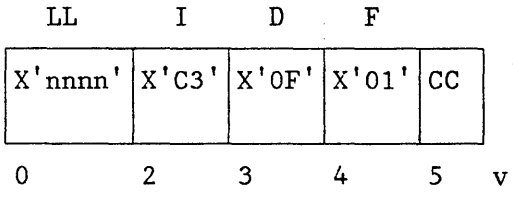
If the RECOVERY-ACTION operand does not appear on an ACKNOWLEDGE command that has a non-zero exception condition class, the recovery action defaults to be the same as if it were specified as X'00', none. Recovery action values are ignored on a normal ACKNOWLEDGE command.

APPENDIX A. SESSION SERVICES OPERANDS

This section contains a detailed discussion of each operand relevant to the DIA Session Services. Each discussion includes an illustration of the operand structure.

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.

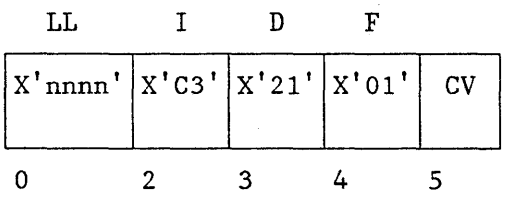


The CC operand value is a variable length character string.

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:



The fields contained in the CV operand and their descriptions follow:

FIELD	LENGTH	CONTENTS
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

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, new, 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:

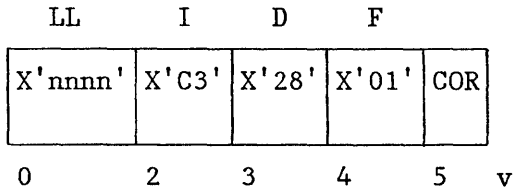
OPERAND NAME	IDENTIFIER	AUTHORIZATION VALUE
SIGN-ON-PASSWORD	X'C338'	none (see note)
DOCUMENT-PASSWORD	X'C32E'	none (see note)

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 may 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:



The COR operand value has the following format:

FIELD	LENGTH	VALUE
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 DIU command sequence in which the requesting command was received.

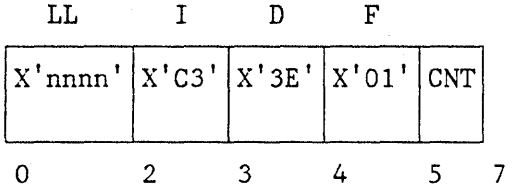
The DIU-ID field matches the DIU-ID field of the DIU Prefix in which the requesting command was received.

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.

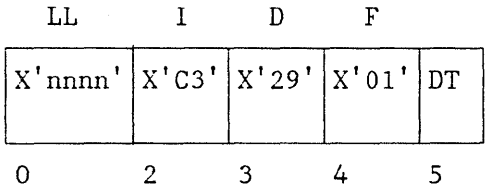
The operand is defined as follows:



The CNT operand value is a 2-byte binary number.

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.



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.

FIELD	LENGTH	CONTENTS
Doc.Type ID	2	2-byte binary number specifying a document type as defined in "Appendix C. DIA Document Types" on page 85.

The use of this operand is restricted to SIGN-ON commands that are sent from process B nodes to process A nodes. The values specified by this operand are applicable to the receive capabilities of the node only, and not to that node's send capability.

The appearance of this operand is optional for individual function sets as defined in "Chapter 4. Function Sets" on page 19. If the operand is omitted, then documents of any type may 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 transforms are not possible or available, then the document will not be sent. Documents queued 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 get them delivered or cancelled.

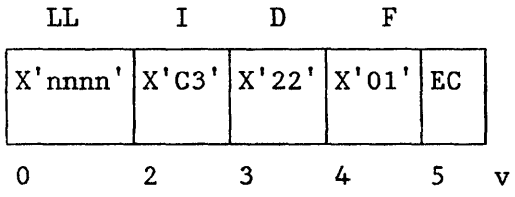
The processing exceptions which can arise during a session from the use of this option are defined in the individual command descriptions for each command requesting delivery of documents.

