IBM

Office Information Architectures: Concepts

GC23-0765-0

Office Information Architectures: Concepts

GC23-0765-0

First Edition (March 1983)

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

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

It is possible that this material may contain references to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such products, programming, or services in your country.

Publications are not stocked at the address 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 506, Building 998, 11400 Burnet Road, Austin, Texas 78758. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Preface

This publication introduces Document Content Architecture and Document Interchange Architecture to those who need to learn the concepts and benefits of these architectures and their purpose in office system networks.[1] This is the basic publication about these architectures; it is intended for managers, system designers, and others involved in making decisions about planning or implementing office system networks.

This book introduces the following architectures:

- Document Content Architecture (DCA)
  - Revisable-Form-Text DCA
  - Final-Form-Text DCA
- Document Interchange Architecture (DIA)

The relationship of these architectures to each other and to the presentation and transport services of Systems Network Architecture are explained. (Systems Network Architecture is introduced in *Systems Network Architecture: Concepts and Products*, GC30-3072.)

This publication is not a primer on electronic office systems. Although no specific prerequisite reading is suggested, readers of this book are assumed to be somewhat familiar with the purposes and capabilities of office systems.

This publication has four chapters.

- Chapter 1 introduces the concepts of an office system network and addresses the uses and benefits of the architectures as a design basis for IBM office systems.
- Chapter 2 provides an overview of the concepts and structure of Document Content Architecture.
- Chapter 3 provides an overview of the concepts and structure of Document Interchange Architecture.
- Chapter 4 provides additional detail about each architecture.

Related Publications

The following publications provide more detailed information about Document Interchange Architecture and Document Content Architecture:

- *Document Interchange Architecture: Concepts and Structures*, SC23-0759.
- *Document Interchange Architecture: Document Library Services Reference*, SC23-0760.

---

1 An architecture is a set of design principles that define the relationships of and interactions between various parts of an information-handling system.

- *Document Interchange Architecture: Application Processing Services Reference*, SC23-0761.
- *Document Interchange Architecture: Document Distribution Services Reference*, SC23-0762.
- *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.

The following publications describe Systems Network Architecture:

- *Systems Network Architecture: Concepts and Products*, GC30-3072.
- *Systems Network Architecture: Technical Overview*, GC30-3073.

Contents

Figures

Chapter 1. Introduction to 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 through the network and, further, define the rules governing the use of the information among the systems of the network.

This chapter identifies some of the main activities of today's offices that are related to document preparation, storage, and distribution and tells how the electronic processing capabilities of IBM office systems can make these activities more efficient. It then explains how the office information architectures help office systems to interchange documents through an office system 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. This publication introduces Document Interchange Architecture (DIA) and Document Content Architecture (DCA) that are used in the design of IBM office systems. These architectures are collectively called *office information architectures*.

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*, GC30-3072.

The two types of Document Content Architecture introduced by this book are (1) Revisable-Form-Text DCA and (2) Final-Form-Text DCA. They define uniform formats for data streams[1] that are interchanged through an office system network. Each DCA defines data formats that are compatible among dissimilar office systems. This compatibility allows all office systems whose design is based on these architectures to have an identical understanding of the data streams interchanged.

Throughout this publication, references to the Document Interchange Architecture and Document Content Architecture apply to the capabilities of the IBM office systems designed in accordance with these architectures.

Document-Related Activities in Today's Offices

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:

---

[1] A continuous stream of data elements being transmitted, or intended for transmission, in character or binary-digit form, using a defined format

- *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 increasing numbers of organizations. Office automation is helping these organizations to improve the productivity and effectiveness of office workers and to improve 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. Some examples of office automation capabilities and advances are as follows.

Revising documents: The ability to substantially revise existing documents without having to retype the entire contents improves the productivity of office workers engaged in that activity.

Formatting documents: The user has much control over how a document is formatted—that is, how the text is arranged on its pages. For example, the user can specify, independently of the content, a document's page size and margin areas. The user can specify and respecify the document format without altering the text.

Printing documents: Because documents can be typed and revised using display devices, the user needs to print the document only after being assured that its content and format are satisfactory. The need for successive draft copies is often greatly reduced. The office system can usually print the document while it performs other office functions.

Distributing documents: Documents can be distributed electronically through office system networks instead of through the mail. By avoiding mail delays, electronic distribution can greatly lessen the delivery time, sometimes by days or weeks.

Filing and retrieving documents: Documents can be filed electronically in, and quickly retrieved from, a central library. The library may be distant from where the document originated or is to be used.

Processing documents: Users can request processing services for documents that are electronically stored either locally in their own office system or in a central library elsewhere in the office system network. For example, a user can create a

document, then send it to a central library, and later request that it be formatted at that location and electronically distributed. Or a user can invoke library searches to identify documents meeting certain criteria (for example, author), then print those documents at the library location or retrieve them for printing or viewing.

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

Office Information Interchange

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 understood 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 extendible 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 Content Architecture and Document Interchange Architecture.

Document Content Architecture

Document Content Architecture describes the form and meaning of the content of a document that office systems can interchange through a network. The text of a document can be in one of two forms: *revisable* and *final.*

A document whose text is in revisable form can have its content and format modified by each person to whom it is distributed or by whom it is obtained from a library. Conversely, a document whose text is in final form is intended for presentation on a printer or display screen rather than for subsequent modification.