DIA neither requires nor specifies any mandatory document type transforms to be supported by OSN products. Transforms when supported, however, 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:

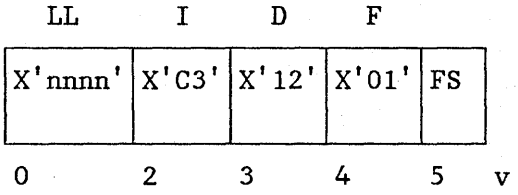


For EC operand values refer to "Exception Condition Classification" on page 63.

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.

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.



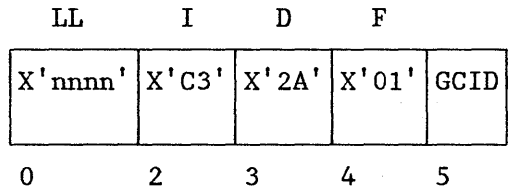
The FUNCTION-SET operand value consists of a vector of one or more 3-byte function set identifiers, each containing a 1-byte role identifier and a 2-byte function set identifier value. The function set identifiers specify which sets of DIA facilities will be used during the DIA session being established. The function sets are described in "Chapter 4. Function Sets" on page 19.

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.

FIELD	LENGTH	CONTENTS
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 4. Function Sets" on page 19.

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.



The GCID 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.

FIELD	LENGTH	CONTENTS
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 4. Function Sets" on page 19. If the operand is omitted, then validation of GCID 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 GCID operand is present in the SIGN-ON command, documents will be checked for GCID 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 GCID usage is known to be consistent with the specified values, then the document will be delivered. If the GCID usage within a valid document type is not consistent with the GCIDs 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 GCID operand.

It should be noted that validity checking for GCID usage of non-DIA defined documents requires knowledge of Document Content Architectures which are outside the domain of DIA. If the data contents of the document are not known, or the OSN is not capable of providing validity checks on document contents, then the document is not delivered in this DIA session.

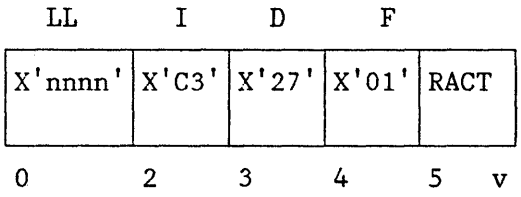
Documents queued for delivery to the signed-on recipient that cannot be delivered in the session will be held in the distribution queue until some action is taken to get them delivered or cancelled.

The processing exceptions which can arise from the use of this session option are defined in the individual command description for each command requesting delivery of documents.

This operand does not apply to the delivery of types X'0008' and X'000A'.

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:

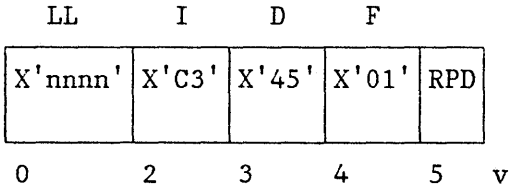


For RACT operand values refer to "Exception Condition Classification" on page 63. If the RECOVERY ACTION operand does not appear on the ACKNOWLEDGE command, the recovery action is to be determined by the receiver of the ACKNOWLEDGE command.

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:

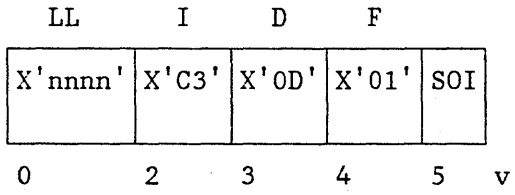


The RPD operand value is a variable length field containing the reply data required by the command being replied to by the command containing this operand.

For example, the REPLY-DATA operand will be used for ACKNOWLEDGE commands that are replying to REQUEST-DISTRIBUTION. The REPLY-DATA will contain 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 and is defined in Document Distribution Services Reference.