Revisable-Form-Text Document Content Architecture specifies how IBM office systems interchange documents that are in revisable form. This architecture defines the structure of the data streams that represent revisable-form-text documents within the office system or network. Besides the text of a document, a

data stream includes fields containing general formatting specifications[2] of the entire document or parts of it. The architecture also specifies the structure of the formatting control codes[3] and text within revisable-form-text documents and prescribes how office systems must interpret them.

Final-Form-Text Document Content Architecture specifies how IBM office systems interchange formatted text documents. Like Revisable-Form-Text Document Content Architecture, it prescribes the structure of the data streams that represent documents within the office system or network.

Unlike the revisable-form-text data stream, the final-form-text data stream does not include general formatting specifications. The process of transforming text from revisable form to final form has converted the formatting specifications into control codes and generated text. (An example of generated text is repetitive headings appearing in the top and bottom margins). The final-form-text data stream therefore contains the original text of the document, interspersed with the generated text, and control codes that cause the output device to print or display the document in the required format.

Final-Form-Text Document Content Architecture thus defines a simple document data stream that a work station can process and present on a printer or display screen. Figure 1-1 on page 1-5 illustrates the difference between a revisable-form-text data stream and a final-form-text data stream.

Document Interchange Architecture

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 or delete them from the library, and to search the library for documents that meet user-specified criteria, such as the name of the author. These criteria are compared with document descriptors that are stored with the document. The user can obtain all documents filed in the document library that meet those criteria.

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

---

2 Such as page width, page depth, margin widths, and placement of headings and pages numbers

3 For example, *character backspace* and *new page*

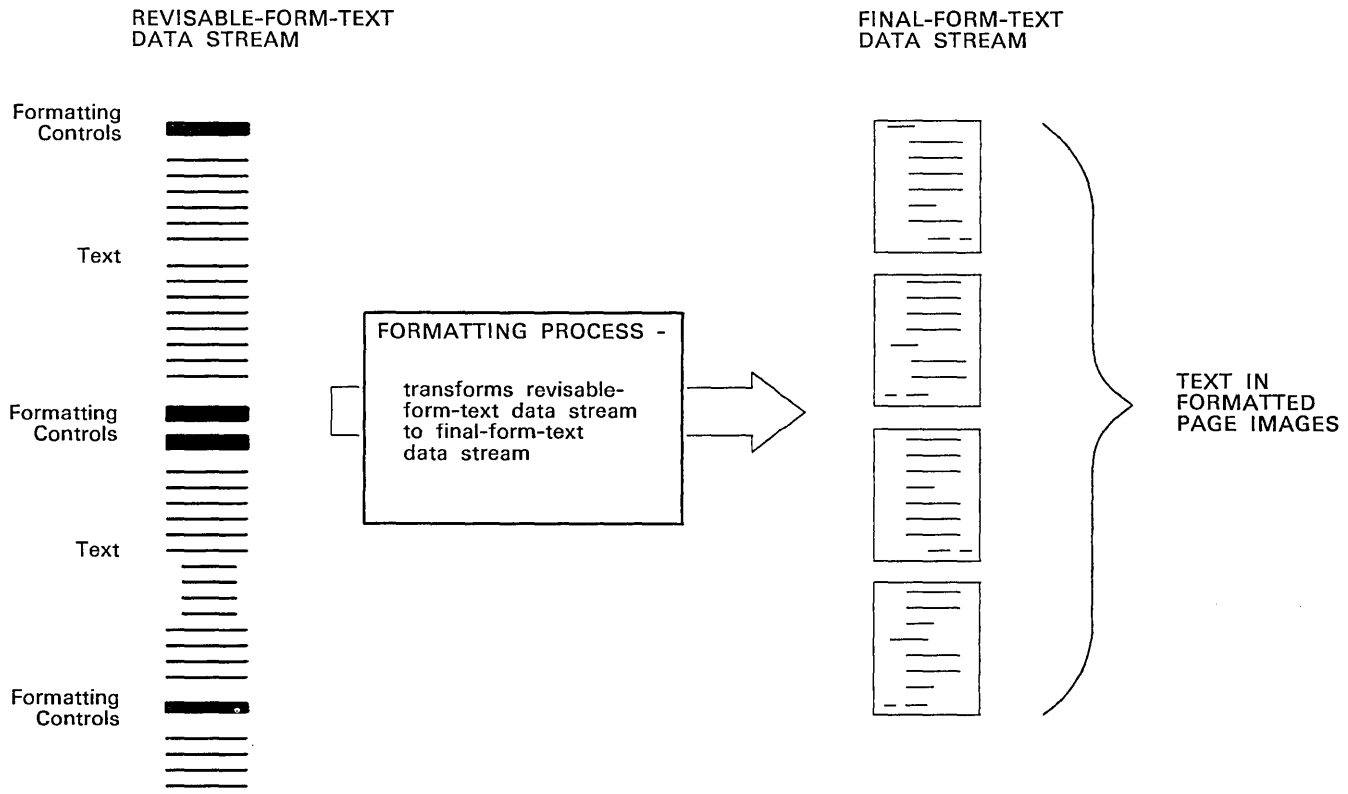

Figure 1-1. Relationship of Revisable-Form-Text and Final-Form-Text Data Streams

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 the Office Information Architectures

Some of the benefits of Document Content Architecture and Document Interchange Architecture and their implementations in IBM office systems are as follows.