SIGN-ON-ID (FORMAT 1)

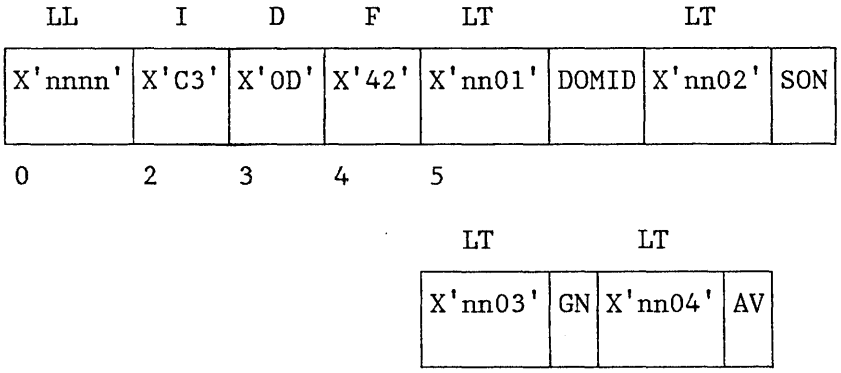
The SIGN-ON-ID (Format 1) operand identifies a DIA end user participating in a DIA session.



The SOI 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.

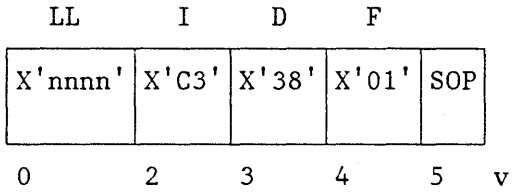
When format 42 of the SIGN-ON-ID operand is used, it must contain one of the following combinations of the individual operand parts. All parts specified for a given combination must be present.

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.



The SOP operand value is a 1- to 8-byte character string. The maximum length for this operand is 8 bytes.

APPENDIX B. CODE POINTS

This appendix consists of tables which list the code points defined for the DIA constructs. Each table gives the description of the construct and the code point which 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'	4
OPERAND - DATA UNIT REFERENCE	X'C4'	5
OPERAND - DOCUMENT UNIT REFERENCE	X'C5'	6
DOCUMENT UNIT	X'C9'	7
DOCUMENT PROFILE	X'CA'	8
DOCUMENT CONTENT INTRODUCER	X'CB'	9
COMMAND - SYNC REPLY REQUIRED	X'CD'	3
DIU SUFFIX	X'CF'	10

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.

Format is always X'x0' when there is no value field; that is, LL = X'05' or X'08'.

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 - SIGN-ON	C10C	x1
NRR - SIGN-OFF	C10D	x1
NRR - DELIVER	C119	x1
NRR - STATUS-LIST	C11E	x1

TABLE 3. SYNCHRONOUS REPLY REQUIRED COMMAND ENCODINGS

CODE POINT DESCRIPTION	GDS ID	F
DIU COMMAND - SYNCHRONOUS REPLY RQRD (SRR)	CD	
SRR - FILE	CD02	x1
SRR - DELETE	CD03	x1
SRR - RETRIEVE	CD04	x1
SRR - SEARCH	CD06	x1
SRR - FORMAT	CD0A	x1
SRR - EXECUTE	CD0B	x1
SRR - SIGN-ON	CD0C	x1
SRR - CANCEL-DISTRIBUTION	CD10	x1
SRR - MODIFY	CD12	x1
SRR - LIST	CD13	x1
SRR - OBTAIN	CD17	x1
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

TABLE 4. 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-PROFILE-DATA	C30B	01
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 MODIFY-CONTROL-DATA	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 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	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

TABLE 4. IMMEDIATE DATA OPERAND ENCODINGS (Cont.)