General Benefits

The architectures offer the following general benefits to users of IBM office systems.

Document Content Architecture and Document Interchange Architecture form a consistent, comprehensive specification for interchanging information between office systems.

An important aspect of these architectures is their independence from each another. This independence allows each to grow or change without affecting the

other. For example, as new services are required of office systems, new functions can be defined in Document Interchange Architecture without affecting Document Content Architecture. As another example, Document Content Architecture can be extended to accommodate documents containing other information such as images, graphics, and audio without affecting the Document Interchange Architecture.

Benefits of Document Content Architecture

Some specific benefits of Document Content Architecture are as follows.

Revisable-Form-Text Interchange: The ability to distribute document text in revisable form among office systems means that users at different locations can separately develop pieces of the same document, and the pieces can later be pulled together to form the whole document. This form of text interchange is useful for distributing the development and assembly of documents.

Documents can be distributed to different office systems in order to balance work loads, to allow documents to be edited or revised at different locations, and to take advantage of special capabilities at specific locations (such as a high-speed printer). Also, individual sections of a single document can be developed at different locations and then transmitted to a central library where they can be assembled into the complete document.

Thus, field reports that are submitted at branch offices and entered into the office system network as revisable-form text can be reviewed, corrected, and added to at a regional office. Or each individual department of an organization—such as administration, legal, and finance—can prepare its own section of a proposal, with the complete proposal then being assembled from the sections on an office system in a different department.

Final-Form-Text Interchange: The ability to interchange text in final form means that documents that are fixed in content and format can be interchanged among office systems with assurance that the recipients will see the documents just as the originator intended—that is, each recipient will see the same information, in the same format, on the same page. Text whose meaning is dependent on its position within a table, for example, will appear in that position regardless of the kind of office system on which it is printed or displayed.

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, when the permanent archiving of critical information is necessary, or when documents must be obtained by many locations. Document library services provide an organization the means to organize, manage, and control its information assets.

Document Distribution Services: The use of electronic document distribution in offices achieves 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.

Chapter 2. Document Content Architecture

This chapter briefly describes how the format of a document is specified and summarizes the functions provided by Revisable-Form-Text DCA and Final-Form-Text DCA.

Revisable-Form-Text Document Content Architecture

A revisable-form-text document consists of text and format information which directs the presentation of the text. The format information of a document is specified in fields of the data stream called *formatting declarations*. These declarations accompany the document as long as it is in revisable form; the declarations can be modified at any time. Some aspects of a formatting declaration are the page width, page depth, and page numbering scheme used in a document. These are independently specified characteristics of the format called *page elements*.

The originator of a document might, for example, specify that it have several sections, each consisting of pages bearing the section heading centered at the top, and that the pages be numbered sequentially within each section.

The originator does not have to specify the format separately for each page or each section of a document. Some aspects of the format can be stated once at the beginning of the document. Other aspects can be stated within the document—for example, at the beginning of each section. In each case, the originator's specification accompanies the text of the document within the data stream that represents the document.

The revisable-form-text data stream can also include pointers to other documents that are to be combined with the present document at the time it is transformed into final form.

Pages consist of page elements as shown in Figure 2-1 on page 2-2. If all pages are to be formatted the same, then this is specified only once in the document, at the beginning. The space that the body text may occupy is described by the boundaries (left, right, top, and bottom) within which the text appears. Footnotes may be specified to appear at the bottom of the page on which they are referred to, or to be collected at the end of the document.

Functions Provided By Revisable-Form-Text DCA

Most functions that are available with Revisable-Form-Text DCA are defined for and applicable to one or more of the page elements shown in Figure 2-1 on page 2-2.

Some of the functions defined in Revisable-Form-Text DCA are:

- Declare top and bottom margins
- Number pages and lines
- Specify the space occupied by body text
- Specify page width and height
- Insert fields from external data records
- Include text from other documents
- Keep specified text together on the same page
- Spelling verification control

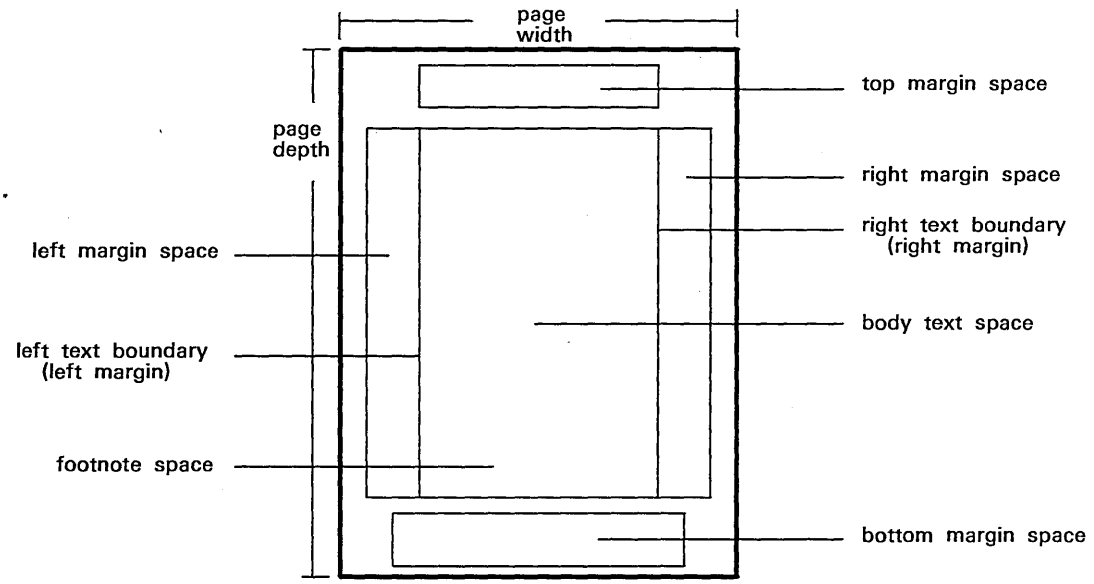

Figure 2-1. Page Elements

Final-Form-Text Document Content Architecture

Final-Form-Text Document Content Architecture defines how the data streams that represent documents to be printed or displayed (rather than revised) are organized. A data stream that contains a final-form text document is meant to be processed sequentially from beginning to end.

When a document is transformed from revisable-form text to final-form text, the formatting declarations specified within the revisable-form-text data stream are converted to formatting control codes within the final-form-text data stream. These codes are imbedded within the text of the document where they are needed in order to format the document. A final-form-text data stream can be interpreted by output devices, such as printers, that may not contain the logic necessary to interpret revisable-form-text data streams.

Documents contained in a final-form-text data stream can be printed on different printers or viewed on different display screens; each will produce the same document content in the same format as the others, within its capabilities.

Functions Provided By Final-Form-Text DCA

The final-form-text document contains formatting control codes at the beginning that establish its initial format, unless they are omitted in favor of predefined default values. The text of the document follows, interspersed with the required formatting control codes.

Among the functions that can be specified by formatting control codes within a final-form-text document are:

- Top margin location
- Left margin location
- Line spacing
- Font definition
- Justify (align) text
- Begin and end underscore
- Begin and end overstrike

Comparing Revisable-Form-Text to Final-Form-Text DCAs

Any functions needed to transform revisable-form-text document control codes and definitions to a final-form-text document are available in Final-Form-Text DCA. However, once a document has been transformed from revisable form to final form, it cannot be returned to its original revisable-form state without interpretation and intervention outside the Final-Form-Text DCA formats and control functions.

Chapter 3. Document Interchange Architecture

This chapter briefly describes 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. These services are performed by processes implemented in the uppermost layer of SNA. DIA specifies how these processes, located throughout the network, communicate with one another to perform required office system functions.

Each DIA service performs specified functions requested by end users. An end user 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 the document library"; and "search the document library for documents that satisfy search criteria *J, K,* and *L.*"

Document Interchange Architecture is considered a part of IBM's Systems Network Architecture (SNA). (Only that part of SNA is introduced by this book; SNA as a whole is described in *SNA Concepts and Products*, GC30-3072.) 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.

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: *source nodes*, *recipient nodes*, and *office system nodes*.

A *source node* provides DIA services, acting on behalf of an end user, that initiate and control the interchange of documents and other information with end users called *recipients*.

A *recipient node* provides DIA services, acting on behalf of an end user (recipient), that control and receive documents and other information 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 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 other information 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 office system node can act as both an originating node and a destination node concurrently. In this case, the originating node address and the destination node address are identical. Similarly, a single node can act concurrently as both a source node and a recipient node, in which case 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

Commands issued by DIA session services enable two DIA processes 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 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.

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 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:

- Allow end users to file documents in, and retrieve or delete them from, the library.
- Allow authorized end users other than the ones that filed the documents to retrieve them from the library.
- Allow 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:

- Place documents received from source nodes into the document library.
- Assign 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.
- Search the profiles of documents in the library that the end user has authority to access and to return to the source node a list of all documents that meet the supplied search criteria.
- Deliver documents to the source node from which they were requested.
- Delete documents from the library upon request from authorized end users.

Document Distribution Services

Document distribution services deliver documents from source nodes to recipient nodes within an office system network. Documents 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 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 document passes from the source node to the office system node. If the recipient node is associated with a different office system node, the document 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 to be delivered, it can take delivery on any or all documents, or it can cancel delivery of any or all documents.

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

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

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

DIA assigns 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:

- Distribute documents and other information to one or more recipients located in the office system network.
- Prioritize the distribution so that documents of higher priority are delivered before documents of lower priority.
- Request that a confirmation-of-delivery message be returned to the sender of a document when the recipient accepts delivery.
- Cancel an outstanding confirmation-of-delivery request. (This cancellation affects only the confirmation request; the request to distribute the document remains in effect.)

- Receive feedback messages relating to the distribution—for example, a notification that the intended recipient is invalid, possibly due to a misspelled recipient address. Feedback messages need not be sent during the same DIA session over which the distribution request flowed.

- Specify that the document is classified as *personal.* A document so classified requires that the intended recipient supply an additional authorization before receiving the document. For example, a manager might distribute personal and confidential information to a group of recipients authorized to receive such material and be assured that only those recipients could receive it.

- Request distributions on behalf of other end users.

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

- Exchange information directly while in a DIA session with the source node.