CODE POINT DESCRIPTION	GDS ID	F
IMMEDIATE DATA OPERAND (IMMED DATA OPND)	C3	
IMMED DATA OPND DESTINATION-NODE-ADDRESS	C32F	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
IMMED DATA OPND DISTRIBUTION-NAME	C341	01
IMMED DATA OPND REPLY-DATA	C345	01

TABLE 5. 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 GDC 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 6. 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 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
NOTE: DOCUMENT UNIT CODES X'C980'-X'C9FF' USER ASSIGNED		

TABLE 8. DOCUMENT PROFILE ENCODINGS

CODE POINT DESCRIPTION	GDS ID	T
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
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. 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'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'

Figure 23. Document Type Code Assignments

APPENDIX D. GRAPHIC CHARACTER SETS

Column		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	→		00				01				10				11		
Row	Bit Pattern ↓	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	: SP13								
B	1011					. SP11	$ SC03	, SP08	# SM01								
C	1100					< SA03	* SM04	% SM02	@ SM05								
D	1101					(SP06	) SP07	— SP09	' SP05								
E	1110					+ SA01	; SP14	> SA05	= SA04								
F	1111					¦ SP02	^ SD15	? SP15	" SP04								

Figure 24. Character Set 103 of Code Page 256

Column		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	→		00				01				10				11		
Row	Bit Pattern ↓	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0	0000					SP01	& SM03	– SP10	ø LO61	Ø LO62	° SM19	µ SM17	¢ SC04	{ SM11	} SM14	\ SM07	0 ND10
1	0001					SP30	é LE11	/ SP12	É LE12	a LA01	j LJ01	~ SD19	£ SC02	A LA02	J LJ02	SP31	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	ƒ 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	ß 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	a SM21	¡ SP03	¬ SM66	– SP32	1 LI61	2 NS02	3 NS03
B	1011					• SP11	$ SC03	, SP08	# SM01	» SP18	o SM20	¿ SP16	| SM13	ô LO15	û LU15	Ô LO16	Û LU16
C	1100					< SA03	* SM04	% SM02	@ SM05	ð LD63	æ LA51	Ð LD62	- SM15	ö LO17	ü LU17	Ö LO18	Ü LU18
D	1101					(SP06	) SP07	— SP09	' SP05	ý LY11	ª SD41	Ý 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	ú LU11	Ó LO12	Ú LU12
F	1111					! SP02	^ SD15	? SP15	" SP04	± SA02	¤ SC01	® SM53	¯ SM10	õ LO19	ý LY17	Õ LO20	SS99

Figure 25. 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	å	a Overcircle Small
LA28	Å	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 26 (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 26 (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	ƒ	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	º	Ordinal Indicator, Masculine
SM21	ª	Ordinal Indicator, Feminine
SM24	§	Section Symbor (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 26 (Part 3 of 3). Description of Code Page 256 Graphics.

APPENDIX E. DIU GENERAL EXCEPTION CONDITIONS

The general exception conditions common to most of the DIA commands are defined in this section. 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 may be 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 is 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 requestor.

Exception = Warning, Syntax, Function-Not-Supported, Command-Operand
Exception Code = X'420108'
Exception data = LLIDF and data of the command operand

- A command received that 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 that 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 that is an invalid length.

  Exception = Catastrophic, Syntax, Length-Invalid, Operand-Value
  Exception Code = X'C20F09'
  Exception data = LLIDF of invalid operand format

- 'First UOR Segment' indicated for a command sequence.

  Exception = Catastrophic, Syntax, Indicator-Invalid, Command
  Exception Code = X'C21007'
  Exception data = LLIDF and introducer extension of command

- Command Sequence Introducer extension contains a sequence number which is invalid.

  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 that is not supported.

  Exception = Catastrophic, Semantic, Data-Not-Supported, Operand-Value
  Exception Code = X'C30209'

- 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'

- 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'

- 'First UOR Segment' indicated for a document unit.

  Exception = Catastrophic, Syntax, Indicator-Invalid, Document-Unit
  Exception Code = X'C2100C'
  Exception data = LLIDF and introducer extension of document unit

- 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'

- 'First UOR Segment' indicated for a data unit.

  Exception = Catastrophic, Syntax, Indicator-Invalid, Data-Unit
  Exception Code = X'C2100A'
  Exception data = LLIDF and introducer extension of data unit

- 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 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

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.

COD. Confirmation-of-delivery.

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.

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. Document interchange unit.

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.

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 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 a 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.

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.

ID. That part of the structured field introducer that defines the class and type of the structured field.

IDP. Interchange document profile.

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.

ISS. Introducer extension.