- Determine which documents are available at the office system node for delivery.

- Obtain documents that are ready at the office system node for delivery (either all documents or only the ones characterized by a particular class of service such as priority, non-priority, or personal).

- Cancel delivery of the recipient's documents that are available at the office system node.

- Request delivery of documents on behalf of other end users.

Document distribution services in an office system node asynchronously distribute documents to recipients located in the office system network. Distributing documents asynchronously means that a recipient node need not have an active DIA session with its office system node to receive documents from the source node. The documents remain in the office system node until the recipient node establishes a DIA session with the destination office system node; then they are 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:

- Assign and return to source nodes a unique distribution document name for each distribution request received.

- Store the distribution request and the document or other information to be distributed.

- Route the distribution request and the associated documents 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 documents to the destination OSN that serves the specified recipients.

- Maintain 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:

- Place distribution requests and documents 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 documents for each recipient listed.

- Deliver distribution requests and documents upon request by recipient nodes.

- Send confirmation-of-delivery messages to the originating OSN when the recipient node takes delivery of the distribution request and document. The originating OSN returns the confirmation-of-delivery message to the source node that requested the confirmation.

- List the names of documents contained in OSN queues for delivery to recipient nodes.

- Cancel delivery of specified documents upon request.

Application Processing Services

Application processing services define 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:

- Request execution of programs within the office system node.

- Request the addition or deletion of descriptors in a document profile.

- Invoke programs to format documents.

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

- Interpret and validate requests from the source nodes.
- Modify descriptors in the document profile specified by end users.
- Schedule execution of programs and procedures requested by end users.
- Execute programs to transform documents from revisable-form text to final-form text.

Chapter 4. Data Streams

A data stream is a continuous stream of data elements being transmitted, or intended for transmission, using a defined format. Document Interchange Architecture defines the format of data streams used to carry information between pairs of DIA services. Document Content Architecture defines the format of data streams that contain information entered by end users into an office system and defines control codes that determine how the end user's information is formatted.

This chapter describes the composition of each data stream as a sequence of *data stream components* and explains the significance of these components to the functions defined by the architectures.

Control Information in Data Streams

Each data stream consists of a sequence of EBCDIC[1] characters, each represented by a byte containing 8 binary digits (bits). Some bytes represent the letters of the alphabet, numerals, symbols, or other printable graphic characters. Others represent control codes that cause specified formatting functions to be performed.

The control information in the data stream is specified using one or more of three general forms.

Single-Byte Control Codes

The first and simplest form of control information is a single-byte control code. The character, usually non-printable, is assigned control significance in the data stream. An example of a single-byte control code is the *required carrier return* code that can be used to mark the end of a line of text on a printer or display screen and cause the following text to be printed or displayed at the beginning of the next line.

Each single-byte control code specifies a single format-control function. The number of control codes that can be specified is limited by the number of single-byte values that can be dedicated in the data stream to control functions.

Multiple-Byte Control Codes

A second form of control code consists of multiple-byte sequences. The first byte identifies the sequence as a multiple-byte control code. The next three bytes indicate the specific control function the code represents and the length of the particular multiple-byte sequence. The control code sequence may include one or more additional parameters, in which case the length value reflects their presence.

An example of this form of control code is the *begin overstrike* code: this marks a character position in the data stream where the subsequent text is to be "overstruck" with another graphic. The overstriking graphic is indicated by a parameter in the control code. A subsequent multiple-byte control code can be used to terminate the overstrike function. This form of control code offers the flexibility of having a variable length that allows control parameters to be specified. It also allows significantly more control codes to be defined than the single-byte form can accommodate.

---

1 Extended Binary Coded Decimal Interchange Code

Structured-Field Control Codes

A third form of control code is the most complex and offers the greatest flexibility. Called a *structured-field control code*, it consists of a 5-byte *introducer* and one or more control fields. The introducer identifies the structure type and states the length of the structured-field control code.

The parameters, or fields, that follow may be fixed or variable in length. In most cases their meaning does not depend on their position in the structured-field control code; therefore they may appear in any order. Because this is the case, optional parameters can be omitted from, and new ones can be added to, existing structured fields. The flexibility of this form of control code allows sets of structured fields to be built if necessary.

An example of a structured field is the *line parameters* declaration that occurs in a revisable-form-text data stream. The structured-field control code contains fields that declare line parameters such as left and right margins, number of lines per inch, and spacing between lines. An example of a set of structured fields is the revisable-form-text *master format unit* that is described later in this chapter. The *line parameters* declaration is just one of several possible structured fields in this set.

The final-form-text data stream uses only single-byte or multiple-byte control codes. DIA services use only structured fields or sets of structured fields. The revisable-form-text data stream uses all three forms of control code.

Components of a Data Stream

The remainder of this chapter briefly describes the components of the revisable-form-text, final-form-text, and DIA data streams and tells where to find detailed information for each.

The Revisable-Form-Text Data Stream

The major components of a revisable-form-text data stream that represents a single document are shown in Figure 4-1 on page 4-3. At least two format units, one text unit (to contain text), and the end unit are required. The body text of the user's document is contained in one or more text units. The number of text units needed to contain the text depends on the number of document pages defined at the time the data stream is constructed. In addition to the text, the text unit may include formatting information that relates to the document page and its page elements.

The format units contain no text except for top-margin and bottom-margin text declarations. The contents of the format units are needed for defining and maintaining the page elements of revisable-form-text documents. The end unit identifies the end of the document.

Format Units

The format units contain definitions and declarations that pertain to the entire document as well as individual page composition (see Figure 4-2 on page 4-4).

Three kinds of format unit exist: *document declaration format units*, *primary master format units*, and *alternate master format units*.

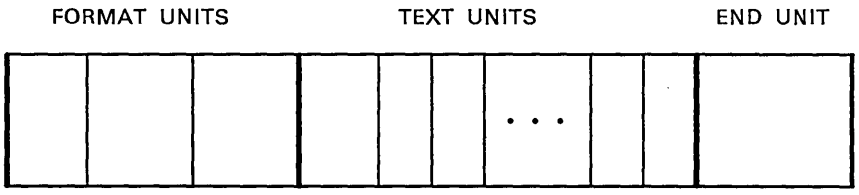

Figure 4-1. Major Components of a Revisable-Form-Text Data Stream

The document declaration format unit contains global document information—that is, information that applies to the document as a whole—as well as punctuation formats for any text to be included in (moved into) the text units. A language dictionary can be specified to assist spelling verification of the document text. The dictionary to be used (English or other languages) is specified in the data stream.

In-line text control codes may be specified by the user to temporarily override global document declarations on a character, word, line, or page basis. For example, user-specified words or phrases in the text can be identified by control codes imbedded in the text data stream that control the spelling verification process for the selected text. This is useful where spelling verification for selected words is not desired. Also, the phrase may be in a different language than that provided by the dictionary selected in the document declaration, hence, a different language dictionary is selected.

The master format units contain information and declarations that relate to page composition and formatting. If, in the simple case, all pages of a document are to be composed and formatted the same, then the master format need be selected only once and can remain the active, or current, format throughout the entry or edit process.

The revisable-form-text data stream must contain the document declaration format unit and the primary master format unit. The alternate master format unit is optional; it can be used when a single master format is not sufficient.

Text Units

The body text is contained in one or more text units; each text unit represents a page of the document. The text unit structure, however, allows this page definition to change as new text is added or current content modified. A later instance of the data stream (as the result of a revision or formatting process) for the same document may have different text units defined, because page formats may have been changed, or pages added or deleted.

The data stream may contain additional text units that do not represent pages of text. These are system-defined text units and contain information referred to from the body text, such as source text for footnote references.

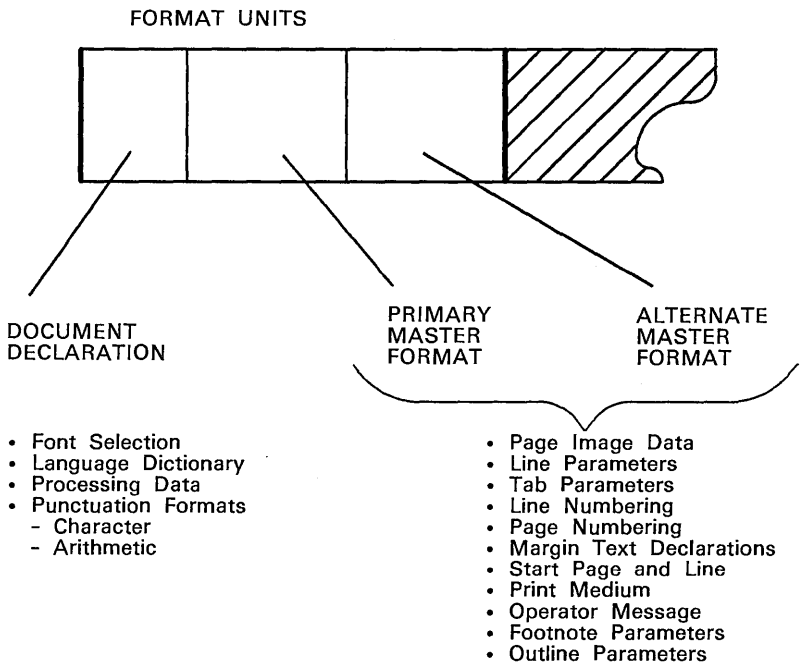

Figure 4-2. Revisable-Form-Text Data Stream Format Units

Each text unit can contain an optional format declaration for the page in addition to the body text (see Figure 4-3 on page 4-5). The format declaration in the text unit is used when the format of the page is to vary from that previously established. The format declaration may temporarily change the format for the current page only, or it may select or re-establish a master format.

The body-text part of the text unit contains text plus imbedded control codes that complement or override control codes in the format unit declaration. Some of the text formatting control codes that may be included in the body text are:

- Footnote references
- Release left margin
- Begin/end overstrike
- Change font
- Begin and end "keep" (span of text be kept together on one page)
- Set horizontal tab
- Align text field
- Copy data from a data base record
- Begin line format change (such as left or right margin change)

The formatting control codes also allow users to define the layout of columns of text and tables, check the text of a document for misspelled words, and control visual attributes of a display station (such as the shape or blinking of the cursor).

Single-byte format control codes are used to mark line end positions, insert subscript or superscript characters, locate system-supplied hyphenation points, note tab requests, and perform other control functions that relate to positioning of text.