LADN. Library assigned document name.

library assigned document name (LADN). A unique name assigned to documents filed in the document library.

message. A collection of information transmitted from one point to another.

No reply required (NRR). A command class used when the function requested does not require a reply.

NRR. No reply required.

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 designated as secondary owner by the primary owner of the document in the document library.

password. A character 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 criterion 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.

source. An end user that requests 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.

user. See end user.

INDEX

A

ACKNOWLEDGE command 15, 16, 20, 47, 49, 55-58, 59, 64, 67, 73, 76
  description 55
  exception conditions 58
    process 58
    semantic 58
    session 58
    syntax 58
  operands 56
    CORRELATION 56
    EXCEPTION-CODE 56
    RECOVERY-ACTION 56
    REPLY-DATA 56
  protocols 57
    exception condition 57
    normal condition 57
Application Processing Services 3, 11, 13, 25, 32, 46
architecture
  See Document Content Architecture
  See Document Interchange Architecture

C

CANCEL-DISTRIBUTION command 20
character set ID
  See GCID
CHARGE-CODE (Format 1) operand 69
COD
  See confirmation-of-delivery
code page
  See GCID
confirmation-of-delivery 8, 9, 10
CONTROL-VALUE (Format 1) operand 69-70
CORRELATION (Format 1) operand 14, 16, 71
COUNT (Format 1) operand 72

D

DCA
  See Document Content Architecture
DELETE command 24
DELIVER command 14, 15, 20, 24, 25, 32, 76
destination node address 6
DIA
  See Document Interchange Architecture
DIA Code Points 79
DIA command 14
  correlation of 14, 31
  general protocol rules 16
DIA command class 13, 14
  asynchronous reply required (ARR) 14, 16, 32
  no reply required (NRR) 14, 16, 32
  synchronous reply required (SRR) 14, 16, 32
DIA Document Types 85
DIA session 6, 8, 9, 16, 19, 45, 47, 52, 53, 57, 76, 77, 78
DIA Session Services 6, 45, 69
distribution document name 8, 10, 76
distribution list 8, 10
distribution priority 8
DIU command sequence 31
  command 32
  operand 33
DIU data unit 34
DIU document unit 34
  document content introducer 34, 37
  document profile 34, 36
  identifier 34
    document type 35
    system code 36
DIU identifier 14, 31
DIU Parsing Rules 40
DIU prefix 14, 31
DIU structured field 27
  extension (ISS) 27
  identifier (IDF) 27
DIU suffix 38

Document Content Architecture 1, 2, 75
document descriptors 3
Document Distribution Services 3, 8, 21, 32, 46
Document Interchange Architecture 1, 2, 3, 5, 63
  See also Application Processing Services
  See also Document Distribution Services
  See also Document Interchange Architecture, services defined by
  See also Document Library Services
  basic concepts of 5
  benefits of 3
  command 5
  goals of 4
  process 5, 16, 38
  relationship to SNA 1
  services defined by 5, 6
document interchange unit (DIU) 14, 16, 27, 63, 67, 73, 93
  component
    command sequence 42
    data unit 42
    document unit 42
    prefix 42
    suffix 42
  components 27, 30
    command sequence 27, 30, 31
    data unit 27, 30, 34
    document unit 27, 30, 34
    prefix 27, 30, 31
    suffix 27, 30, 38
  flow between processes 16
  identifier 14
  prefix 14
  structure of 30
document library 3, 6, 7
Document Library Services 3, 7, 24, 25, 32, 46
document profile 7, 8, 11, 27, 34, 36
DOCUMENT-TYPE (Format 1) operand 72-73
document, definition of 1, 5

E

end user 5, 6, 9
exception condition 63
  class values 65
  classification 63
    no-exception 63
    process 63
    semantic 63
    sender 63
    session 63
    syntax 63
  code format 65
  condition code field 64, 66
    values 66
  data field 64
  detection 63
  object field 63, 64, 66
    values 66
  reporting 64
  severity 64, 65
    catastrophic 64
    information 64
    severe 64
    warning 64
  severity values 65
exception conditions, general 93-98
  command 94
  data unit 97
  document unit 96
  prefix 93
  suffix 97
EXCEPTION-CODE (Format 1) operand 73
EXECUTE command 25

F

FILE command 24, 32
final-form-text 3, 11, 36
FORMAT command 25
function sets 6, 19-26, 45
  negotiation of 19
  reason for 6
FUNCTION-SET (Format 1) operand 73-74

G

GCID 42, 49, 74, 75
  definition 42
GRAPHIC-CHARACTER-SET-ID
 (Format 1) operand 74-75

I

interchange document profile
 (IDP) 37
introducer 28, 29, 30, 36, 37,
 38, 64, 93
introducer extension 28, 29, 30,
 31, 32, 34, 35, 38

L

LADN 7, 42, 55
  description
Library Assigned Document Name
  See LADN
library, document 3, 6, 7
LIST command 20
LLIDF
  See introducer
LLIDF(ISS)
  See introducer extension
LLIDFISS
  See introducer extension

M

MODIFY command 25

O

OBTAIN command 20
office information
 architectures 1
office system network 1, 3, 5, 6,
 8
  logical components of 5
office system node (OSN) 6, 7, 9,
 11, 20, 42, 72, 75
  destination OSN functions 10
  end user function 6
  functions 7, 11
  originating OSN functions 10
operand class 33
  data unit reference 33
  document unit reference 33
  immediate data 33
operand type 33
originating node address 6

P

PROCESS-BIT-STRING command 21
publications, related iii

R

recipient 5
recipient address 6
recipient node 5, 8, 9, 20
  end user function 5
  functions 9
recovery actions 67
RECOVERY-ACTION (Format 1)
 operand 75-76
related publications iii
REPLY-DATA (Format 1) operand 76
REQUEST-DISTRIBUTION command 21,
 76
RETRIEVE command 13, 14, 15, 24
revisable-form-text 3, 11, 36

S

SEARCH command 24
segmentation 29, 34, 38
SET-CONTROL-VALUE command 20,
 59-61
  description 59
  exception conditions 60
    semantic 60-61
    syntax 60
  operand 59
    CONTROL-VALUE 59
  protocols 59
    exception condition 60
    normal condition 59
  support considerations 61
SIGN-OFF command 19, 45, 47,
 54-55, 57
  description 54
  exception conditions 55
  protocols 54
SIGN-ON command 16, 19, 45,
 46-54, 72

description 46
exception conditions 50
  process 52
  semantic 51-52
  session 50
  support considerations 52
  syntax 50-51
operands 47
  CHARGE-CODE 48
  CORRELATION 49
  COUNT 47
  DOCUMENT-TYPE 48
  FUNCTION-SET 47
  GRAPHIC-CHARACTER-SET-ID 49
  SIGN-ON-ID 47
  SIGN-ON-PASSWORD 48
protocols 49
  exception condition 49
  normal condition 49
SIGN-ON-ID (Format 1) operand 76
SIGN-ON-ID (Format 42) operand 77
SIGN-ON-PASSWORD (Format 1)
 operand 78
source 5
source address 6
source node 5, 8, 9, 11, 20
  end user function 5
  function 9
  functions 7, 11
STATUS-LIST command 21

U

user-written programs 3

Document Interchange Architecture:
Concepts and Structures
Order No. SC23-0759-0

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:

_____

_____

_____

_____

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

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.)

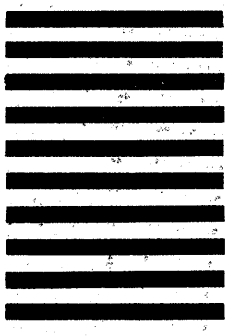
SC23-0759-0

Reader's Comment Form

Cut or Fold Along Line

Fold and tape **Please Do Not Staple** Fold and tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department 506, Building 998
11400 Burnet Rd.
Austin, Texas 78758

Fold and tape **Please Do Not Staple** Fold and tape

IBM®

Document Interchange Architecture: Concepts and Structures Printed in U.S.A. SC23-0759-0

SC23-0759-0

Document Interchange Architecture: Concepts and Structures Printed in U.S.A. SC23-0759-0