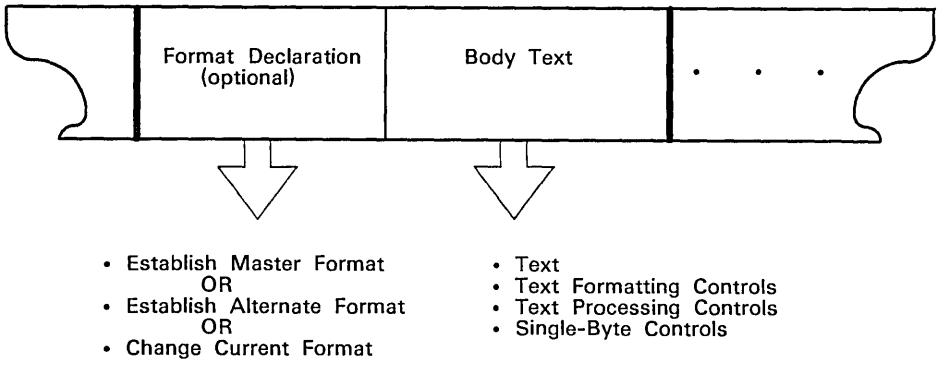

Figure 4-3. Revisable-Form-Text Data Stream Text Unit

End Units

The final component in the revisable-form-text data stream is the end unit: it marks the end of the revisable-form-text data stream for the document. It contains no other information about the document content.

For more detail about the revisable-form-text data stream, see *Document Content Architecture: Revisable-Form-Text Reference*, SC23-0758.

The Final-Form-Text Data Stream

The data stream for final-form-text consists of the text of the document and control codes imbedded in the text. Control codes activate functions to format the text when it is printed or displayed.

The control codes may be either single-byte or multiple-byte codes. Both kinds can appear throughout the data stream. However, some control codes can occur only on a line boundary and some can occur only on a page boundary. For example, a *set vertical margin* control code can occur only at a page boundary and a *set horizontal tab settings* control code can occur only at a line boundary.

Multiple-byte control codes may appear before the start of text or imbedded within the text and usually remain in effect throughout the document or until reset. Figure 4-4 on page 4-6 shows an example of the organization of a final-form-text data stream.

When an office system recognizes that it is to print or display a final-form-text document, it initializes itself to a predefined set of default control values and parameters. Therefore, if the formatting requirements of the document match the default control values, the data stream could start immediately with the first line of text. Only the control values that differ from the defaults are required.

For more information about Final-Form-Text DCA, see *Document Content Architecture: Final-Form-Text Reference*, SC23-0757.

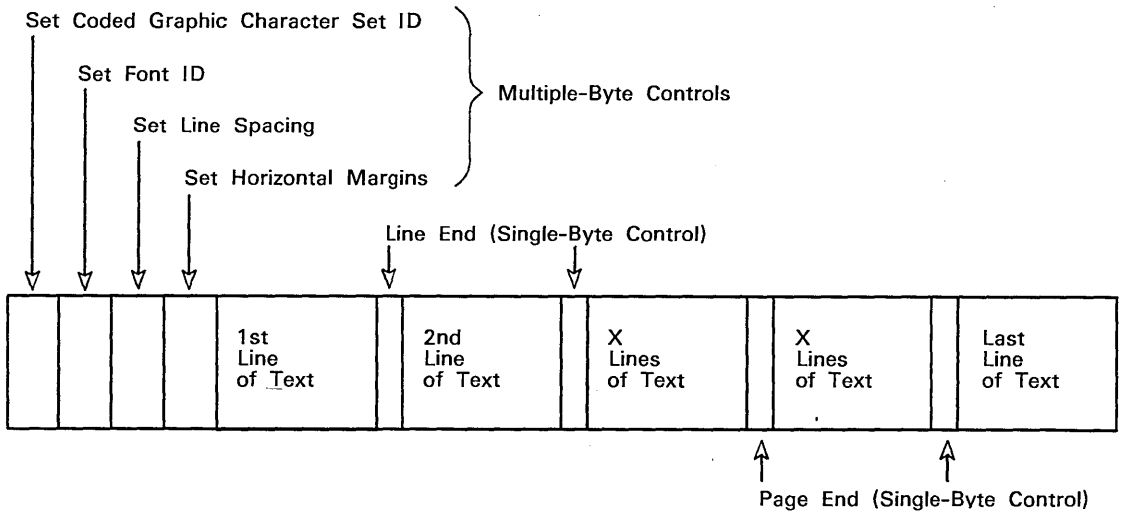

Figure 4-4. Typical Final-Form-Text Data Stream Organization

The DIA Data Stream

The basic unit of interchange exchanged between DIA processes is the *document interchange unit* (DIU). The DIU consists of the following five data stream components (see Figure 4-5 on page 4-7):

- The *prefix* introduces and identifies the information that follows in the data stream as a document interchange unit.
- The *command sequence* contains the DIA command that specifies the function to be performed.
- The *data unit* contains information that may be referred to by the DIA command in the command sequence. This field is optional and is present when defined by the command.
- The *document unit* contains the document profile and optionally the document content (text and control codes). This field is optional and is present only when a document profile and content are sent from one DIA process to another.
- The *suffix* specifies the end of the DIU and indicates whether any abnormal conditions occurred while the DIU was being transmitted.

These data stream components may be composed of substructures called *subcomponents*. Examples of subcomponents are command operands and document profiles.

All DIU components and their subcomponents begin with a structured field called an *introducer*. The introducer uniquely identifies each field and indicates its length. Consequently all fields and components (and hence, the entire data stream) are self describing and may be variable in length.

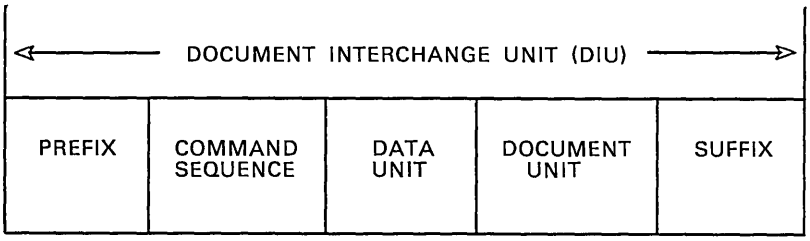

Figure 4-5. Document Interchange Unit

For more detailed information about the DIA data stream, see *Document Interchange Architecture: Concepts and Structures*, SC23-0759.

Relationships between Data Streams

Figure 4-6 on page 4-8 shows how the document-content data stream, the DIA data stream, and the SNA message units are related. As an end user at a work station creates or edits a document, the work station generates a document-content data stream. This may be a revisable-form-text or a final-form-text data stream, as determined by the end user and the capabilities of the work station. Document distribution services in the work station insert this data stream into a DIA document interchange unit, thus forming a DIA data stream. The presentation and transport services of SNA in turn surround the DIA data stream with other SNA control data to form the message units that flow through the SNA network.

A corresponding process occurs when the document reaches its destination; that is, the lower-layer SNA data is removed from around the DIA data stream, then the DIA data is removed from around the document-content data stream.

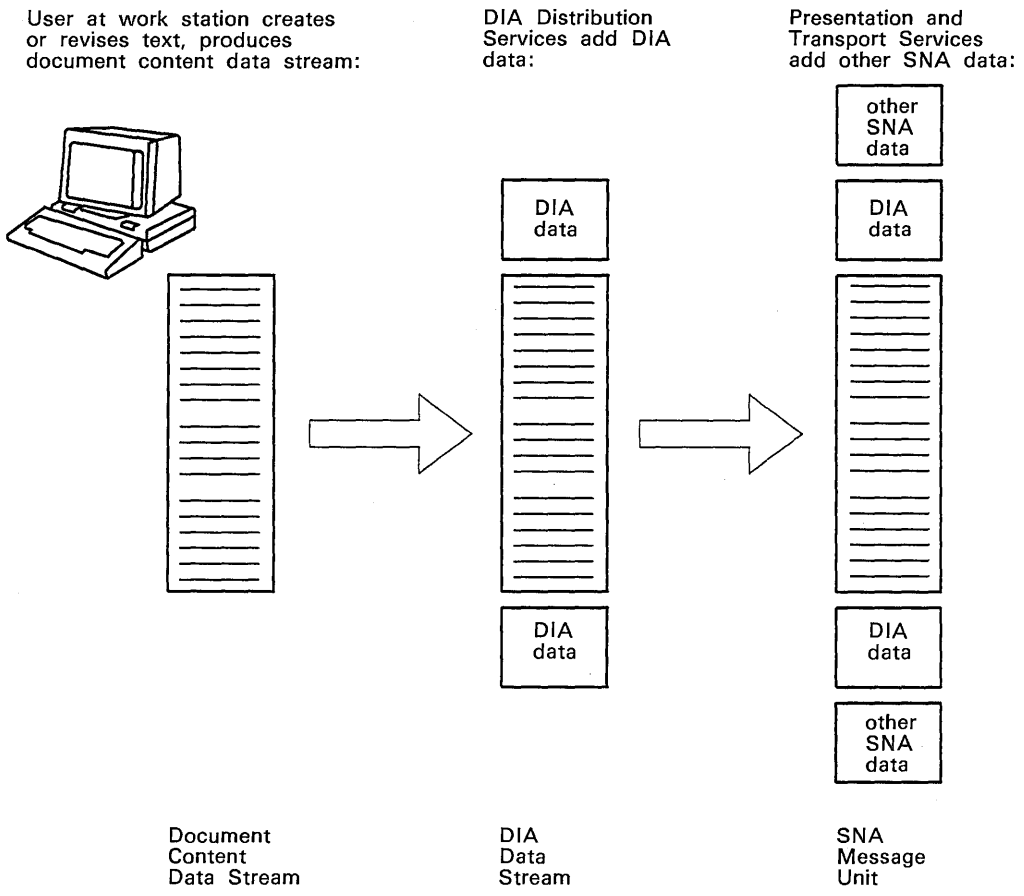

Figure 4-6. Relationship of Document-Content and DIA Data Streams and SNA Message Units

Index

Office Information
Architectures: Concepts

Order No. GC23-0765-0

READER'S COMMENT FORM

You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you. Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name, company, mailing address, and date:

_____

_____

_____

_____

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

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

GC23-0765-0

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®

GC23-0765-0

Office Information Architechtures: Concepts Printed in U.S.A. GC23-0765-0

IBM

GC23-0765-0